Abstract: New computational technologies generating systematic investment portfolios by coordinating forecasting algorithms contributed by researchers are provided. Work on challenges is efficiently facilitated by the algorithm developer's sandbox ("ADS"). Second, the algorithm selection system performs a batch of tests that selects the best developed algorithms, updates the list of open challenges and translates those scientific forecasts into financial predictions. The algorithm controls for the probability of backtest overfitting and selection bias, thus providing for a practical solution to a major flaw in computational research involving multiple testing. Third, the incubation system verifies the reliability of those selected algorithms. Fourth, the portfolio management system uses the selected algorithms to execute investment recommendations. A dynamically optimal portfolio trajectory is determined by a quantum computing solution to combinatorial optimization representation of the capital allocation problem. Fifth, the crowdsourcing of algorithmic investments controls the workflow and interfaces between all of the hereinafter introduced components.
FIELD OF THE INVENTION

The present invention relates to systems and methods for improved forecasting and generation of investment portfolios based upon algorithmic forecasts.

BACKGROUND OF THE INVENTION

Computational forecasting systems are important and widely used as essential tools in finance, business, commerce, governmental agencies, research organizations, environment, sciences, and other institutions. There are myriad different reasons why disparate organizations need to predict as accurately as possible future financial or scientific trends and events. Many different types of forecasting systems and methods have been developed over the years including highly complex and sophisticated financial forecasting systems, business demand forecasting systems, and many other computational forecasting methods and systems. While current methods appear to have justified the expenses incurred in developing and purchasing them, there is a growing demand in many of the above-mentioned pes of organizations for accurate, improved, novel, and differentiated computational forecasting algorithms. At least in the financial industry, forecasting systems have had deficiencies including but not limited to products that have limited investment capabilities, models based on spurious relationships, lack of appropriate analysis of overfitting, reliance on staff analysts’ discretion, and limited capability to evaluate forecast algorithms. These and other drawbacks may not be limited only to financial systems.

To further clarify, companies have in the past implemented significant software and hardware resources to accurately develop forecast algorithms. In one respect, companies hire a staff of analysts with the primary directive of forecasting. One drawback of this approach is that individuals on the staff appear, over time, to converge to have similar approaches or ideas. As such, diversity in thought and creativity is lost. For example, standard business practice for alpha generation is to hire portfolio managers with good track records, typically expressed in terms of high Sharpe ratios. This often leads to selecting portfolio managers with similar traits, which happened to do well in previous years. And even if these portfolio managers were originally selected for being complementary, their daily interaction and work on the same platform will tend to undermine that sought diversification. The consequence is...
the misuse of capital and resources, because these portfolio managers will tend to perform as one.

Another drawback is that the individual experts that are focused on a career in a particular field of science are the best people in that field of science to create corresponding forecasting algorithms. Pursuing forecasting algorithm contributions from others can be a deficient approach because those individuals likely have their own primary field of endeavor that is different from the needed field of expertise. Our invention facilitates the contribution of forecasting algorithms by those who are experts in the relevant field of science, so that such contribution does not require them to abandon their field or make a career change.

Another issue of relevance relates to the computer resources that institutions consume to accomplish the development of forecast algorithms and apply to production using the forecast algorithms. In many cases, institutions apply significant computer resources in these endeavors where improvement in the process and improved accuracy can significantly improve (e.g., reduce) the need for computational resources (e.g., memory, processors, network communications, etc.) and thereby provide improved accuracy at a much quicker rate.

Another area of deficiency relates to performance evaluation systems. Traditionally, investment funds allocate capital to portfolio managers or algorithms following a heuristic procedure. Those allocations are reviewed on a quarterly or semi-annual basis, based on previous performance as well as subjective considerations. This inevitably leads to inconsistent and erroneous investment decisions.

Another related issue has to do with problems connected to research-based projects. There is discussion in academic papers that explains problems associated with such research in which the results or proposed forecast algorithms, in the case, can be inaccurate or not trustworthy. This can include situations involving backtest overfitting or selection bias. For example, as multiple tests take place on a same dataset, there is an increased probability of encountering false positives. Because many scientific research processes do not account for this increase in probability of false positives, several scientists have concluded that most published research findings are false, Ioannidis JPA (2005), Why Most Published Research Findings Are False, PLoS Med 2(8): e124. doi: 10.1371/journal.pmed.0020124. Available at http://journals.plos.org/plosmedicine/article?id = 10.1371/journal.pmed.0020124. A practical solution to address critical flaws in the modern scientific method is therefore in high demand.
Thus in view of the hereinabove, the presently disclosed embodiments of the invention now provide such solutions creating an interface between scientists and investors, and also provide other advantages that are understood from the present disclosure.

SUMMARY OF THE INVENTION

In accordance with a preferred non-limiting embodiment of the present invention, a computer-implemented system for automatically generating financial investment portfolios is contemplated. The system may comprise an online crowdsourcing site having one or more servers and associated software that configures the servers to provide the crowdsourcing site, and further comprise a database of open challenges and historic data. The site may register experts, who access the site from their computers, to use the site over a public computer network, publishes challenges on the public computer network wherein the challenges include challenges that define needed individual scientific forecasts for which forecasting algorithms are sought, and implements an algorithmic developers sandbox that may comprise individual private online workspaces that are available remotely accessible for use to each registered expert and which include a partitioned integrated development environment comprising online access to algorithm development software, historic data, forecasting algorithm evaluation tools including one or more tools for performing test trials using the historic data, and a process for submitting one of the expert's forecasting algorithms authored in their private online workspace to the system as a contributed forecasting algorithm for inclusion in a forecasting algorithm library.

The system may further comprise an algorithm selection system comprising one or more servers and associated software that configures the servers to provide the algorithm selection system, wherein on the servers, the algorithm selection system receives the contributed forecast algorithms from the algorithmic developers sandbox, monitors user activity inside the private online workspaces including user activity related to the test trials performed within the private online workspaces on the contributed forecasting algorithms before the contributed forecasting algorithms were submitted to the system, determines from the monitored activity test related data about the test trials performed in the private online workspaces on the contributed forecasting algorithms including identifying a specific total number of times a trial was actually performed in the private online workspace on the contributed forecasting algorithm by the registered user, determines accuracy and performance of the contributed forecasting algorithms using historical data and analytics software tools including determining from the test related data a corresponding probability of
backtest overfitting associated with individual ones of the contributed forecasting algorithms, and, based on determining accuracy and performance, identifies a subset of the contributed forecasting algorithms to be candidate forecasting algorithms.

The system may further comprise an incubation system comprising one or more servers and associated software that configures the servers to provide the incubation system, wherein on the servers, the incubation system receives the candidate forecasting algorithms from the algorithm selection system, determines an incubation time period for each of the candidate forecasting algorithms by receiving the particular probability of backtest overfitting for the candidate forecasting algorithms and receiving minimum and maximum ranges for the incubation time period, in response determines a particular incubation period that varies between the maximum and minimum period based primarily on the probability of backtest overfitting associated with that candidate forecasting algorithm, whereby certain candidate forecasting algorithms will have a much shorter incubation period than others, includes one or more sources of live data that are received into the incubation system, and applies the live data to the candidate forecasting algorithms for a period of time specified by corresponding incubation time periods, determines the accuracy and performance of the candidate forecasting algorithms in response to the application of the live data including by determining accuracy of output values of the candidate forecast algorithms when compared to actual values that were sought to be forecasted by the candidate forecasting algorithms, and in response to determining accuracy and performance of the candidate forecasting algorithms, identifies and stores a subset of the candidate forecasting algorithms as graduate forecasting algorithms as a part of a portfolio of operational forecasting algorithms that are used to forecast values in operational systems.

In a further embodiment, the system may implement a source control system that tracks iterative versions of individual forecast algorithms, while the forecast algorithms are authored and modified by users in their private workspace. The system may determine test related data about test trials performed in the private workspace in specific association with corresponding versions of an individual forecasting algorithm, whereby the algorithm selection system determines the specific total number of times each version of the forecasting algorithm was tested by the user who authored the forecasting algorithm. The system may determine the probability of backtest overfitting using information about version history of an individual forecast algorithm as determined from the source control system. The system may associate a total number of test trials performed by users in their private workspace in association with a corresponding version of the authored forecasting algorithm by that user.
The system determines, from the test data about test trials including a number of test trials and the association of some of the test trials with different versions of forecast algorithms, the corresponding probability of backtest overfitting.

In a further embodiment, the system may include a fraud detection system that receives and analyzes contributed forecasting algorithms, and determines whether some of the contributed forecasting algorithms demonstrate a fraudulent behavior.

In a further embodiment, the online crowdsourcing site may apply an authorship tag to contributed forecasting algorithm and the computer-implemented system maintains the authorship tag in connection with the contributed forecasting algorithm including as part of a use of the contributed forecasting algorithm as a graduate forecasting algorithm in operation use. The system may determine corresponding performance of graduate algorithms, and then generates an output, in response to the corresponding performance that is communicated to the author identified by the authorship tag. In some embodiments, the output may further communicate a reward.

In a further embodiment, the system may further comprise a ranking system that ranks challenges based on corresponding difficulty.

In a further embodiment, the algorithm selection system may include a financial translator that comprises different sets of financial characteristics that are associated with specific open challenges, wherein the algorithm selection system determines a financial outcome from at least one of the contributed forecasting algorithms by applying the set of financial characteristics to the at least one of the contributed forecast algorithms.

In a further embodiment, the system may further comprise a portfolio management system having one or more servers, associated software, and data that configures the servers to implement the portfolio management system, wherein on the servers, the portfolio management system receives graduate forecasting algorithms from the incubation system, stores graduate forecasting algorithms in a portfolio of graduate forecasting algorithms, applies live data to the graduate forecasting algorithms, and in response, receives output values from the graduate forecasting algorithms, determines directly or indirectly, from individual forecasting algorithms and their corresponding output values, specific financial transaction orders, and transmits the specific financial transaction orders over a network to execute the order. The portfolio management system may comprise at least two operational modes. In the first mode, the portfolio management system processes and applies graduate forecasting algorithms that are defined to have an output that is a financial output and the portfolio management system determines from the financial output the specific financial
order. In the second mode, the portfolio management system processes and applies graduate forecasting algorithm that are defined to have an output that is a scientific output, applies a financial translator to the scientific output, and the portfolio management system determines from the output of the financial translator a plurality of specific financial orders that when executed generate or modify a portfolio of investments that are based on the scientific output. The portfolios from these first and second modes are "statically" optimal, in the sense that they provide the maximum risk-adjusted performance at various specific investment horizons.

In another embodiment, the statically optimal portfolios that resulted from the first and second mode are further subjected to a "global" optimization procedure, which determines the optimal trajectory for allocating capital to the static portfolios across time. In this embodiment, a procedure is set up to translate a dynamic optimization problem into an integer programming problem. A quantum computer is configured to solve this integer programming problem by making use of linear superposition of the solutions in the feasibility space. As such, in one embodiment, the portfolio management system may comprise a quantum computer configured with software that together processes graduate forecasting algorithms and indirect cost of associated financial activity, and in response determines modifications to financial transaction orders before being transmitted, wherein the portfolio management system modifies financial transaction orders to account for overall profit and loss evaluations over a period of time. In another embodiment, the portfolio management system may comprise a quantum computer that is configured with software that together processes graduate forecasting algorithms by generating a range of parameter values for corresponding financial transaction orders, partitioning the range, associating each partition with a corresponding state of a qubit, evaluating expected combinatorial performance of multiple algorithms overtime using the states of associated qubits, and determining as a result of the evaluating, the parameter value in the partitioned range to be used in the corresponding financial transaction order before the corresponding financial transaction order is transmitted for execution.

In a further embodiment, the portfolio management system is further configured to evaluate actual performance outcomes for graduate forecasting algorithms against expected or predetermined threshold performance outcomes for corresponding graduate forecast algorithm, based on the evaluation, determine underperforming graduate forecasting algorithms, remove underperforming graduate forecasting algorithms from the portfolio, and communicate actual performance outcomes, the removal of graduate algorithms, or a status of
graduate forecasting algorithms to other components in the computer-implemented system. The portfolio management system evaluates performance of graduate forecasting algorithms by performing a simulation after live trading is performed that varies input values and determines variation in performance of the graduate forecasting algorithm portfolio in response to the varied input values, and determines from the variations in performance, to which ones of the graduate forecasting algorithms in the portfolio the variations should be attributed.

In a further embodiment, the algorithm selection system is further configured to include a marginal contribution component that determines a marginal forecasting power of a contributed forecasting algorithm, by comparing the contributed forecasting algorithm to a portfolio of graduate forecasting algorithm operating in production in live trading, determines based on the comparison a marginal value of the contributed forecasting algorithm with respect to accuracy, performance, or output diversity when compared to the graduate forecasting algorithms, and in response, the algorithm selection system determines which contributed forecasting algorithm should be candidate forecasting algorithm based at least partly on the marginal value.

In a further embodiment, the algorithm selection system is further configured to include a scanning component that scans contributed forecasting algorithms, and in scanning searches for different contributed forecasting algorithms that are mutually complementary. The scanning component determines a subset of the contributed forecasting algorithms that have defined forecast outputs that do not overlap.

In a further embodiment, the incubation system may further comprise a divergence component that receives and evaluates performance information related to candidate forecasting algorithm, over time, determines whether the performance information indicates that individual candidate forecasting algorithm systems have diverged from in sample performance values determined prior to the incubation system, and terminates the incubation period for candidate forecasting algorithms that have diverged from their in-sample performance value by a certain threshold.

In accordance with another preferred non-limiting embodiment of the present invention, a computer-implemented system for automatically generating financial investment portfolios is contemplated. In this system, it may include an online crowdsourcing site comprising one or more servers and associated software that configures the servers to provide the crowdsourcing site and further comprising a database of challenges and historic data,
wherein on the severs, the site publishes challenges to be solved by users, implements a development system that comprises individual private online workspaces to be used by the users comprising online access to algorithm development software for solving the published challenges to create forecasting algorithms, historic data, forecasting algorithm evaluation tools for performing test trials using the historic data, and a process for submitting the forecasting algorithms to the computer-implemented system as contributed forecasting algorithms.

The system may also include an algorithm selection system comprising one or more servers and associated software that configures the servers to provide the algorithm selection system, wherein on the servers, the algorithm selection system receives the contributed forecast algorithms from the development system, determines a corresponding probability of backtest overfitting associated with individual ones of the received contributed forecasting algorithms, and based on the determined corresponding probability of backtest overfitting, identifies a subset of the contributed forecasting algorithms to be candidate forecasting algorithms.

The system further includes an incubation system comprising one or more servers and associated software that configures the servers to provide the incubation system, wherein on the servers, the incubation system receives the candidate forecasting algorithms from the algorithm selection system determines an incubation time period for each of the candidate forecasting algorithms, applies live data to the candidate forecasting algorithms for a period of time specified by corresponding incubation time periods, determines accuracy and performance of the candidate forecasting algorithms in response to the application of the live data, and in response to determining accuracy and performance of the candidate forecasting algorithms, identifies and stores a subset of the candidate forecasting algorithms as graduate forecasting algorithms as a part of a portfolio of operational forecasting algorithms that are used to forecast values in operational systems.

In accordance with yet another preferred non-limiting embodiment of the present invention, a computer-implemented system for automatically generating financial investment portfolios is contemplated. In this system, it may include a site comprising one or more servers and associated software that configures the servers to provide the site and further comprising a database of challenges, wherein on the severs, the site publishes challenges to be solved by users, implements a first system that comprises individual workspaces to be used by the users comprising access to algorithm development software for solving the published challenges to create forecasting algorithms, and a process for submitting the
forecasting algorithms to the computer-implemented system as contributed forecasting algorithms.

The system may also include a second system comprising one or more servers and associated software that configures the servers to provide the second system, wherein on the servers, the second system evaluates the contributed forecast algorithms, and based on the evaluation, identifies a subset of the contributed forecasting algorithms to be candidate forecasting algorithms.

The system may further include a third system comprising one or more servers and associated software that configures the servers to provide the third system, wherein on the servers, the third system determines a time period for each of the candidate forecasting algorithms, applies live data to the candidate forecasting algorithms for corresponding time periods determined, determines accuracy and performance of the candidate forecasting algorithms in response to the application of the live data, and based on the determination of accuracy and performance, identifies a subset of the candidate forecasting algorithms as graduate forecasting algorithms, the graduate forecasting algorithms are a part of a portfolio of operational forecasting algorithms that are used to forecast values in operational systems.

In accordance with yet another preferred non-limiting embodiment of the present invention, a computer-implemented system for developing forecasting algorithms is contemplated. In this system, it may include a crowdsourcing site which is open to the public and publishes open challenges for solving forecasting problems; wherein the site includes individual private online workspace including development and testing tools used to develop and test algorithms in the individual workspace and for users to submit their chosen forecasting algorithm to the system for evaluation.

The system may also include a monitoring system that monitors and records information from each private workspace that encompasses how many times a particular algorithm or its different versions were tested by the expert and maintains a record of algorithm development, wherein the monitoring and recording is configured to operate independent of control or modification by the experts.

The system may further include a selection system that evaluates the performance of submitted forecasting algorithms by performing backtesting using historic data that is not available to the private workspaces, wherein the selection system selects certain algorithms that meet required performance levels and for those algorithms, determines a probability of backtest overfitting and determines from the probability, a corresponding incubation period for those algorithm that varies based on the probability of backtest overfitting.
Counterpart methods and computer-readable medium embodiments would be understood from the above and the overall disclosure. Also, to emphasize, broader, narrower, or different combinations of described features are contemplated, such that, for example features can be removed or added in a broadening or narrowing way.

**BRIEF DESCRIPTION OF THE DRAWINGS**

The nature and various advantages of the present invention will become more apparent upon consideration of the following detailed description, taken in conjunction with the accompanying drawings, in which like reference characters refer to like parts throughout, and in which:

FIG. 1 depicts an illustrative embodiment of a system for crowdsourcing of algorithmic forecasting in accordance with some embodiments of the present invention.

FIG. 2 depicts an illustrative embodiment of a development system associated with developing an algorithm in accordance with some embodiments of the present invention.

FIG. 3 depicts an illustrative embodiment of a development system associated with developing an algorithm in accordance with some embodiments of the present invention.

FIG. 4 depicts an illustrative embodiment of a selection system associated with selecting a developed algorithm in accordance with some embodiments of the present invention.

FIG. 5 depicts an illustrative embodiment of a selection system associated with selecting a developed algorithm in accordance with some embodiments of the present invention.

FIG. 6 depicts an illustrative incubation system in accordance with some embodiments of the present invention.

FIG. 7 depicts an illustrative incubation system in accordance with some embodiments of the present invention.

FIG. 8 depicts an illustrative management system in accordance with some embodiments of the present invention.

FIG. 9 depicts one mode of capital allocation in accordance with some embodiments of the present invention.

FIG. 10 depicts another mode capital allocation in accordance with some embodiments of the present invention.

FIG. 11 depicts an illustrative embodiment of a crowdsourcing system in accordance with embodiments of the present invention.
FIGS. 12-16 illustrate example data structure in or input/output between systems within the overall system in accordance with embodiments of the present invention.

FIG. 17 depicts an illustrative core data management system in accordance with some embodiments of the present invention.

FIG. 18 depicts an illustrative backtesting environment in accordance with some embodiments of the present invention.

FIG. 19 depicts an illustrative paper trading system in accordance with some embodiments of the present invention.

FIGS. 20-22 depict various illustrative alert notifications and alert management tools for managing the alert notifications in accordance with some embodiments of the present invention.

FIG. 23 depicts an illustrative deployment process or deployment process system in accordance with some embodiments of the present invention.

FIG. 24 depicts a screen shot of an illustrative deployment tool screen from intra web in accordance with some embodiments of the present invention.

FIG. 25 depicts an illustrative parallel processing system in accordance with some embodiments of the present invention.

FIG. 26 depicts an illustrative performance evaluation system in accordance with some embodiments of the present invention.

FIG. 27 depicts an illustrative screen shot of the performance results generated by the performance evaluation system or the performance engine on the infra web in accordance with some embodiments of the present invention.

FIG. 28 depicts a screen shot of an illustrative intra web in accordance with some embodiments of the present invention.

FIG. 29 depicts illustrative hardware and software components of a computer or server employed in a system for crowdsourcing of algorithmic forecasting in accordance with some embodiments of the present invention.

**DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS**

Improving the accuracy and rate at which forecasting algorithms are developed, tested, and deployed can have significant value to the scientific, business, and financial community. The evaluation of algorithms can involve significant amount of data, processing, and risk (e.g., if the algorithm is inaccurate in production). In addition, the development of forecasting algorithm can be complex and require multiple iterations.
In accordance with embodiments of the present invention, a system is deployed that combines different technical aspects to arrive on improved systems. In one respect, the system implements an online crowdsourcing site that publicizes open challenges for experts to engage. The challenges can be selected by the system automatically based on analysis already performed. The crowdsourcing site can not only publish the challenges but also provide each expert with an online algorithm developers sandbox. The site will give each expert that chooses to register, the ability to work virtually in the sandbox in a private workspace containing development tools such as algorithm development software, evaluation tools, and available storage. The private workspace provides a virtual remote workspace and is partitioned and private from other registered experts so that each expert can develop a forecast algorithm for a challenge independently and in private. In other words, the system is configured to prevent other experts registered on the site from being able to accessor see the work of other experts on the site. However, the system implements certain limitations on maintaining the privacy of in-workspace data or activity as described below.

The system includes the interactive option for the expert to apply historic data to their authored algorithm to test the performance of the algorithm in their workspace. This is accomplished by the system providing the option to perform one or more trials in which the system applies historic data to the expert's authored forecast algorithm. The system will further include additional interactive features such as the ability in which each expert can select to submit and identify one of their authored forecasting algorithms (after conducting test trials in the workspace) to the system for evaluation. In response to the user's selection of an algorithm for contribution, the system will transmit a message from the expert to another part of the system and the message, for example, will contain the contributed forecast algorithm or may have a link to where it is saved for retrieval.

The system includes an algorithm selection system that receives contributed forecasting algorithms from the crowd or registered experts on the site. The algorithm selection system includes features that apply evaluation tools to the contributed forecast algorithm. As part of the evaluation, the system generates a confidence level in association with each contributed forecast algorithm and applies further processing to deflate the confidence level. In particular, the overall system is configured to private workspaces that are partitioned and private between experts, but the system is further configured to track and store at least certain activity within the private workspace. The system is configured to monitor and store information about test trials that the expert performed in the workspace on the contributed algorithm. This includes the number of test trials that the expert performed
on the contributed forecast algorithm (e.g., before it was sent to the system as a contribution for evaluation). In the algorithm selection system, the algorithm selection system can select forecasting algorithms based on performing additional testing, or evaluation of the contributed forecasting algorithms and/or can select contributed forecasting algorithms that meet matching criteria such as the type of forecast or potential value of the forecast. In response, the system identifies certain contributed forecasting algorithms as candidate algorithms for more intensive evaluation, in particular testing within an incubation system. As part of this, the system retrieves information about test trials performed in a private workspace and applies that information to determine a deflated confidence level for each contributed forecasting algorithm. In particular, for example, the total number of trials that the expert performed on the algorithm is retrieved and is used to determine a probability of backtest overfitting of the forecast algorithm. Other data, such as from the prior test data in the workspace can also be used as part of this determination and process.

The deflated confidence level can be the same as the probability of backtest overfitting (“PBO”), or PBO can be a component of it. The purpose is that this value is applied by the system to determine the incubation period for each contributed forecasting algorithm that is moving to the next stage as a candidate forecasting algorithm. There are often-times in such systems a standard approach in which a preset incubation period is used for all algorithms. The confidence level, or PBO, is applied by the system to the standard incubation period, and by applying it, the system determines and specifies different incubation periods for different candidate forecasting algorithms. This is one way that the system reduces the amount of memory and computational resources that are used in the algorithm development process. Reducing the incubation period for some candidate forecasting algorithms can also allow a quicker time to production and more efficient allocation of resources.

The determined incubation period is applied in an incubation system that receives candidate forecasting algorithms. The incubation system is implemented to receive live data (current data, e.g., as it is generated and received as opposed to historic data that refers to data from past periods), and to apply the live data to the candidate forecasting algorithms. The incubation system is a pre-production system that is configured to simulate production but without applying the outputs of the candidate forecasting algorithms to real-life applications. For example, in the financial context, the incubation system will determine financial decisions and will generate financial transaction orders but the orders are virtually executed based on current market data at that time. The incubation system evaluates this
virtual performance in "an almost production" setting over the specific incubation period. The incubation system evaluates the performance of candidate forecasting algorithms and based on the evaluation, determines which candidate forecasting algorithms should be selected to be graduate forecasting algorithms for inclusion in the portfolio of graduate forecasting algorithms. The portfolio of graduate forecasting algorithms will be part of the production system.

The production system, a system that is in operative commercial production, can include a management system that controls the use of graduate forecasting algorithms in the portfolio. The production system can determine the amount of financial capital that is allocated to different graduate forecasting algorithms. The production system can also apply financial translators to the graduate forecasting algorithms and, based on the information about the financial translators generate a portfolio involving different investments.

Overall, the system and its individual systems or components implement a system for crowdsourcing of algorithmic forecasting (which can include different combination of features or systems as illustratively described herein or would otherwise be understood). With respect to financial systems, the system (which for convenience is also used sometimes to refer to methods and computer readable medium) can generate systematic investment portfolios by coordinating the forecasting algorithms contributed by individual researchers and scientists. In its simplest form, embodiments of the system and method can include i) a development system (as a matter of brevity and convenience, the description of systems, components, features and their operation should also be understood to provide a description of steps without necessarily having to individually identify steps in the discussion) for developing a forecasting algorithm (which is sometimes referred to as a development system, algorithm development system, or algorithmic developer's sandbox), ii) a selection system for selecting a developed algorithm (which is sometimes referred to as an algorithm selection system), iii) an incubation system for incubating a selected algorithm (which is sometimes referred to as an incubation of forecasting algorithms systems), iv) a management system for managing graduate forecasting algorithm (which is sometimes referred to as a portfolio management system or management of algorithmic strategies system), and v) a crowdsourcing system for the development system that is used to promote and develop new high quality algorithms.

For clarification, different embodiments may implement different components in different parts of the system for illustration purposes. In addition, different embodiments may describe varying system topology, communication relationships, or hierarchy. For
example, in some embodiments, crowdsourcing is described as a characteristic of the whole system while other embodiments describe online crowdsourcing to be one system as part of an overall group of systems.

As it would be understood, reference to a system means a computer or server configured with software from non-transient memory that is applied to the computer or server to implement the system. The system can include input and output communications sources or data inputs and storage or access to necessary data. Therefore, it would also be understood that it refers to computer-implemented systems and the features and operations described herein are computer implemented and automatically performed unless within the context that user-intervention is described or would normally be understood. If there is no mention of user involvement or intervention, it would generally be understood to be automated.

An example of one embodiment in accordance with principles of the present invention is illustratively shown in FIG. 1. Initially, an example of the overall systems and its components is described in connection with FIG. 1, followed by descriptions of features of embodiments of the components or systems that further clarify discussed aspects, detail different embodiments, or provide more detailed descriptions. There are at times some redundancy that may further assist in clarifying and communicating the relationships of the systems and components. In FIG. 1, system 100 for crowdsourcing of algorithmic forecasting and portfolio generation is shown. System 100 comprises an online crowdsourcing site 101 comprising algorithmic developer's sandbox or development system 103, algorithm selection system 120, incubation system 140, and portfolio management system 160. In this figure, the crowdsourcing component is specifically identified as online crowdsourcing site 101, but in operation other parts of the system will communicate with that site or system and therefore could be considered relationally part of a crowdsourcing site.

FIG. 1 depicts that, in one embodiment, development system 103 may include private workspace 103. Online crowdsourcing site 101 can include one or more servers and associated software that configures the servers to provide the crowdsourcing site. Online crowdsourcing site 101 includes a database of open challenges 107 and also contains other storage such as for storing historical data or other data. The online crowdsourcing site 101 or the development system 103 may further comprise a ranking system that ranks the opening challenges based on corresponding difficulty. It should be understood that when discussing a system, it is referring to a server configured with corresponding software and includes the associated operation on the server (or servers) of the features (e.g., including intervention with users, other computers, and networks).
Online crowdsourcing site 101 is an Internet website or application-based site (e.g., using mobile apps in a private network). Site 101 communicates with external computers or devices 104 over communications network connections including a connection to a wide area communication network. The wide area network and/or software implemented on site 101 provide open electronic access to the public including experts or scientists by way of using their computers or other devices 104 to access and use site 101. Site 101 can include or can have associated security or operational structure implemented such firewalls, load managers, proxy servers, authentication systems, point of sale systems, or others. The security system will allow public access to site 101 but will implement additional access requirements for certain functions and will also protect system 100 from public access to other parts of the system such as algorithm selection system 120.

Development system 103 can include private workspace 105. Development system 103 registers members of the public that want user rights in development system 103. This can include members of the general public that found out about site 101 and would like to participate in the crowdsourcing project. If desired, development system 103 can implement a qualification process but this is generally not necessary because it may detract from the openness of the system. Experts can access the site from their computers 104, to use the site over a public computer network (meaning that there is general access by way of public electronic communications connection to view the content of the site, such as to view challenges and to also register). Individuals can register to become users on site 101 such as by providing some of their identifying information (e.g., login and password) and site 101 (e.g., by way of development 103) registers individuals as users on development system 103. The information is used for authentication, identification, tracking, or other purposes.

Site 101 can include a set of open challenges 107 that were selected to be published and communicated to the general public and registered users. It will be understood that systems generally include transient and non-transient computer memory storage including storage that saves for retrieval data in databases (e.g., open challenges) and software for execution of functionality (as described herein, for example) or storage. The storage can be contained within servers, or implemented in association with servers (such as over a network or cloud connection). The challenges include challenges that define needed individual scientific forecasts for which forecasting algorithms are sought. These are forecasts that do not seek or directly seek that the algorithm forecast a financial outcome. The challenges can include other types of forecasting algorithms such as those that seek a forecast of a financial
outcome. Each challenge will define a forecasting problem and specify related challenge parameters such as desired outcome parameters (e.g., amount of rain) to be predicted.

Site 101 includes algorithmic developer's sandbox or algorithmic development system 103. Development system 103 includes a private development area for registered users to use to develop forecasting algorithms such as private online workspaces 105. Private online workspace 105 includes individual private online workspaces that are available as remotely accessible places for use to each registered user and which include a partitioned integrated development environment. Each partitioned integrated development environment provides a private workspace for a registered expert to work in to the exclusion of other registered users and the public. The development environment provides the necessary resources to perform the algorithm development process to completion. Private workspaces 105 may also be customized to include software, data, or other resources that are customary for the field of science or expertise of that user. For example, a meteorologist may require different resources than an economist. Development site 103 by way of private workspaces 105 and the development environment therein provides registered users with online access to algorithm development software, historic data, forecasting algorithm evaluation tools including one or more tools for performing test trials using the historic data, and a process for submitting one of the user's forecasting algorithms authored in their private online workspace to the system as a contributed forecasting algorithm for inclusion in a forecasting algorithm portfolio.

The online algorithm development software is the tool that the registered expert uses to create and author forecast algorithms for individual open challenges. Different types or forms of algorithm development software exist and are generally available. At a basic level, it is a development tool that an individual can use to build a forecasting model or algorithm as a function of certain input (also selected by the user). The forecasting algorithm or model is the item that is at the core of the overall system and it is a discrete software structure having one or more inputs and one or more outputs and which contains a set of interrelated operations (which use the input) to forecast a predicted value(s) for a future real life event, activity, or value. Generating an accurate forecasting algorithm can be a difficult and complex task which can have great impact not only in the financial field but in other areas as well.

The partitioned workspace is provided with access to use and retrieve historic data, a repository of past data for use as inputs into each forecasting algorithm. The data repository also includes the actual historic real life values for testing purposes. The forecasting
algorithm evaluation tools that are available within the development environment provide software tools for the registered expert to test his or her authored forecasting algorithm (as created in their personal workspace on site 101). The evaluation tools use the historic data in the development environment to run the forecast algorithm and evaluate its performance. The tools can be used to determine accuracy and other performance characteristics. As used herein, the term "accuracy" refers to how close a given result comes to the true value, and as such, accuracy is associated with systematic forecasting errors. Registered experts interact with (and independently control) the evaluation tools to perform testing (test trials) in their private workspace. Overall, site 101 is configured to provide independent freedom to individual experts in their private workspace on site 101 in controlling, creating, testing, and evaluating forecasting algorithms.

Evaluation tools may generate reports from the testing (as controlled and applied by the user), which are stored in the corresponding workspace for that user to review. In some embodiments, development system 103 (or some other system) performs an evaluation of an authored forecasting algorithm in a private workspace without the evaluation being performed or being under the control of the registered expert that authored the forecasting algorithm. The evaluation tools (one or more) can apply historic data (e.g., pertinent historic data that is not available to the expert in their workspace for use in their testing of the algorithm) or other parameters independent of the authoring expert and without providing access to the results of the evaluation report to the authoring expert. Historic data that was not made available for the experts to use in their testing in their workspace is sometimes referred to as out-of-sample data.

In some embodiments, site 101 or some other system can include a component that collects information about individual users activity in their workspace and stores the information external to the private workspaces without providing access or control over the collected information (e.g., expert users cannot modify this information), or stores evaluation reports generated from the collected information.

Private workspace 105 includes a process for submitting one of the user's forecasting algorithms authored in their private online workspace to the system (e.g., the overall system or individual system such as development system 103) as a contributed forecasting algorithm for inclusion in a forecasting algorithm portfolio. Private workspace 105 can include an interactive messaging or signaling option in which the user can select to send one of their authored forecasting algorithms as a contributed forecasting algorithm for further evaluation.
In response to the selection or submission of a contributed forecasting algorithm, algorithm selection system 120 receives (e.g., receives via electronic message or retrieves) the contributed forecasting algorithm for further evaluation. This is performed across submissions by experts of their contributed forecasting algorithms. Algorithm selection system 120 includes one or more servers and associated software that configures the servers to provide the algorithm selection system. On the servers, the algorithm selection system provides a number of features.

In some embodiments, the algorithm selection system monitors user activity inside the private workspaces including monitoring user activity related to test trials performed within the private online workspaces on the contributed forecasting algorithms before the contributed forecasting algorithms were submitted to the system. This can be or include the generation of evaluations such as evaluation reports that are generated independent of the expert and outside of the expert’s private workspace.

The algorithm selection system can include a component that collects information about individual users activity in their workspace and stores the information external to the private workspaces without providing access or control over the collected data (e.g., expert users cannot modify this information), or stores evaluation reports generated from the collected data and it is not available in their private workspace. The component can determine, from the monitored activity, test-related data about test trials performed in the private workspace on the contributed forecasting algorithm including identifying a specific total number of times a trial was actually performed in the private workspace on the contributed algorithm by the registered user. This monitoring feature is also described above in connection with development site 103. In implementation, it relates to both systems and can overlap between or be included as part of both systems in a cooperative sense to provide the desired feature.

Algorithm selection system 120 determines the accuracy and performance of contributed algorithms using hisorical data and evaluation or analytics software tools including determining, from test data about test trials actually performed in the private workspace, a corresponding probability of backtest overfitting associated with individual ones of the contributed forecasting algorithms. Algorithm selection system 120, based on determining the accuracy and performance, identifies a subset of the contributed forecasting algorithms to be candidate forecasting algorithms.

In preferred embodiments, the system, such as one of its parts, algorithm selection system 120 implements a source control (version control) system that tracks iterative versions
of individual forecast algorithms while the forecast algorithms are authored and modified by
users in their private workspace. This is performed independent of control or modification by
the corresponding expert in order to lock down knowledge of the number of versions that
were created and knowledge of testing performed by the expert across versions in their
workspace. The system, such as one of its parts, algorithm selection system 120, determines
test related data about test trials performed in the private workspace in specific association
with corresponding versions of an individual forecasting algorithm, whereby algorithm
selection system 120 determines the specific total number of times each version of the
forecasting algorithm was tested by the user who authored the forecasting algorithm.

If desired, the system, such as one of its parts, algorithm selection system 120,
determines the probability of backtest overfitting using information about version history of
an individual forecast algorithm as determined from the source control system. If desired, the
system can also associate a total number of test trials performed by users in their private
workspace in association with a corresponding version of the authored forecasting algorithm
by that user. The system can determine, from the test data about test trials including a
number of test trials and the association of some of the test trials with different versions of
forecast algorithms, the corresponding probability of backtest overfitting.

Algorithm selection system 120 can include individual financial translators, where,
for example, a financial translator comprises different sets of financial characteristics that are
associated with specific open challenges. Algorithm selection system 120 determines a
financial outcome from at least one of the contributed forecasting algorithms by applying the
set of financial characteristics to at least one of the contributed forecast algorithms. In
implementation, system 100 can be implemented, in some embodiments, without financial
translators. There may be other forms of translators or no translators. The financial
translators are implemented as a set of data or information (knowledge) which requires a set
of forecast values in order to generate financial trading decisions. The system operator can
assess the collection of knowledge and from this set of financial parameters identify
challenges, forecasting algorithms that are needed to be applied to the financial translators so
as to generate profitable financial investment activities (profitable investments or portfolios
over time). The needed forecast can be non-financial and purely scientific or can be
financial, such as forecasts that an economist may be capable of making. In some
embodiments, preexisting knowledge and system are evaluated to determine their reliance on
values for which forecasting algorithms are needed.
Determining trading strategies (e.g., what to buy, when, or how much) can itself require expertise. System 101, if implementing translators, provides the translators as an embodiment of systematic knowledge and expertise known by the implementing company in trading strategies. This incentivizes experts to contribute to the system knowing that they are contributing to a system that embodies an expert trading and investment system that can capitalize on their scientific ability or expertise without the need for the experts to gain such knowledge.

Financial translators or translators (e.g., software) can be used in algorithm selection system 120, incubation system 140, and portfolio management system 160. The translators can be part of the evaluation and analytics in the different systems as part of determining whether a forecasting algorithm is performing accurately, or is performing within certain expected performance levels.

Incubation system 140 receives candidate forecasting algorithm from algorithm selection system 102 and incubates forecasting algorithms for further evaluation. Incubation system 140 includes one or more servers and associates software that configures the servers to provide the incubation system. On the servers, the incubation system performs related features. Incubation system 140 determines an incubation time period for each of the contributed forecasting algorithms. Incubation system 140 determines the period by receiving the particular probability of backtest overfitting for the candidate forecasting algorithms and receives (e.g., predetermined values stored for the system) minimum and maximum ranges for the incubation time period.

In response, incubation system 140 determines a particular incubation period that varies between the maximum and minimum period based primarily on the probability of backtest overfitting associated with that candidate forecasting algorithm, whereby certain candidate forecasting algorithms will have a much shorter incubation period than others. In operation, the system conserves resources and produces accurate forecasts at a higher rate by controlling the length of the incubation period. This is done by monitoring user activity and determining the probability of backtest overfitting using a system structure. This can also avoid potential fraudulent practices (intentional or unintentional) by experts that may compromise the accuracy, efficiency, or integrity of the system.

Incubation system 140 includes one or more sources of live data that are received into the incubation system. incubation system 140 applies live data to the candidate forecasting algorithms for a period of time specified by corresponding incubation time periods for that algorithm. The system can, in operation, operate on a significant scale such as hundreds of
algorithms and large volumes of data such from big data repositories. This can be a massive operational scale.

Incubation system 140 determines accuracy and performance of the candidate forecasting algorithms in response to the application of the live data including by determining the accuracy of output values of the candidate forecast algorithms when compared to actual values that were sought to be forecasted by the candidate algorithms. In response to determining accuracy and performance of the candidate forecasting algorithms, incubation system 140 identifies and stores a subset of the candidate forecasting algorithms as graduate forecasting algorithms as a part of a portfolio of operational forecasting algorithms that are used to forecast values in operational/production systems. In operation, incubation system 140 is implemented to be as close to a production system as possible. Live data, referring to current data such as real time data from various sources, are received by the candidate forecasting algorithms and applied to generate the candidate forecasting algorithm's forecast value or prediction before the actual event or value that is being forecast occurs. The live data precedes the event or value that is being forecasted and the algorithms are operating while in the incubation system to generate these forecasts. The accuracy and performance of algorithms in the incubation are determined from actuals (when received) that are compared to the forecast values (that were determined by the forecasting algorithm before the actuals occurred).

Incubation system 140 can communicate with a portfolio management system. A portfolio management system can include one or more servers, associated software, and data that configures the servers to implement the portfolio management system. On the servers, portfolio management system 160 provides various features. Portfolio management system 160 receives graduate forecasting algorithms from incubation system 140. Portfolio management system 160 stores graduate forecasting algorithms in a portfolio of graduate forecasting algorithms and applies live data to the graduate forecasting algorithms and, in response, receives output values from the graduate forecasting algorithms. Portfolio management system 160 determines directly or indirectly, from individual forecasting algorithms and their corresponding output values, specific financial transaction orders. Portfolio management system 160 transmits the specific financial transaction orders over a network to execute the order. The orders can be sent to an external financial exchange or intermediary for execution in an exchange. In this way, stock order or other financial investment can be fulfilled in an open or private market. The orders can be in a format that is compatible with the receiving exchange, broker, counterparty, or agent. An order when
executed by the external system will involve an exchange of consideration (reflected electronically) such as monetary funds for ownership of stocks, bonds, or other ownership vehicle.

Portfolio management system 160 is a production system that applies forecasting algorithms to real life applications of the forecasts before the actual value or characteristic of the forecasts are known. In the present example, forecasts are applied to financial systems. The system will operate on actual financial investment positions and generate financial investment activity based on the forecast algorithms. The system may in production execute at a significant scale or may be in control (automatic control) of significant financial capital.

Portfolio management system 160, in some embodiments, can include at least two operational modes, wherein in a first mode, portfolio management system 160 processes and applies graduate forecasting algorithms that are defined to have an output that is a financial output. Portfolio management system 160 determines from the financial output the specific financial order. In a second mode, portfolio management system 160, in some embodiments, processes and applies graduate forecasting algorithms that are defined to have an output that is a scientific output, applies a financial translator to the scientific output, and determines from the output of the financial translator a plurality of specific financial orders that when executed generate or modify a portfolio of investments that are based on the scientific output.

Portfolio management system 160 is further configured to evaluate actual performance outcomes for graduate forecasting algorithms against expected or predetermined threshold performance outcomes for corresponding graduate forecast algorithm, and based on the evaluation, determine underperforming graduate forecasting algorithms. In response, portfolio management system 160 removes underperforming graduate forecasting algorithms from the portfolio. Portfolio management system 160 can communicate actual performance outcomes, the removal of graduate algorithms, or a status of graduate forecasting algorithms to other components in the computer-implemented system.

In some embodiments, portfolio management system 160 evaluates performance of graduate forecasting algorithms by performing a simulation after live trading is performed that varies input values and determines variation in performance of the graduate forecasting algorithm portfolio in response to the varied input values, and determines from the variations in performance to which ones of the graduate forecasting algorithms in the portfolio the variations should be attributed. Using this identification, portfolio management system 160 removes underperforming graduate forecasting algorithms from the portfolio.
The management system can gradually reassess capital allocation objectively and, in real-time, gradually learn from previous decisions in a fully automated manner. In some embodiments, online crowdsourcing site 101 (e.g., within development system 103) applies an authorship tag to individual contributed forecasting algorithms and the system maintains the authorship tag in connection with the contributed forecasting algorithms including as part of a use of the contributed forecasting algorithm in the overall system such as in connection with corresponding graduate forecasting algorithms in operational use. The system determines corresponding performance of graduate algorithms and generates an output (e.g., a reward, performance statistics, etc.) in response to the corresponding performance that is communicated to the author identified by the authorship tag. As such, the system can provide an added incentive of providing financial value to individuals who contributed graduate forecasting algorithms. The incentive can be tied to the performance of the graduate algorithm, or the actual financial gains received from the forecast algorithm.

If desired, the system can include a fraud detection system that receives and analyzes contributed forecasting algorithms and determines whether some of the contributed forecasting algorithms demonstrate fraudulent behavior.

FIG. 2 depicts features of one embodiment a development system 200 for developing a forecasting algorithm. Development system 200 includes first database 201 storing hard-to-forecast variables that are presented as challenges to scientists and other experts (or for developing an algorithm), second database 202 storing structured and unstructured data for modeling the hard-to-forecast variables (or for verifying the developed algorithm), analytics engine 206 assessing the degree of success of each algorithm contributed by the scientists and other experts, and report repository 208 storing reports from evaluations of contributed algorithm.

Development system 200 communicates to scientists and other experts a list of open challenges 201 in the form of variables, for which no good forecasting algorithms are currently known. These variables may be directly or indirectly related to a financial instrument or financial or investment vehicle. A financial instrument may be stocks, bonds, options, contract interests, currencies, loans, commodities, or other similar financial interests. Without being limited by theory, as an example, a forecasting algorithm directly related to a financial variable could potentially predict the price of natural gas, while a forecasting algorithm indirectly related to a financial variable would potentially predict the average temperatures for a season, month or the next few weeks. Through the selection system, such as the incubation system, management system, and online crowdsourcing system, the variable
that is indirectly related to finance is translated through a procedure, such as a financial
translator. The translation results in executing investment strategy (based on the forecast
over time).

Development system 200 (which should be understood as development system 103 in
FIG. 1) provides an advanced developing environment which enables scientists and other
researchers to investigate, test and code those highly-valuable forecasting algorithms. One
beneficial outcome is that a body of practical algorithmic knowledge is built through the
collaboration of a large number of research teams that are working independently from each
other, but in a coordinated concerted effort through development system 200. In order for the
scientists and other experts to begin developing the algorithms 204, as described hereinabove,
hard-to-forecast variables which are presented as open challenges 201 to the scientists and
other experts, initiates the development process. For example, an open challenge in database
201 could be the forecasting of Non-Farm Payroll releases by the U.S. Department of Labor.
Development system 200 may suggest a number of variables that may be useful in predicting
future readings of that government release, such as the ADP report for private sector jobs.
Development system 200 may also suggest techniques such as but not limited to the X-13
ARIMA or Fast Fourier Transformation (FFT) methods, which are well-known to the skilled
artisan in order to adjust for seasonality effects, and provide class objects that can be utilized
in the codification of the algorithm 204. If desired, these can be limited to challenges in
order to predict or forecast variations in data values that are not financial outcomes. A series
of historical data resources or repositories 202 (data inputs for forecast algorithms) are used
by the scientists in order to model those hard-to-forecast variables.

In the preferred embodiments, historical data repositories 202 could, for example, be
composed of a structured database, such as but not limited to tables, or unstructured data,
such as collection of newswires, scientific and academic reports and journals or the like.
Hence historical data resources 202 are used by the scientists in order to collect, curate and
query the historical data by running them through the developed algorithms or contributed
algorithms 204 using forecasting analytics engine 206. The forecasting analytics engine
comprises algorithm evaluation and analytics tools for evaluating forecasting algorithms.
Subsequent to running historical data 202 through the contributed algorithms 204, the
analytics engine outputs the analysis and a full set of reports to repository 208. The reports
and outputs are generated for the primary purpose of analyzing the forecasting algorithms
204, and how well and accurate the algorithms are serving their purpose.
In some embodiments, reports that are created in forecasting analytics engine 206 are made available to the corresponding scientists and other experts who authored the algorithm, such that they can use the information in order to further improve their developed algorithms. In other embodiments, some of the reports may be kept private in order to control for bias and the probability of forecast overfitting.

As discussed above, as part of registration with development system 200, scientists are given individual logins that afford them the possibility of remote access to virtual machines. Private workspaces can be provided through a cluster of servers with partitions reserved to each user that are simulated by virtual machines. These partitions can be accessed remotely with information secured by individual password-protected logins. In these partitions, scientists can store their developed algorithms, run their simulations using the historical data 202, and archive their developed reports in repository 208 for evaluating how well the algorithms perform.

As such, analytics engine 206 assesses the degree of success of each developed algorithm. As discussed above, such evaluation tools are accessible in private workspace and under user control and if desired are accessible by the system or system operation to evaluate and test algorithms without the involvement, knowledge or access of the authoring scientist/expert.

In some embodiments, scientists are offered an integrated development environment, access to a plurality of databases, a source-control application (e.g., source-control automatically controlled by the system), and other standard tools needed to perform the algorithm development process.

In addition, analytics engine 206 is also used to assess the robustness and overfitting of the developed forecasting model. In its simplest form, a forecasting model is considered overfit when it generates a greater forecasting power by generating a false positive result which is mainly caused by its noise rather than its signal. Thus, in order to overcome the issue with forecast overfitting, it is preferably desired to have as high signal-noise-ratio as possible wherein the signal-noise-ratio compares the level of a desired signal to the level of background noise. Generally, scientists and other experts may also have a tendency to report only positive outcomes, a phenomenon known as selection bias. Not controlling for the number of trials involved in a particular discovery and testing of an algorithm may lead to over-optimistic performance expectations characterized as described hereinabove due to its higher noise than signal detection. As such, analytics engine 206 can evaluate the probability of forecast overfitting conducted by the scientists (as part of the independent evaluation that
the system or system operator performs external to private workspaces). This is largely performed by evaluating different parameters, such as the number of test trials for evaluation of the probability of overfitting.

In order to minimize and overcome the problem of forecast overfitting, the Deflated Sharpe Ratio ("DSR") may be determined, which corrects for two leading sources of performance overestimation, which are i) the selection bias under multiple testing and ii) non-normally distributed returns. In doing so, DSR helps to separate legitimate empirical findings from otherwise erroneous statistical flukes causing high noise/signal ratio. These concepts of how to control the issue of forecast overfitting are readily known to persons having ordinary skill in the art and are more thoroughly described in greater detail in the publications Bailey, David H. and Lopez de Prado, Marcos, The Deflated Sharpe Ratio: Correcting for Selection Bias, Backtest Overfitting and Non-Normality (July 31, 2014). Journal of Portfolio Management, 40 (5), pp. 94-107. 2014 (40th Anniversary Special Issue). Available at SSRN: [http://ssicom"abstract=246055](http://dx.doi.org/10.2139/ssrn.2326253) Bailey, David H. and Borwein, Jonathan M. and Lopez de Prado, Marcos and Zhu, Qiji Jim, The Probability of Backtest Overfitting (February 27, 2015). Journal of Computational Finance (Risk Journals), 2015, Forthcoming. Available at SSRN: [http://sssm.com/abstract=2326253](http://dx.doi.org/10.2139/ssrn.2326253) and Bailey, David H. and Borwein, Jonathan M. and Lopez de Prado, Marcos and Zhu, Qiji Jim, Pseudo-Mathematics and Financial Charlatanism: The Effects of Backtest Overfitting on Out-of-Sample Performance (April 1, 2014). Notices of the American Mathematical Society, 61(5), May 2014, pp.458-471. Available at SSRN: [http://ssrn.com/abstract=2308659](http://dx.doi.org/10.2139/ssrn.2308659) which disclosures are hereby fully incorporated by reference in their entirety and also included in the Appendix to this application. If desired, the system can apply DSR to deflate or determine a confidence level for each forecasting algorithm.

In addition to the analytics engine, a quantitative due diligence engine can be further added to development system 200 (or as part of another system such as the algorithm selection system). The quantitative due diligence engine carries out a large number of tests in order to ensure that the developed algorithms are consistent and that the forecasts generated are reproducible under similar sets of inputs. Moreover the quantitative due diligence engine also ensures that the developed algorithms are reliable, whereby the algorithm does not crash, or does not fail to perform tasks in a timely manner under standard conditions, and wherein the algorithm does not further behave in a fraudulent way. The development system or the quantitative due diligence engine also processes the trial results in order to determine if the
characteristics of the results indicate a fraudulent implementation (or if it is "bona fide," e.g. that is, the algorithm does not behave in a fraudulent way, e.g., its output complies with Benford's law, results are invariant to the computer's clock, its behavior does not change as the number of runs increases, etc.). A process is performed through an application that receives the results of test trials and processes the results. The process evaluates the distribution or frequency of data results and determines whether the result is consistent with an expected or random distribution (e.g., does the output indicate that the algorithm is genuine). If the process determines based on the evaluation that the algorithm is fraudulent, not bona fide, the system rejects or terminates further evaluation or use of algorithm.

The quantitative due diligence engine can be exclusively accessible to the system or system operator and not the experts in their workspace. The quantitative due diligence engine can provide additional algorithm evaluation tools for evaluating contributed forecasting algorithm. As a matter of process, testing and evaluation is performed by the system, initially, on contributed forecast algorithms. In other words, the system is not configured to evaluate algorithms that are in a work in progress before the expert affirmatively selects an algorithm to submit as a contributed forecasting algorithm to the system.

FIG. 3 depicts features of one embodiment of the development system. As shown in FIG. 3, development system 300 may comprise a platform 305 for developing an algorithm and first database 315 for storing hard-to-forecast variables that are presented as challenges to scientists and other experts, second database 320 storing structured and unstructured data for modeling the hard-to-forecast variables (including historic data), analytics engine 325 for assessing quality of each algorithm contributed by the scientists and other experts, quantitative due diligence engine 330 assessing another quality of each algorithm contributed by the scientists and other experts, and report repository 335 storing each contributed algorithm and assessments of each contributed algorithm.

To develop an algorithm through development system 300, contributors 302, such as scientists and other experts, first communicate with the platform 305 and first database 315 via their computers or mobile devices. Using the hard-to-forecast variables stored in the first database 315 and the tools (e.g., algorithm development software and evaluation tools) provided by platform 305, contributors 302 develop algorithm in their workspace. Contributed algorithms (those selected to be submitted to the system by users from their workspace) are provided to analytics engine 325 (this is for evaluation beyond that which the individual expert may have done). Second database 320 stores structured and unstructured
data and is connected to analytics engine 325 to provide data needed by the contributed algorithm under evaluation. Analytics engine 325 runs the data through the contributed algorithm and stores a series of forecasts. Analytics engine 325 then assesses the quality of each forecast. The quality may include historical accuracy, robustness, parameter stability, overfitting, etc. The assessed forecasts can also be analyzed by quantitative due diligence engine 330 where the assessed forecasts are subject to another quality assessment. Another quality assessment may include assessing the consistency, reliability, and genuineness of the forecast. Assessment reports of the contributed algorithms are generated from the analytics engine 325 and the quantitative due diligence engine 330. The contributed algorithms and assessment reports are stored in report repository 335.

The development of an algorithm through the development system 200, 300 concludes with building a repository 208, 335 of the developed algorithms and assessment reports, and the reports repository 208, 335 is subsequently provided as an input to a selection system.

FIG. 4 depicts features of one embodiment of an algorithm selection system (or ASP system) and steps associated with selection system for selecting a developed algorithm. Algorithm selection system 400 evaluates candidate forecasting algorithms from registered experts and based on the evaluation determination determines which ones of the contributed forecasting algorithms should be candidate forecasting algorithms for additional testing. Selection system 400 comprises forecasting algorithm selection system 404, signal translation system 406, and candidate algorithm library 408. The steps associated with selection system 400 for selecting a developed algorithm can include scanning the contributed algorithms and the reports associated with each contributed forecasting algorithm from the reports repository 402. Algorithm selection system 400 selects from among them a subset of distinct algorithms to be candidate forecasting algorithms. Algorithm selection system 400 translates, if necessary, those forecasts into financial forecasts and/or actual buy/sell recommendations 406, produces candidate forecasting algorithms in database 408, stores candidate forecasting algorithms, and updates the list of open challenges in database 401 based on the selection of contributed algorithms for further evaluation.

In forecasting algorithm selection system 404, it may have a scanning component that scan the contributed forecasting algorithms in the reports repository 402 and that, in scanning, searches for different contributed forecasting algorithms that are mutually
complementary. The scanning component may also determine a subset of the contributed forecasting algorithms that have defined forecast outputs that do not overlap.

Forecasting algorithm selection system 404 or the algorithm selection system 400 may further have a marginal contribution component that determines the marginal forecasting power of a contributed forecasting algorithm. The marginal forecasting power of a contributed forecasting algorithm, in one embodiment, may be the forecasting power that a contributed forecasting algorithm can contribute beyond that of those algorithms already running in live trading (production). The marginal contribution component, in one embodiment, may determine a marginal forecasting power of a contributed forecasting algorithm by comparing the contributed forecasting algorithm to a portfolio of graduate forecasting algorithms (described below) operating in production in live trading, determining, based on the comparison, a marginal value of the contributed forecasting algorithm with respect to accuracy, performance, or output diversity when compared to the graduate forecasting algorithms, and, in response, the algorithm selection system determines which contributed forecasting algorithms should be candidate forecasting algorithms based on at least partly on the marginal value.

Signal translation system 406 (or financial translators) translates the selected algorithms into financial forecasts or actual buy/sell recommendations since the forecasts provided by the selected algorithms, or selected contributed algorithms, may be directly or indirectly related to financial assets (e.g., weather forecasts indirectly related to the price of natural gas). The resulting financial forecasts, or candidate algorithms are then stored in candidate algorithm library 408. As described hereinabove, the algorithm selection system 404 can include: i) a procedure to translate generic forecasts into financial forecasts and actual buy/sell recommendations; ii) a procedure to evaluate the probability that the algorithm is overfit, i.e., that it will not perform out of sample as well as it does in-sample; iii) a procedure to assess the marginal contribution to forecasting power made by an algorithm and iv) a procedure for updating the ranking of open challenges, based on the aforementioned findings.

Because system 100 for crowdsourcing of algorithmic forecasting provides a unified research framework that logs all the trials occurred while developing an algorithm, it is possible to assess to what an extent the forecasting power may be due to the unwanted effects of overfitting. Thus, in selection system 404, for example, it is reviewed how many trials a given scientist has used in order to develop and test a given algorithm with historical data, and based upon the number of trials used by the scientist, a confidence level is subsequently
determined by the analytics engine for the contributed forecasting algorithm. It should be understood that the established confidence level and number of trials used by a given scientist are inversely connected and correlated, such that a high trial number would result in a more greatly deflated confidence level. As used herein, the term "deflated" refers to the lowering of the confidence level determined as described above. If it turns out that a given algorithm is characterized by having a confidence level above a preset threshold level, this specific algorithm would then be qualified as a candidate algorithm in FIG. 4. As a result, advantageously, a lower number of spurious algorithms will ultimately be selected, and therefore, less capital and computation or memory resources will be allocated to superfluous algorithms before they actually reach the production stage. Other techniques can be implemented as alternative approaches or can be combined with this approach.

FIG. 5 depicts features of another embodiment of an algorithm selection system. Algorithm selection system 500 can comprise a forecasting algorithm scanning and selection system 504 (may be configured to perform similar functions as the scanning component described above), forecast translation system 506, forecasting power determination system 508 (similar to the marginal contribution system described above and may be configured to perform similar functions), overfitting evaluation system 510, and a candidate algorithm library 408. Algorithm selection system 500 for selecting a developed algorithm to be a candidate algorithm comprises scanning and selecting the developed algorithms from the reports repository, translating the selected algorithms or forecasts into financial forecasts and/or actual buy/sell recommendations (in component 506), and determining the forecasting power of the financial forecasts (in component 508), evaluating overfitting of the financial forecasts (in component 510), and producing and storing candidate algorithms (in component 512).

FIG. 6 illustrates features of one embodiment of an incubation system. As shown, incubation system 600 is for incubating candidate forecasting algorithms. Incubation system 600 comprises database or data input feed 602, which stores structured and unstructured data (or historical data) for modeling hard-to-forecast variables, candidate algorithm repository 604, "paper" trading environment 606, and performance evaluation system 608. Database or data input feed 602 may provide an input of live data to the candidate forecasting algorithms. The steps for this feature can include simulating 606 the operation of candidate algorithms in a paper trading environment, evaluating performance of the simulated candidate algorithms, and determining and storing graduate algorithms based on the results of the evaluation.
As described hereinbefore, the candidate algorithms that were determined by the selection system are further tested by evaluating the candidate algorithms under conditions that are as realistically close to live trading as possible. As such, before the candidate algorithms are released into the production environment, they are incubated and tested with data resources that comprise live data or real-time data and not by using the historical data resources as explained previously in the development and selection systems and steps. Data such as liquidity costs, which include transaction cost and market impact, are also simulated. This paper trading ability can test the algorithm's integrity in a staging environment and can thereby determine if all the necessary inputs and outputs are available in a timely manner. A person having ordinary skill in the art would appreciate that this experimental setting gathers further real-time confirmatory evidence such that only reliable candidate algorithms are deployed and used in production, whereas unreliable candidate algorithms are discarded before they eventually reach production. Once again this effectuates efficient capital, resource, and time saving. It is important to keep in mind and stress that candidate algorithms can only reach the incubation system 600 only if the achieved confidence level is found to be higher than a predetermined level. Similarly, in incubation system 600, a higher confidence level is used to require the candidate algorithms to be incubated a less amount of time, and an elevated deflated confidence level (e.g., meaning lower confidence level) will be used by the system so as to require the candidate algorithms to be incubated an increased amount of time. Thus, during the evaluation step 608, incubation system 600 determines if the candidate algorithms passes the evaluation. If the candidate algorithms pass the evaluation, evaluation system 608 outputs the passed candidate algorithms, designates the passed candidate algorithms as graduate algorithms 610, and stores the graduate algorithms in a graduate algorithm repository. Further, candidate algorithms 604 are also required to be consistent with minimizing backtest overfitting as previously described.

FIG. 7 shows features of one embodiment of an incubation system. As shown, incubation system 720 can communicate using signaling and/or electronic messaging with management system 730. Using the graduate algorithms from the incubation system that provide investment recommendations, management system 730 determines investment strategies or how the capital should be allocated. Incubation system 720 performs "paper" trading on candidate forecasting algorithms. Incubation system 720 evaluates the performance of candidate forecasting algorithms over time such as by factoring in liquidity costs and performing a divergence assessment by comparing in-sample results to results from out-of-sample data. If for example, the paper trading (simulating "live data" production
operations without the actual real life application of the output) is not performing within an expected range of performance (e.g., accuracy) from actual data values over a minimum period of time, the corresponding candidate forecasting algorithm is terminated from paper trading and removed. The divergence assessment may be performed by a divergence assessment component of incubation system 720. The divergence assessment component, in one embodiment, may be configured to receive candidate forecasting algorithms from the algorithm selection system, evaluate performance information related to the received candidate forecasting algorithms, determine, over time, whether the performance information indicates that individual candidate forecasting algorithms have diverged from in-sample performance values determined prior to the incubation system (or prior to providing the candidate forecasting algorithms to the incubation system), and terminate the incubation period for candidate forecasting algorithms that have diverged from their in-sample performance value by a certain threshold. The divergence assessment component can for example also evaluate the performance of the forecast algorithm (candidate algorithm) in relation to the expected performance determined from backtesting in an earlier system (e.g., the algorithm selection system) and determines when the performance in the incubation is not consistent with the expected performance from backtesting and terminates the paper trading for that algorithm, which can increase resources for additional testing. For example, the expected profit from earlier testing for a period is \( X +/- y \), the divergence analysis will terminate the incubation system’s testing of that algorithm before the incubation period is completed when the performance of the algorithm is below the expected \( X-y \) threshold. The divergence assessment component can also applied in operation during production within the portfolio management system. In conventional systems, an algorithm is terminated from production when a preset threshold that is often times arbitrarily selected and applied to all algorithms is satisfied. In some embodiments of the invention, the management system operates at a more efficient and fine-tuned level by comparing the performance results of the algorithm in production to the algorithms performance in earlier systems (incubation, selection, and/or development system) and terminates the algorithm from production when the performance has diverged from the expected earlier performance (performs more poorly than worst expected performance from earlier analysis).

FIG. 8 illustrates features of one embodiment of a portfolio management system 800. Portfolio management system 800 includes steps associated with management system 800 for managing individual graduate algorithms that were previously incubated and graduated from the incubation system. FIG. 8 also shows connections between incubation system 805 and
management system 800. Management system 800 may comprise survey system 810, decomposition system 815, first capital allocation system 820, second capital allocation system 825, first evaluation system 830 evaluating the performance of the first capital allocation system 820, and second evaluation system 835 evaluating the performance of second capital allocation system 825. The steps may comprise surveying (or collecting) investment recommendations provided by the graduate algorithms (which in context can sometimes refer to the combination of a graduate algorithm and its corresponding financial translator), decomposing the investment recommendations, allocating capital based on decomposed investment recommendations, and evaluating performance of the allocation.

In decomposition system 815, space forecasts are decomposed into state or canonical forecasts. Space forecasts are the result forecasts on measurable financial variables, or in simpler terms, the financial forecasts provided by the graduate algorithms. The decomposition may be performed by procedures such as Principal Components Analysis ("PCA"), Box-Tiao Canonical Decomposition ("BTCD"), Time Series Regime Switch Models ("TSPvS"), and others. The canonical forecasts can be interpreted as representative of the states of hidden "pure bets." For example, a space forecast may be a forecast that indicates that the Dow-Jones index should appreciate by 10% over the next month. This single forecast can be decomposed on a series of canonical forecasts such as equities, U.S. dollar denominated assets, and large capitalization companies. By decomposing every forecast into canonical components, the system can manage and package risk more efficiently, controlling for concentrations. Capital can then be allocated to both types of forecasts, resulting in different portfolios.

Capital allocation may have two modes as depicted in FIGS. 9 and 10. In the first mode 900, optimal capital allocations 906 are made to graduate algorithms 904 based on their relative performance. These optimal capital allocations 906 determine the maximum size of the individual algorithms' 906 positions. Portfolio positions are the result of aggregating the positions of all algorithms 906. In simple terms, it should be understood by a skilled artisan that in the first mode 900, single actual buying/selling recommendations are made, and orders are automatically generated by algorithm trader system 908, which is followed by step 910, wherein the financial backed activities are performed and completed and during step 912, the performance of the graduate algorithms 904 is thus evaluated. To better define this concept, single buying/selling recommendations are executed, which could be, e.g., buying/selling oil or buying/selling copper, or buying/selling gold, etc. As such, in the first mode, a portfolio of
investments are not conducted, as it solely pertains to single buying/selling recommendations based on forecasting graduate algorithm 906.

In the second mode 1000, investment recommendations are translated into forecasts on multiple time horizons. The system's confidence level on those forecasts is a function of the algorithms’ 1004 past performance and a portfolio overlay is run on those forecasts. In simple terms, in the second mode 1000, multiple buying/selling recommendations are made and orders are automatically in real-time generated by algorithm trading system 1010. This is followed by step 1012, wherein financial backed activities are performed and completed. During step 1014, the performance of the graduate algorithms 1004 are evaluated in step 1016. As such, in the second mode, every forecast is decomposed into individual canonical components which affords improved risk management for the individual or organization. If during the performance evaluation in step 1016, some of the graduate algorithms 1004 do not perform as expected, a new portfolio overlay may then be performed in step 1008. Moreover, since the resulting portfolio is not a linear combination of the original recommendations in the second mode (unlike the first mode in which individual algorithms' performance can be directly measured), performance needs to be attributed 1014 back to the graduate strategies. This attribution 1014 is accomplished through a sensitivity analysis, which essentially determines how different the output portfolio would have been if the input forecasts had been slightly different.

Generalized dynamic portfolio optimization problems have no known closed-form solution. These problems are particularly relevant to large asset managers, as the costs from excessive turnover and implementation shortfall may critically erode and influence the profitability of their investment strategies. In addition, and essentially, an investor's own decisions, as well as the entire investment market will directly influence the price of a given share, and as such, there is still an unmet need to implement and incorporate systems and methods that can aid in better determining how to optimize the portfolio, how much to buy/sell, and thereby maximize the portfolio with forecasts that exhibit the ability to predict and pre-calculate an output of prices at multiple horizons. As used herein, the term "horizon" refers to different time-points. Without being limited by theory, for example, some forecasts could refer to an optimized portfolio comprising rice over the next year, stocks over the next 3 months, and soybeans over the next 6 months. Since forecasts involve multiple horizons, the optimal portfolios at each of these horizons would have to be determined, while at the same time, minimizing the transaction costs involved in those portfolio rotations. This financial problem can be reformulated as an integer optimization problem, and such
representation makes it amenable to be solved by quantum computers. Standard computers only evaluate and store feasible solutions sequentially, whereas the goal and ability of quantum computers is to evaluate and store all feasible solutions all at once. Now the principle of integer optimization will be explained in greater detail. The whole purpose of quantum computing technology is to pre-calculate an output, and thereby determining an optimal path through calculating the optimal trading trajectory \( \omega \), which is an \( N \times H \) matrix, wherein \( N \) refers to assets and \( H \) defines horizons. This can be envisioned by establishing a specific portfolio, which can for example be comprised of \( K \) units of capital that is allocated among \( N \) assets over \( X \) amount of months, e.g. horizons. For each horizon, the system would then create partitions or grids of a predetermined value set by the system. For example, if the system has been set to create partitions of incremental increase or decrease of a value of 10, it would then pre-calculate an investment output \( r \) for share numbers that either increase or decrease by the value of 10, such that, if for example 1000, 2000, 3000 contracts of soybean were bought in January, March and May, respectively, the system would then be able to compute and pre-calculate the optimal trading trajectory \( \omega \) at 990, 980, 970 contracts, etc. or 1010, 1020, 1030 contracts, etc. for January. Similar computing would be executed for March, which would in this case be for 1990, 1980, 1970, etc. or 2010, 2020, 2030, etc. contracts and for May, e.g. 2990, 2980, 2970, etc. or 3010, 3020, 3030, etc. In this example, an incremental increase or decrease of a partition of the value 10 is chosen and is shown merely as an example, but a person of ordinary skill in the art would readily know and understand that the partition could advantageously also assume a value of 100, 50, 25, 12, etc. such that it would either decrease or increase incrementally by the aforementioned values. The system would then be able to determine the optimal path of the entire portfolio at multiple horizons from the pre-calculated values, as well as over many different instruments.

The system is configured to apply an additional portfolio management aspect that takes into the account indirect cost of investment activity. For example, it can estimate the expected impact to a stock price in response to trading activity (such as if the system decided to sell a large volume in a stock). It also does this across many algorithms or investment positions. As such, where each algorithm may be capable in specifying the best position for each of a set of different investment (e.g., at a particular time), the system can apply this additional level of processing (having to do with indirect costs) and take into other factors such as investment resources and determine a new optimal/best investment position for the positions that accounts for the quantum issue. The system can implement the process on a quantum computer because the fundamental way that such computers operate appears to be
amenable to this situation. The qubits of the quantum computer can have direct correspondence to the partitioned sections of an individual investment/stock. In one embodiment, the quantum computer is configured with software that together processes graduate forecasting algorithms and indirect cost of associated financial activity and in response determines modifications to financial transaction orders before being transmitted, wherein the portfolio management system modifies financial transaction orders to account for overall profit and loss evaluations over a period of time. In another embodiment, the quantum computer is configured with software that together processes graduate forecasting algorithms by generating a range of parameter values for corresponding financial transaction orders, partitioning the range, associating each partition with a corresponding state of a qubit, evaluating expected combinatorial performance of multiple algorithms overtime using the states of associated qubits, and determining as a result of the evaluating, the parameter value in the partitioned range to be used in the corresponding financial transaction order before the corresponding financial transaction order is transmitted for execution. Consequently the advantage of employing quantum computers can make processing and investment much easier as they provide solutions for such high combinatorial problems.

A skilled artisan would appreciate that from a mathematical perspective, taking the previously defined \( N \times H \) matrix into account, assuming as an example that \( K=6 \) and \( N=3 \), the partitions \((1,2,3)\) and \((3,2,1)\) are treated differently by the system, which means that all distinct permutations of each partition are considered. As used herein the term 'permutation' relates to the act of rearranging, or permuting, all the members of a set into a special sequence or order. Thus if \( K \) assumes the value 6, and \( N \) assumes the value 3, each column of \( \omega \) can adopt one of the following 28 arrays:

\[
\begin{align*}
(1, 1, 4), (1, 4, 1), (4, 1, 1), (1, 2, 3), (1, 3, 2), (2, 1, 3), (3, 1, 2), (2, 3, 1), (3, 2, 1),
(1, 5, 0), (1, 0, 5), (5, 1, 0), (0, 1, 5), (5, 0, 1), (0, 5, 1), (2, 2, 2), (2, 4, 0), (2, 0, 4), (4, 2, 0),
(0, 2, 4), (4, 0, 2), (0, 4, 2), (3, 3, 0), (3, 0, 3), (0, 3, 3), (6, 0, 0), (0, 6, 0), (0, 0, 6).
\end{align*}
\]

Since \( \omega \) has \( H \) columns, there would be \( 28^H \) possible trajectory matrices \( \omega \). For each of these possible trajectories, as explained hereinabove, the system would then compute the investment output \( r \) of the optimal trading trajectory \( \omega \). This procedure is highly computationally intensive. However, quantum computers offer the advantage of simulating multiple matrices for various risk scenarios, such that better and improved investment decisions and strategies can be executed as described and detailed hereinabove. Examples of systems, formulas and application that support features herein are described in greater detail in the following article, which is hereby incorporated herein by reference in its entirety and

As discussed above, the portfolio management system can implement a divergence process that determines whether to terminate certain algorithms. This is performed by determining the performance of individual algorithms and comparing to the algorithm's performance in development, selection, and/or incubation system. For example, in the portfolio management system, there is an expected performance and range of performance based on backtesting; an expectation that in production it will move consistent with previous testing. The system will cut off use of the algorithm if it is inconsistent with the expected performance from backtesting. Rather than continue to run the algorithm until a poor threshold in performance is reached, it gets decommissioned because performance is not possible according to backtesting.

In some embodiments, marginal contribution can be a feature that is implemented by starting with a set, e.g., 100, previously identified forecasting algorithms. The set is running in production and generating actual profit and loss. When a new algorithm is identified by the selection system, the marginal value can be determined by the system by computing the performance of a virtual portfolio that includes the set and in addition that one potential new forecasting algorithm. The performance of that combined set is evaluated and the marginal contribution of the new algorithm is evaluated. The greater contribution is evaluated, the more likely it is added to production (e.g., if above a certain threshold).

FIG. 11 depicts features of one embodiment of a crowdsourcing system 1100 and steps associated with the crowdsourcing system for coordinating the development system, selection system, incubation system, and management system. The crowdsourcing system 1100 essentially provides the means and tools for the coordination of scientists who are algorithmic developers, testing of their contributions, incubation in simulated markets, deployment in real markets, and optimal capital allocation. Thus, in simple terms, the system 1100 integrates the i) algorithmic developer's sandbox, ii) algorithm selection system, iii) incubation system and iv) management of algorithmic strategies system into a coherent and fully automated research & investment cycle.

The steps performed by system 1100 comprise step 1102 where algorithms are developed by scientists and other experts, and the selected developed algorithms 1102 are received and further undergo due diligence and backtests in step 1104. After evaluating for
backtest overfitting and applying a selection process, candidate algorithms 1106 (in a database) are further exercised by evaluating the candidate algorithms 1106 in an incubation process. The candidate algorithms are incubated and tested with live data resources that are obtained in incubation system or step 1108. Graduate algorithms 1110 are obtained and automated single or multiple buying/selling order or recommendations are next conducted (automatically generated) in steps 1112 and 1114 and the performances of the graduated algorithms are then evaluated in step 1116. If during the performance attribution in step 1016, some of the graduate algorithms 610 do not perform as expected, a new portfolio may then be created in step 1112 by, e.g., removing or adding graduate forecasting algorithms.

Thus collectively in sum, the portfolio management system 800 advantageously offers 1) a system that surveys recommendations from the universe of graduate algorithms 904; 2) a system that is able to decompose space forecasts into canonical state forecasts or "pure bets"; 3) a system that computes an investment portfolio as the solution of a strategy capital allocation problem (e.g., first mode 900); 4) a system that computes an investment portfolio as the solution to a dynamic portfolio overlay problem (e.g., second mode 1000); 5) a system that slices orders and determines their aggressiveness in order to conceal the trader's presence; 6) a system that attributes investment performance back to the algorithms that contributed forecasts; 7) a system that evaluates the performance of individual algorithms, so that the system that computes investment portfolios gradually learns from past experience in real-time; 8) building portfolios of algorithmic investment strategies, which can be launched as a fund or can be securitized and 9) building portfolios of canonical state forecasts as "pure bets" rather than the standard portfolios of space forecast.

As discussed above, financial forecasting algorithm is one application of the described technology, and other applications exist. For example, the overall system can be adapted to implement a system that develops and builds a portfolio of forecasting algorithm that are directed to detect fraudulent or criminal behavior. The system can publish open challenges directed to forecasting or predicting the probability of fraudulent or criminal activity. Different challenges can be published. The system can be configured to the private workspace for individuals that want to develop an algorithm to solve one of various challenges directed to such forecasts. The algorithms may identify likely classification of illegal activity based on selected inputs. Overall, the system would operate the same as described herein with respect to financial systems but adapted for forecasting algorithms as a portfolio of algorithms that are specific to determining or predicting fraudulent activity.
FIGS. 12-16 illustrate different data or structures of different data being stored, applied to or used by systems and/or transmitted between the systems in the performance of related features described herein. Referring to FIG. 12, this figure shows one embodiment of structure 1200 of a data transmitted from the development system to the selection system or a data output by the development system. Structure 1200 may have four components, with first component 1205 being the challenge solved by the contributor, second component 1210 being the historical data used to solve the challenge or to verify the developed algorithm, third component 1215 being the algorithm developed or contributed by the contributor, and fourth component 1220 being quality assessment result of the contributed algorithm.

Referring to FIG. 13, this figure shows another embodiment of structure 1300 of the data transmitted from the development system to the selection system or a data output by the development system. Structure 1300 may have only two components, with first component 1305 being the actual algorithm developed or contributed by the contributor and with second component 1310 being quality assessment information that includes the challenge solved, the historical data used for verification and/or assessment, and the result of the assessment.

Referring to FIG. 14, this figure shows one embodiment of structure 1400 of data transmitted from the selection system to the incubation system or a data output by the incubation system. Structure 1400 may have three components, with a first component being translated contributed algorithm 1405, second component 1410 containing information regarding the contributed algorithm, forecasting power of the translated contributed algorithm, and overfitting effect of the translated contributed algorithm, and third component 1415 for updating the list of challenges.

Referring to FIG. 15, this figure shows one embodiment of structure 1500 of data transmitted from the incubation system to the management system or a data output by the management system. Structure 1500 may have two components, with a first component 1505 being the candidate algorithm and a second component 1510 containing information regarding the paper trading performance, liquidity cost performance, and out-of-sample performance.

Referring to FIG. 16, this figures shows one embodiment of structure 1600 of a data output by the management system. With respect to the data being generated by the first mode, structure 1605 of that data may have three components, with first component 1610 being decomposed canonical or state forecasts, a second component containing an investment strategy, and a third component being an investment portfolio containing investments based on the investment strategy. With respect to the data being generated by the second mode,
structure 1660 of that data may have three components, with first component 1665 being decomposed canonical or state forecasts, second component 1670 containing another investment strategy different from the investment strategy employed in the first mode, and third component 1675 being an investment portfolio containing investments based on the another investment strategy.

FIGS. 17-28 provide additional detailed descriptions for some embodiments of the present invention related to implementing features of the embodiments.

Referring to FIG. 17, this figure depicts one embodiment of core data management system 1700. System 1700 may comprise a plurality of vendor data sources 1705, 1710, 1715, a core data processor 1720, and core data storage 1725, and cloud-based storage 1730. The plurality of vendor sources may comprise exchanges 1705, where tradable securities, commodities, foreign exchange, futures, and options contracts are sold and bought such as NASDAQ, NYSE, BATS, Direct Edge, Euronext, ASX, and/or the like, and financial data vendors 1710, such as Reuters and other vendors. The core data may be historical or real-time data, and data may be financial or non-financial data. Historical data or historical time-series may be downloaded every day, and the system 1700 may alert any restatement to highlight potential impact on algorithms behavior. Real-time data, such as real-time trade or level-1 data, may be used for risk and paper trading. Non-financial data may include scientific data or data used and/or produced by an expert in a field other than finance or business.

Core data provided by exchanges 1705 may be supplied to and consumed by a system 1715 consuming market data. The system 1715 may be created through software development kits (such as Bloomberg API) developed Bloomberg. The data consumed by system 1715 and the data from the vendors 1710 are fed to core data processor 1720. Core data processor 1720 processes all the received data into formats usable in the development system or the system for crowdsourcing of algorithmic forecasting. The processed data is then stored in core data storage 1725 and/or uploaded to cloud 1730 for online access. Stored data or old data may be used to recreate past results, and data in storage 1725 (or local servers) and cloud 1730 (or remote servers) may be used for parallel processing (described below).

Referring to FIG. 18, this figure depicts one embodiment of a backtesting environment (e.g., for development system) 1800. The backtesting environment 1800 is an automated environment for backtesting selected algorithms from or in the development system. Algorithms may be coded in a computing environment and/or programming
language such as Python, Matlab, R, Eviews, C++, or in any other environments and/or programming languages used by scientists or other experts during the ordinary course of their professional practice. The coded forecasting algorithms and the core data from core data storage are provided to automation engine to run backtest or test the coded algorithms with the core data. The automation engine then generates backtest results and the results are available in the internet (discussed below). The backtest results may also be compared with backtest results produced previously. The system for crowdsourcing of algorithmic forecasting or the backtesting environment may keep track of backtesting results for all versions of the coded algorithms, and monitor and alert any potential issues.

Referring to FIG. 19, this figure depicts one embodiment of paper trading or incubation system. Forecasting algorithms coded in different computing environments and/or programming languages are employed to trade financial instruments, and the trades can be performed at various frequencies such as intraday, daily, etc. Coded forecasting algorithms have access to core data and core data storage such as by having access to real-time market data. Targets produced by coded forecasting algorithms are processed by risk manager in real time. Targets can be individual messages or signals that are processed by risk manager. Risk manager processes the targets and determines corresponding investment actions (e.g., buy, sell, quantity, type of order, duration, etc.) related to the subject of target messages (e.g., oil). The execution quality of the coded forecasting algorithms is in line with live trading. Real time market data is used for trading simulation through a market simulator and various algorithms may be used to simulate market impact. Risk manager may also perform risk checks and limits validations (or other risk or compliance evaluation) for investment activity performed by risk manager. The risk manager can reject the order in case limits exceed (compared to limits previously stored or set by the user of the paper trading system), be aware of corporate actions, trade based on notional targets, allow manual approval limits, and check if the order is compliance with government requirements or regulations etc. Based on the actions executed by the risk manager, performance of the coded forecasting algorithms can be determined. Paper trading system may further be designed with failover and DR abilities and may also monitor and alert any potential issues. The critical components of paper trading system or coded algorithms may be coded in C++.

A monitoring and alerting system may be implemented in the backtesting environment, the paper trading system, or any other system within the system for
crowdsourcing of algorithmic forecasting. The monitoring and alerting system may monitor the system or environment (hat wants to be monitored and send various alerts or alert notifications if there are any issues. Processes and logs within each monitored system and environments are monitored for errors and warnings. Market data is also monitored keeping track of historical update frequency. Monitoring may further include expecting backiests to finish by a certain time every day and in case of issues alerts are sent.

The monitoring and alerting system may send alerts or alert notifications in various forms such as emails, text messages, and automated phone calls. The alert notifications may be sent to the contributors, the entity providing the system for crowdsourcing of algorithmic forecasting, support team, or any others to whom the alert notifications are important. In an alert notification, it may include well defined support, history of alerts raised, and available actions.

The test system may be treated as part of a production system. FIGS. 20-22 depict various embodiments of the alert notifications and alert management tools for managing the alert notifications. FIG. 20 shows an example of an email alert notification 2000, FIG. 21 shows an example of an alert management tool maintained by a third party vendor 2100, and FIG. 22 shows an example of an alert management tool on intra web 2200.

Referring to FIG. 23, this figure depicts one embodiment of deployment process system 2300. Coded algorithms or new coded algorithms 2305 may be deployed through an automated process. The deployment may be carried out through a one-click deployment using intra web. There may be controls or authentication tools in place to initiate the deployment process or to operate the deployment tool 2320 for initiating the deployment process. The deployment tool 2320 may be integrated with a source control (GIT) 2310, and it would not be possible to deploy local builds and uncommitted software. The deployment tool 2320 takes codes or algorithms from the source control (GIT) and deploys on target machine. The process is configured to create a label or tag for each algorithm that is automatically generated and assigned to individual algorithms. The process assigns a unique identifier relative to other algorithms that are on the system. The source control (GIT) 2310 implements the system source control feature that maintains source control over algorithm development without control by individual users who created them. FIG. 24 depicts a screen shot of the deployment tool screen from intra web. For example, this can be for the development system.

Referring to FIG. 25, this figure depicts one embodiment of a parallel processing system 2500 for implementing the various systems described herein. Algorithms are
uploaded to the cloud and the parallel processing system 2500 has the ability to run on multiple cloud solutions at the same time. Both core data and market data available to the parallel processing system 2500. The parallel processing system 2500 uses proprietary framework to break jobs into smaller jobs and to manage status of each job and resubmit. The parallel processing system 2500 may have tools to monitor status and ability to resubmit individual failed jobs. The parallel processing system 2500 can access high number of cores based on need. Various algorithms may be used for parallel processing such as per symbol, per day or year based on task complexity and hardware requirements. The parallel processing system 2500 can combine results and upload back to the cloud. The results can be combined on incremental basis or full as need basis. The parallel processing system 2500 supports both Windows and Linux-based computers.

Referring to FIG. 26, this figure depicts one embodiment of a performance evaluation system 2600. The performance evaluation system 2600 comprises a performance engine 2620 that evaluates backtest results 2605 and paper trading results 2610 and that determines backtest performance 2625 and paper trading performance 2630. For backtest performance 2625, the performance may be calculated based on close price and fixed price transaction cost applied. For the paper trading performance 2630, the performance may be calculated based on trades and actual fill prices used from the risk manager. The performance evaluation system 2600 or the performance engine 2620 may compare the performances of backtest and paper trading to understand slippage. The performance evaluation system 2600 may keep track of historic performances, and versions and various other analytics. All the performance and comparison information may be made available on the intra web, and FIG. 27 is a screenshot of the performance results generated by the performance evaluation system 2600 or the performance engine 2620 on the intra web.

The intra web may be an internal website of the entity that provides the system for crowdsourcing of algorithmic forecasting. The intra web provides information related to algorithms and their performances, portfolio level performances, backtest and paper trading results, performances, and comparisons, real-time paper trading with live orders and trades, algorithm level limits, system monitoring and alerts, system deployment, deployment process, analytics reports for work in progress, and user level permissions and controls implements. The intra web also provides features and tools that may adjust different parameters and enable further analysis of all the above information. FIG. 28 is a screenshot of one embodiment of the intra web.
Embodiments of the present invention can take a radically different approach than known systems. Rather than selecting algorithms with a high forecasting power, a subset of algorithms that are mutually complementary among the most profitable can be selected instead. Each of the algorithms forecasts variables that can explain distinct portions of market volatility, minimizing their overlap. The outcome is a portfolio of diversified algorithms, wherein each algorithm makes a significant contribution to the overall portfolio. From the embodiments various advantages can be attained for example:

- Identifying what problems are worth solving, so that researchers do not spend time working on problems that have already been cracked, or problems of little investment significance;
- Allowing a large community of researchers to contribute algorithmic work without having to join a financial firm;
- The forecasting algorithms do not need to be directly associated with financial variables;
- Algorithmic contributions do not involve trading signals or trading rules. This means that researchers on any field can contribute to this endeavor regardless of their background or familiarity with financial concepts;
- All the data, computers and specialized software needed to perform the work are provided by the system, so that the researchers can focus their efforts in solving a very specific problem: Modeling a hard-to-forecast variable from our list of outstanding problems;
- The analytics engine assesses the degree of success of each algorithm. This information will guide the researcher’s future efforts;
- The system builds a library of contributed algorithms with links to the history of studies performed. The library of forecasting algorithms can then later be analyzed to search for profitable investment strategies. This is critical information that is needed to control for the probability of forecast overfitting (a distinctive feature of our approach). A model is considered overfit when its greater complexity generates greater forecasting power in-sample (“IS”), however this comes as a result of explaining noise rather than signal. The implication is that the forecasting power out-of-sample (“OOS”) will be much lower than what was attained IS. Thus the system can evaluate whether the performance of IS departs from the performance OOS, net of transaction costs;
• Discarding unreliable candidate algorithms before they reach the production environment, thus saving capital and time;
• Offering a backup analysis to the algorithm selection process by incorporating OOS evidence to the algorithm selection process;
• Tracking the skills of various developers. Rankings of researchers are generated and graded certificates are issued of quantitative knowledge. Junior researchers could potentially use those certificates when applying for a job in a financial institution;
• Adjusting for the performance inflation that results from running multiple trials before identifying a "candidate algorithm". This has been identified as a critical flaw in the scientific method (refer to www.alltrials.net). But because the algorithm selection process can provide a unified research framework that logs all trials associated with a forecast, it makes it possible to assess to what extent the forecasting power is due to overfitting. As a result, a lower number of spurious algorithms will be selected, and less capital will be allocated to superfluous algorithms;
• Constructing portfolios of forecasting algorithms that are resilient to changes in market regimes also known as "structural breaks" and
• Identifying what challenges researchers struggle with, thus guiding the work of researchers. The more challenging a variable is to forecast, the greater the economic value. Algorithm selection process can recognize this situation thanks to its logging of all trials.

Some of the features of embodiments of the present invention that by themselves or in combination aid in achieving the aforementioned advantages include:

• A system that conveys the list of open challenges, a ranking of hard-to-forecast variables, e.g. variables that are expected to generate significant profits if properly modeled, whether they are directly or indirectly related to financial instruments.
• A system that collects, curates and queries Big Data sets (e.g., historical data set related to core processing).
• A system that provides developers with a working environment where they can write and test their code with minimum effort.
A system that simulates the forecasts that the algorithm would have produced in history, generating proprietary analytical reports that are communicated to the researchers, so that they can improve the forecasting power of their algorithms.

- A system that evaluates the probability of forecast overfitting.
- A system that monitors possible fraudulent or inconsistent behavior.

Now referring to FIG. 29, exemplary hardware and software components of computer 2900 employed in the embodiments of the present invention are shown and will be described in greater detail. Computer 2900 may be implemented by combination of hardware and software components. Although FIG. 29 illustrates only one computer, the embodiments of the present invention may employ additional computers 2900 to perform their functions if necessary. FIG. 29 depicts one embodiment of computer 2900 that comprises a processor 2902, a main memory 2904, a display interface 2906, display 2908, a second memory 2910 including a hard disk drive 2912, a removable storage drive 2914, interface 2916, and/or removable storage units 2918, 2920, a communications interface 2922 providing carrier signals 2924, a communications path 2926, and/or a communication infrastructure 2928.

In another embodiment, computer 2900, such as a server, may not include a display, at least not just for that server, and may have transient and non-transient memory such as RAM, ROM, and hard drive, but may not have removable storage. Other configuration of a server may also be contemplated.

Processor or processing circuitry 2902 is operative to control the operations and performance of computer 2900. For example, processor 2902 can be used to run operating system applications, firmware applications, or other applications used to communicate with users, online crowdsourcing site, algorithm selection system, incubation system, management system, and multiple computers. Processor 2902 is connected to communication infrastructure 292.8, and via communication infrastructure 2928, processor 2902 can retrieve and store data in the main memory 2904 and/or secondary memory 2910, drive display 2908 and process inputs received from display 2908 (if it is a touch screen) via display interface 2906, and communicate with other, e.g., transmit and receive data from and to, other computers.

The display interface 2906 may be display driver circuitry, circuitry for driving display drivers, circuitry that forwards graphics, texts, and other data from communication infrastructure 2928 for display on display 2908, or any combination thereof. The circuitry
can be operative to display content, e.g., application screens for applications implemented on the computer 2900, information regarding ongoing communications operations, information regarding incoming communications requests, information regarding outgoing communications requests, or device operation screens under the direction of the processor 2902. Alternatively, the circuitry can be operative to provide instructions to a remote display.

Main memory- 2904 may include cache memory, semi-permanent memory such as random access memory ("RAM"), and/or one or more types of memory used for temporarily storing data. Preferably, main memory 2904 is RAM. In some embodiments, main memory 2904 can also be used to operate and store the data from the system for crowdsourcing of algorithmic forecasting, the online crowdsourcing site, the algorithm selection system, the incubation system, the management system, live environment, and/or second memory 2910.

Secondary memory 2910 may include, for example, hard disk drive 2912, removable storage drive 2914, and interface 2916. Hard disk drive 2912 and removable storage drive 2914 may include one or more tangible computer storage devices including one or more tangible computer storage devices including a hard-drive, solid state drive, flash memory, permanent memory such as ROM, magnetic, optical, semiconductor, or any other suitable type of storage component, or any combination thereof. Second memory 2910 can store, for example, data for implementing functions on the computer 2900, data and algorithms produced by the systems, authentication information such as libraries of data associated with authorized users, evaluation and test data and results, wireless connection data that can enable computer 2900 to establish a wireless connection, and any other suitable data or any combination thereof. The instructions for implementing the functions of the embodiments of the present invention may, as non-limiting examples, comprise non transient software and/or scripts stored in the computer-readable media 2910.

The removable storage drive 2914 reads from and writes to a removable storage unit 2918 in a well-known manner. Removable storage unit 2918 may be read by and written to removable storage drive 2914. As will be appreciated by the skilled artisan, the removable storage unit 2918 includes a computer usable storage medium having stored therein computer software and/or data. Removable storage is option is not typically include as part of a server.

In alternative embodiments, secondary memory 2910 may include other similar devices for allowing computer programs or other instructions to be loaded into computer 2900. Such devices may include for example a removable storage unit 2920 and interface 2916. Examples of such may include a program cartridge and cartridge interface, a removable memory chip (such as an erasable programmable read only memory ("EPROM"),
or programmable read only memory ("PROM") and associated socket, and other removable storage units 2920 and interfaces 2916, which allow software and data to be transferred from the removable storage unit 2920 to computer 2900.

The communications interface 2922 allows software and data to be transferred between computers, systems, and external devices. Examples of communications interface 2922 may include a modem, a network interface such as an Ethernet card, or a communications port, software and data transferred via communications interface 2922 are in the form of signals 2924, which may be electronic, electromagnetic, optical or other signals capable of being received by communications interface 2922. These signals 2924 are provided to communications interface 2922 via a communications path (e.g., channel) 2926. This path 2926 carries signals 2924 and may be implemented using wire or cable, fiber optics, a telephone line, a cellular link, a radio frequency ("RF") link and/or other communications channels. As used herein, the terms "computer program medium" and "computer usable medium" generally refer to media such as transient or non-transient memory including for example removable storage drive 2914 and hard disk installed in hard disk drive 2912. These computer program products provide software to the computer 2900.

The communication infrastructure 2928 may be a communications-bus, cross-over bar, a network, or other suitable communications circuitry operative to connect to a network and to transmit communications between processor 2902, main memory 2904, display interface 2906, second memory 2910, and communications interface, and between computer 2900 or a system and other computers or systems. When the communication infrastructure 2928 is a communications circuitry operative to connect to a network, the connection may be established by a suitable communications protocol. The connection may also be established by using wires such as an optical fiber or Ethernet cable.

Computer programs also referred to as software, software application, or computer control logic are stored in main memory 2904 and/or secondary memory 2910. Computer programs may also be received via communications interface 2922. Such computer programs, when executed, enable or configure the computer 2.900 to perform the features of the present invention as discussed herein. In particular, the computer programs, when executed, enable the processor 2902 to perform the features of the present invention. Accordingly, such computer programs represent controllers of the computer 2900.

In an embodiment in which the invention is implemented using software, the software may be stored in a computer program product and loaded into computer 2900 using removable storage drive 2914, hard drive 2912, or communications interface 2922. The
control logic, which is the software when executed by the processor 2902 causes the processor 2902 to perform the feature of the invention as described herein.

In another embodiment, the invention is implemented primarily in hardware using for example hardware components, such as application specific integrated circuits ("ASICs"). Implementation of the hardware state machine so as to perform the functions described herein will be apparent to persons skilled in the relevant arts.

In yet another embodiment, the embodiments of the instant invention are implemented using a combination of both hardware and software.

Computer 2900 may also include input peripherals for use by users to interact with and input information into computer 2900. Users such as experts or scientists can use a computer or computer-based devices such as their PC to access and interact with the relevant systems described herein such as using a browser or other software application running on the computer or computer-based device to use the online crowdsourcing site and the development system. Computer 2900 can also be a database server for storing and maintaining a database. It is understood that it can contain a plurality of databases in the memory (in main memory 2904, in secondary memory 2910, or both). In some embodiments, a server can comprise at least one computer acting as a server as would be known in the art. The server(s) can be a plurality of the above mentioned computer or electronic components and devices operating as a virtual server, or a larger server operating as a virtual server which may be a virtual machine, as would be known to those of ordinary skill in the art. Such possible arrangements of computer(s), distributed resources, and virtual machines can be referred to as a server or server system. Cloud computing, for example, is also contemplated. As such the overall system or individual systems such as the selection system or incubation system can be implemented on a separate servers, same server, or different types of computers. Each system or combinations of systems can also be implemented on a virtual server that may be part of a server system that provides one or more virtual servers. In a preferred version, the portfolio management system is a separate system relative to the development system, selection system, and incubation system. This can maintain its security by way of features of additional security such as firewalls.

The present systems, methods, or related inventions also relate to a non-transient computer readable medium configured to carry out any one of the methods disclosed herein. The application can be a set of instructions readable by a processor and stored on the non-transient computer readable medium. Such medium may be permanent or semi-permanent
memory such as hard drive, floppy drive, optical disk, flash memory, ROM, EPROM, EEPROM, etc., as would be known to those of ordinary skill in the art.

Users such as experts or scientists can use a computer or computer-based devices such as their PC to access and interact with the relevant systems described herein such as using a browser or other software application running on the computer or computer-based device to use the online crowdsourcing site and the development system.

It should be understood by those of ordinary skill in the art of computers and telecommunications that the communications illustratively described herein typically include forming messages, packets, or other electronic signals that carry data, commands, or signals, to recipients for storage, processing, and interaction. It should also be understood that such information is received and stored, such as in a database, using electronic fields and data stored in those fields.

In some embodiments, the system is implemented to monitor and record all activity within each private workspace associated with the user of that workspace in creating, modifying, and testing a particular forecast algorithm (including through each incremental version of the algorithm). The collected data is used by the system to evaluate an expert's contributed forecast algorithm that is associated with the collected data. The collected data can be data that includes the number of test trials, the type of data used for trials, diversity in the data, number of versions of the algorithm, data that characterizes the correlation of test trials, different parameters used for inputs, time periods selected for testing, and results or reports of testing performed by the expert in his or workspace (Including for example the results of analytical or evaluation tools that were applied to the algorithm or the results of testing). The total number of test trials and the correlation value related to the diversity of testing can be one set of data, by itself, for example. The system can be configured to collect data that can accurately evaluate a preferred confidence level in the contributed algorithm based on information generated from the development and testing of the algorithm before the algorithm was submitted as a contributed forecasting algorithm. For example, necessary data for determining BPQ can be collected and used for the evaluation. The system can perform the collection, storage, and processing (e.g., for analytics) independent of the control of the corresponding user in the workspace and as generally understood herein is performed automatically. It would be understood that preferably data (e.g., user activity in the workspace) that is unrelated to the objective such as formatting related activity or mouse locations or other trivial or unrelated data are not necessarily collected and stored. Examples of systems, formulas, or applications that support features herein are in the following articles.

In one respect, in some embodiments, evaluation information developed or generated in the system is progressively used in subsequent parts of the system. The application of the evaluation information, as shown, by the examples herein, provides an improved system that can generate better performance with fewer resources.

Another point of clarification relates to known existing systems in certain fields. In known financial systems, financial companies deploy portfolio management systems that
automatically trade equities such as financial investments (buy/sell orders). These automated systems that detect or receive input and automatically trade are used in the financial world, but as discussed above, they have certain deficiencies and improvement of these known equipment is achieved by features of the present invention. For example, the automated trading system incorporates the improved techniques for using forecasting algorithms.

It will readily be understood by one having ordinary skill in the relevant art that the present invention has broad utility and application. Other embodiments may be discussed for additional illustrative purposes in providing a full and enabling disclosure of the present invention. Moreover many embodiments such as adaptations, variations, modifications, and equivalent arrangements will be implicitly disclosed by the embodiments described herein and fall within the scope of the present invention.

Accordingly, while the embodiments of the present invention are described herein in detail in relation to one or more embodiments, it is to be understood that this disclosure is illustrative and exemplary of the present invention, and is made merely for the purposes of providing a full and enabling disclosure of the present invention. The detailed disclosure herein of one or more embodiments is not intended nor is to be construed to limit the scope of patent protection afforded by the present invention, which scope is to be defined by the claims and the equivalents thereof. It is not intended that the scope of patent protection afforded the present invention be defined by reading into any claim a limitation found herein that does not explicitly appear in the claim itself.

Thus for example any sequence(s) and/or temporal order of steps of various processes or methods (or sequence of system connections or operation) that are described herein are illustrative and should not be interpreted as being restrictive. Accordingly, it should be understood that although steps of various processes or methods (or connections or sequence of operations) may be shown and described as being in a sequence or temporal order, but they are not necessary limited to being carried out in any particular sequence or order. For example, the steps in such processes or methods generally may be carried out in various different sequences and orders, while still falling within the scope of the present invention. In addition systems or features described herein are understood to include variations in which features are removed, reordered, or combined in a different way.

Additionally it is important to note that each term used herein refers to that which the ordinary artisan would understand such term to mean based on the contextual use of such term herein. It would be understood that terms that have component modifiers are intended
to communicate the modifier as a qualifier characterizing the element, step, system, or component under discussion.

Although the present invention has been described and illustrated herein with referred to preferred embodiments, it will be apparent to those of ordinary skill in the art that other embodiments may perform similar functions and/or achieve like results. Thus it should be understood that various features and aspects of the disclosed embodiments can be combined with or substituted for one another in order to form varying modes of the disclosed invention.


A MIXTURE OF GAUSSIANS APPROACH TO MATHEMATICAL PORTFOLIO OVERSIGHT: THE EF3M ALGORITHM

Marcos Lopez de Prado

Head of Quantitative Trading - Hess Energy Trading Company and
Research Affiliate - Lawrence Berkeley National Laboratory
lopezdeprado@lbl.gov

Matthew D. Foreman
Professor of Mathematics - University of California, Irvine
mforeman@math.uci.edu

First version: August 2011
This version: June 2013

We wish to thank Robert Almgren (Quantitative Brokers, NYU), Marco Avellaneda (NYU), David H. Bailey (Lawrence Berkeley National Laboratory), Sid Browne (Guggenheim Partners), John Campbell (Harvard University), Peter Carr (Morgan Stanley, NYU), David Easley (Cornell University), Ross Garon (SAC Capital), Robert Jarrow (Cornell University), Andrew Karolyi (Cornell University), David Leinweber (Lawrence Berkeley National Laboratory), Attilio Meucci (Kepos Capital, NYU), Maureen O'Hara (Cornell University), Riccardo Rebonato (PIMCO, University of Oxford), Luis Viceira (HBS), and participants at Morgan Stanley's Monthly Quant Seminar for their helpful comments.
A MIXTURE OF GAUSSIANS APPROACH TO MATHEMATICAL PORTFOLIO OVERSIGHT: THE EF3M ALGORITHM

ABSTRACT

An analogue can be made between: (a) the slow pace at which species adapt to an environment, which often results in the emergence of a new distinct species out of a once homogeneous genetic pool, and (b) the slow changes that take place over time within a fund, mutating its investment style. A fund's track record provides a sort of genetic marker, which we can use to identify mutations. This has motivated our use of a biometric procedure to detect the emergence of a new investment style within a fund's track record. In doing so, we answer the question: "What is the probability that a particular PM's performance is departing from the reference distribution used to allocate her capital?"

The EF3M algorithm, inspired by evolutionary biology, may help detect early stages of an evolutionary divergence in an investment style, and trigger a decision to review a fund's capital allocation.

Keywords: Skewness, Kurtosis, Mixture of Gaussians, Moment Matching, Maximum Likelihood, EM algorithm.

JEL Classifications: C13, C15, C16, C44.
1. INTRODUCTION
Shortly after the publication of Darwin (1859), several statistical methods were devised to find empirical evidence supporting the Theory of Evolution. To that purpose, Francis Galton set the foundations of "regression analysis". With the help of other students of Darwin's theory, Galton established the journal "Biometrika", where Karl Pearson, Ronald Fisher, William ("Student") Gosset, Francis Edgeworth, David Cox and other "founding fathers" of modern statistical analysis published their work. "Mixture distributions" were originally devised as a tool to demonstrate "Evolutionary divergence". Pearson (1894) noted that the breadth of Naples crabs' forehead could be accurately modeled by mixing two Gaussian distributions, which would indicate that a new species of crab was emerging from, and becoming distinctly different to, a once homogeneous species. Many statistical methods were inspired by "Evolutionary" ideas, and remembering that connection can help us see apparently unrelated matters in a new light.

Ever since Pearson's work, mixtures have been applied to problems as varied as modeling complex financial risks (Alexander (2001, 2004), Tashman and Frey (2008)), fitting the implied volatility surface (Rebonato and Cardoso (2004)), stochastic processes (Brigo, Mercurio and Sartorelli (2002)), handwriting recognition (Bishop (2006)), housing prices, topics in a document, speech recognition, and many examples of clustering or unsupervised learning procedures in the fields of Biology, Medicine, Psychology, Geology, etc. (Makov, Smith and Titterington (1985)). In this paper, we will apply mixtures to the problem of "portfolio oversight". In the financial application we present in this paper, the connection between mixtures and "Evolution" is more evident than in other instances cited above.

Mixture distributions are derived as convex combinations of other distribution functions. They are non-Normal, because their observations are not drawn simultaneously from all distributions, but from one distribution at a time. For example, in the case of a mixture of two Gaussians, each observation has a probability \( p \) of being drawn from the first distribution, and a probability \( 1-p \) of coming from the second distribution (the observation cannot be drawn from both). Mixtures of Gaussians are extremely flexible non-Normal distributions, and even the mixture of two Gaussians covers an impressive subspace of moments' combinations (Bailey and Lopez de Prado (2011)).

Pearson (1894) was the first to propose the method of Moment Matching (MM), which consists in finding the parameters of a mixture of two Gaussians to match the first five moments. A gigantic work of algebraic manipulation allowed him to represent the solution in terms of a 9th degree (nonic) polynomial in five variables. This results in a variety of roots, and the issue of making the correct choice arises. Undoubtedly he must have thought that this problem was worth the weeks and months that deriving this complex system must have consumed. Computational limitations of those days made this approach intractable. More recently, Craigmille and Titterington (1997), Wang (2001) and McWilliam and Loh (2008) have revived interest in MM algorithms.

While working at NASA, Cohen (1967) devised a rather convoluted way to circumvent Pearson's nonic equation by initially assuming equality of the variances. A cubic equation could then be solved for a unique negative root, which could then be fed into an iterative process. With the help of a conditional maximum likelihood procedure, he attempted to eliminate the effect of
sampling errors resulting from the direct use of the fifth moment. Similarly, Day (1969) published in "Biometrika" a procedure for estimating the components of a mixture of two Normal distributions through Maximum Likelihood (ML).

Since Dempster, Laird and Rubin (1977) introduced the Expectation-Maximization (EM) algorithm, this has become the preferred general approach to fitting a mixture distribution (see Hamilton (1994) for an excellent reference). The EM algorithm searches for the parameter estimates that maximize the posterior conditional distribution function over the entire sample. Higher moments, for which the researcher may have no theoretical interpretation or confidence, are impacting the parameter estimates. For example, financial markets theories typically have no interpretation for moments beyond the fourth order. It seems reasonable to focus, as the algorithm we present does, primarily on the first four moments for which one has higher confidence and a theoretical interpretation.

Although the MM, ML and EM approaches are extremely valuable, a number of reasons have motivated our proposal for a new answer to this century-long question. First, researchers usually have greater confidence in the first three or four moments of the distribution than on higher moments or the overall sample (Bailey and Lopez de Prado (2011)). An exact match of the fourth and particularly fifth moment is not always desirable due to their significant sampling errors, which are a function of those moments' magnitude. Biasing our estimates in order to accommodate even higher (and therefore noisier) moments, as the ML and EM-based algorithms do, is far from ideal. We would rather have a distribution of parameter estimates we can trust, than a unique solution that is derived from unreliable higher moments. Second, in the Quantitative Finance literature, it is the first four moments that play a key role in the theoretical modeling of risk and portfolio optimization (see Hwank and Satchell (1999), Jurczenko and Maillet (2002), Favre and Galeano (2002), to mention only a few), not the fifth and beyond. Third, in the context of risk simulation, often we face the problem of modeling a distribution that exactly matches the empirically observed first three or four moments (e.g., Brooks and Kat (2002), Lopez de Prado and Peijan (2004), Lopez de Prado and Rodrigo (2004)). Fourth, EM algorithms are computationally intensive as a function of the sample size and tend to get trapped on local minima (Xu and Jordan (1996)). Speed, and therefore simplicity, is a critical concern, considering that datasets nowadays often exceed hundreds of millions of observations. Fifth, a mixture of two Gaussians offers sufficient flexibility for modeling a wide range of skewness and kurtosis scenarios. Risk and portfolio managers would greatly benefit from an intuitive algorithm that liberates them from the ubiquitous assumption of Normality.

In this paper we present a new, practical approach to exactly matching the first three moments of a mixture of two Gaussians. The fourth and fifth moments are used to guide the convergence of the mixing probability, but they are not exactly matched. We call this algorithm EF3M, as it delivers an Exact Fit of the first 3 Moments. We believe this framework is more representative of the standard problem faced by many researchers. Our examples are inspired by financial applications, however the algorithm is valid to any mixture of two Gaussians in general. Our algorithm is purely algebraic -given the first few moments of the mixture, we algebraically estimate the means and variances of the two Gaussians and the parameter p for mixing them. The only interaction with the data is in extracting the moments of the mixture. Thus, unlike ML and EM, the EF3M algorithm does not require numerically intensive tasks, and its performance is
independent of the sample size, making it more efficient in "big data" settings, like high-frequency trading.

The moments used to fit the mixture may be derived directly from the data or be the result of an annualization or any other type of time projection, such as proposed by Meucci (2010). For example, we could estimate the moments based on a sample of monthly observations, project them over a horizon of one year (i.e., the projected moments for the implied distribution of annual returns), and then fit a mixture on the projected moments, which can then be used to draw random annual (projected) returns.

Standard structural break tests (see Maddala and Kim (1999) for a treatise on the subject) attempt to identify a "break" or permanent shift from one regime to another within a time series. In contrast, the methodology we present here signals the emergence of a new regime as it happens, while it co-exists with the old regime (thus the mixture). This is a critical advantage of EF3M, in terms of providing an early signal. For example, in the particular application discussed in Section 4, the "portfolio oversight" department will be able to assess the representativeness of a track record very early in their post-track observations. The assumptions and data demands are minimal.

The rest of the paper is organized as follows: Section 2 presents a brief recitation on mixtures. Section 3 introduces the EF3M algorithm. A first variant uses the fourth moment to lead the convergence of the mixing probability, \( p \). A numerical example and the results of a Monte Carlo experiment are presented. In this variant of EF3M, the fifth moment is merely used to select one solution for each run of EF3M. Some researchers may have enough confidence and understanding of the fifth moment to use it for guiding the convergence of \( p \), in which case we propose a second variant of EF3M. Section 4 introduces the concept of Probability of Divergence. Section 5 discusses possible extensions to this methodology. Section 6 outlines our conclusions. Three mathematical appendices proof the equations used by EF3M, and a fourth appendix offers the Python code that implements both variants. Sometimes a researcher is only concerned with modeling the variance and tails of the distribution, and not with its mean and skewness. For that particular case, Appendix 5 provides an exact analytical solution.

2. MIXTURES OF DISTRIBUTIONS

Here we provide the highlights of the theory behind mixtures of distributions. Readers familiar with the topic may prefer to skip it and move to Section 3.

Consider a set of distributions \( D_0, D_1, \ldots, D_{n-1} \) and positive real coefficients \( \alpha_0, \alpha_1, \ldots, \alpha_{n-1} \) that sum to one. A new distribution \( D \) can be defined as a convex combination \( D_0, D_1, \ldots, D_{n-1} \) with coefficients \( \alpha_0, \alpha_1, \ldots, \alpha_{n-1} \), where \( D = \sum_{i=0}^{n-1} \alpha_i D_i \). If \( d_0, d_1, \ldots, d_{n-1} \) are the densities associated with \( D_0, D_1, \ldots, D_{n-1} \) then it is immediate that \( d = \sum_{i=0}^{n-1} \alpha_i d_i \) is the density associated with \( D \).

We now give a more conceptual description of \( D \). We put a standard probability measure on \( \mathbb{R} \) in the usual way by setting \( \mu([a, b]) = \int_a^b dx \) and similarly \( \mu([a, b]) = \int_a^b d \, dx \). Let \((X_i, \nu_i)\) be measure spaces isomorphic to \((\mathbb{R}, \mu_i)\) where the \( X_i's \) are pairwise disjoint. Define a new measure space \((Y, v)\) by putting \( Y = \bigcup_i X_i \) and \( v(A) = \sum_i \alpha_i \nu_i(A \cap X_i) \). If \( f : X_i \to \mathbb{R} \) is the isomorphism
between $X_i$ and $\mathbb{R}$, then we can define a random variable $f : X \rightarrow l$ by setting $f(y) = f_{iiy}$ if $y \in X_i$.

Heuristically, $f$ is defined as follows: We first choose $i$ with probability $\alpha_i$. Then, conditioned on this choice, we use the measure $\nu_i$ to choose $a y \in X_i$. The value of $f$ is $f(\nu_i y)$. Direct computation then shows that $D$ is the distribution of $f$. Indeed the probability that $f(\nu_i y) \in [a, ft]$ is given as $\sum_i \alpha_i E_i[f(y)] \in [a, ft]$, where $E_i$ is the probability conditioned on $y \in X_i$. This in turn is $\sum_i \alpha_i \int_a^b f_i(x) dx$.

Summarizing in the case of two distributions $D_0$ and $D_1$, we can build a third by first choosing with some probabilities $\{p, 1 - p\}$ either the first or second distribution and then using the corresponding density to choose a value for our random variable. Then the new distribution $D$ is given formally by $pD_0 + (1 - p)D_1$. We will denote a $D$ distribution built this way a mixture of $D_0$ and $D_1$.

In general an arbitrary distribution $D$ can be decomposed into mixtures $D = \sum \alpha_i D_i$ in infinitely many ways. However there is a unique canonical distribution of a mixture of Gaussians. Namely, if $D = \sum_i \alpha_i D_i = \sum_j \beta_j E_j$ where each $D_i$ and each $E_j$ is Gaussian, then for all $i$, $\alpha_i = \beta_1$ and $D_i = E_i$. This follows immediately from the linear independence of the family of Gaussian density functions: If we let $d(a, b, c) = \exp(ax^2 + bx + c)$, then the collection of functions $\mathcal{F} = \{1\} \cup \{d(a, b, c) : \text{at least one of a or } b \text{ is not zero}\}$ forms a linearly independent family over $\mathbb{R}$. Hence a density $d$ can be expressed in at most one way as a mixture of functions from $\mathcal{F}$.

Since $\mathcal{F}$ contains all of the density functions of Gaussians, a representation of an arbitrary density function as a mixture of Gaussians is unique.

3. THE EF3M ALGORITHM

Suppose that we are given the first four or five moments of a distribution $D$ that we assume to be a mixture of two Gaussian distributions $D_0$ and $D_1$. How can we estimate the parameters determining $D_0$, $D_1$ and the probability $p$ giving the mixture? This requires estimating five parameters in total: The mean and variance of the first Gaussian, the mean and variance of the second Gaussian, and the probability with which observations are drawn from the first distribution. Knowing that probability, $p$, implies the probability for the second distribution, $1 - p$, because by definition the sum of both probabilities must add up to one.

If $D$ is a mixture of Gaussians then the moments of $D$ can be computed directly from the five parameters determining it. In Appendix 1 we derive $D$’s moments from $D$’s parameters. Unfortunately, in general, knowledge of the first five moments of a mixture of Gaussians is not sufficient to recover the unique parameters of the mixture, so we cannot reverse this computation. On the other hand, using higher moments to recover a unique set of parameters is problematic, as they have substantial measuring errors. Our approach to finding $D$ starts with the first five observed moments about the origin $E[r^i]$ determined by data sampled from $D$ (which we assume to be a mixture of Gaussians). The algorithm starts with some random data and generates mixture parameters $(\mu_1, \mu_2, \sigma_1, \sigma_2, \rho)$ that give implied $E[r^i]$ well approximating $\bar{E}[r^i]$. As we will see later, all we need is an estimate of the first four or five moments, $\bar{E}[r^i]$. 

062
computed on a sample or population. This is the only stage of our analysis in which we deal with actual observations. Should the moments have been computed about the mean instead of the origin, Appendix 1 also shows how to derive the latter from the former.

**Notation:** Let \( \mu_1, \mu_2 \) be the means of the first two distributions, \( \sigma_1, \sigma_2 \) the standard deviations, and \( p \) be the probability determining the mixing. We use the notation \( E[r^t] \) to denote the \( t^{\text{th}} \) moment of our mixture, as implied by \( \mu_1, \mu_2, \sigma_1, \sigma_2 \) and \( p \). Appendix 1 shows how the mixture’s moments are implied from the mixture’s parameters. Later in the paper we will be concerned about fitting some data we have observed that we assume is sampled from \( D \), with observed raw moments \( \bar{E}[r^t] \). We will assume that \( D \) is a mixture with actual parameters \((\tilde{\mu}_1, \tilde{\mu}_2, \tilde{\sigma}_1, \tilde{\sigma}_2, \tilde{p})\).

The equations for the moments \( E[r^t] \) in terms of \( \mu_1, \mu_2, \sigma_1, \sigma_2 \) and \( p \) have some useful properties that allow our algorithm to work:

a) The moments of \( D \) about the mean \( E[r] \) can be computed as polynomial combinations of the moments about the origin and vice versa. Thus it suffices to do our computations using moments about the origin.

b) From the expressions for the first four moments about the origin we get some rational functions that successively express \( \mu_1 \) in terms of \( \mu_2 \) and \( p, \sigma_1 \) in terms of \( \mu_1, \mu_2 \) and \( p, \sigma_2 \) in terms of the 4-tuple \((p, \mu_2, \mu_1, \sigma_1)\).

c) From the expressions for the first five moments about the origin we get another, independent, rational function that expresses \( p \) in terms of \((\mu_2, \mu_1, \sigma_1, \sigma_2)\).

This leads to a very general form of algorithm: To compute some parameters \( c_0, c_1, \ldots, c_n \), we express \( c_1 \) in terms of \( c_0, c_2 \) in terms of \((c_0, c_1)\), \( c_3 \) in terms of \((c_0, c_1, c_2)\) etc. Finally we express \( c_0 \) independently in terms of \((c_0, \ldots, c_n)\). Thus, the algorithm then runs as follows:

i. An initial guess is made for \( c_0 \).

ii. The relations defining \( c_i \) in terms of \((c_0, c_1, \ldots, c_{i-1})\) are used to compute a guess for a candidate for \( c_i \).

iii. Having defined \((c_1, \ldots, c_n)\) in terms of \( c_0 \), we get a new estimate for \( c_0 \) using its independent expression in terms of \((c_0, \ldots, c_n)\).

iv. One loops back the previous two steps to get new guesses for \((c_0, \ldots, c_n)\).

v. The algorithm runs until some termination criterion is achieved.

In the following sections we describe two algorithms of this form and consider their convergence behavior.

**3.1. CONVERGENCE OF THE FOURTH MOMENT**

In this section we describe in more detail our algorithm for recovering the actual parameters \((\mu_1, \mu_2, \sigma_1, \sigma_2, p)\), by fitting the raw moments \((E[r^1], E[r^2], E[r^3], E[r^4])\) implied by \((\tilde{\mu}_1, \tilde{\mu}_2, \tilde{\sigma}_1, \tilde{\sigma}_2, \tilde{p})\) to the observed \((E[r], E[r^2], E[r^3], E[r^4])\). We refer the reader to Appendix 2 for the expressions needed for the expressions needed for the algorithm. The algorithm starts by taking an initial guess for \( \mu_2 \) and \( p \). We then successively estimate \( \mu_1, \sigma_2 \) and \( \sigma_1 \) using equations 21, 23 and 22 from Appendix 2. Finally equation 24 allows us to get a new guess for \( p \). We iterate this procedure.
until the results are stable within some tolerance \( \varepsilon \). By trying alternative \( \mu_2 \) seeds we obtain potential solutions whose first three moments \((\bar{E}[r], \bar{E}[r^2], \bar{E}[r^3])\) exactly match \((\bar{E}[r], \bar{E}[r^2], \bar{E}[r^3])\). Among these solutions we then choose the one that minimizes the error \( \omega(\bar{E}[r^4] - \bar{E}[r^4])^2 + (1 - \omega)(\bar{E}[r^5] - \bar{E}[r^5])^2 \), where \( \omega \in \left[ \frac{1}{2}, 1 \right] \) represents the greater confidence the researcher has on \( \bar{E}[r^4] \) relative to \( \bar{E}[r^5] \). This procedure can be repeated as many times as needed to generate a distribution of mixture parameters consistent with the observed moments.

More precisely, let \( \lambda \) define the range of the search, i.e., we will scan for solutions in \( \mu_2 \in \left[ \bar{E}[r], \bar{E}[r] + \lambda \sqrt{\bar{E}[r^2] - (\bar{E}[r])^2} \right] \). For a \( \varepsilon \) tolerance threshold, this defines a step size \( \delta = \frac{\lambda \sqrt{E[r^2] - (\bar{E}[r])^2}}{\varepsilon} \). Scanning equidistant \( \mu_2 \) within that range, with a sufficiently small step size \( \delta \), is approximately equivalent to uniform sampling of \( \mu_2 \) values. Therefore the algorithm is bootstrapping the distribution of solutions from the subspace of mixture parameters that fit the first three moments.

Given the moments \((\bar{E}[r], \bar{E}[r^2], \bar{E}[r^3], \bar{E}[r^4], \bar{E}[r^5])\), EF3M algorithm requires the following steps:

1. \( \mu_2 = \bar{E}[r] \).
2. A random seed for \( \rho \) is drawn from a \( U(0, 1) \) distribution, \( 0 < \rho < 1 \).
3. Sequentially estimate:
   a. \( \mu_1 \): Eq. (22).
   b. \( \sigma_2^2 \): Eq. (24). If the estimate of \( \sigma_2^2 < 0 \), go to Step 7.
   c. \( \sigma_2^2 \): Eq. (23). If the estimate of \( \sigma_2^2 < 0 \), go to Step 7.
4. Adjust the guess for \( \rho \): Eq. (25). If invalid probability, go to Step 7.
5. Loop to Step 3 until \( \rho \) converges within a tolerance level \( \varepsilon \).
6. Store \((\mu_1, \mu_2, \sigma_1, \sigma_2, \rho)\) and the corresponding \( \bar{E}[r^i], i = 1, \ldots, 5 \).
7. Add \( \delta \) to \( \mu_2 \) and loop to Step 2 until \( \mu_2 = \bar{E}[r] + \lambda \sqrt{\bar{E}[r^2] - (\bar{E}[r])^2} \).
8. Optional tiebreak: Among all stored results, we can select the \((\mu_1, \mu_2, \sigma_1, \sigma_2, \rho)\) for which \( \omega(\bar{E}[r^4] - \bar{E}[r^4])^2 + (1 - \omega)(\bar{E}[r^5] - \bar{E}[r^5])^2 \) is minimal.

Steps 2 to 5 are represented in Figure 1. Our solution requires a small number of operations thanks to the special sequence we have followed when nesting one equation into the next (see Appendix 2). A different sequence would have led to the polynomial equations that made Cohen (1967) somewhat convoluted.

[FIGURE 1 HERE]
Because not all guessed $\mu_2$ can match all sets of first three moments, we must include the possibility of imaginary roots considered by the algorithm (steps 3.b and 3.c), and invalid probability (step 4). For a feasible $\mu_2$, each iteration of EF3M delivers values that exactly matches $E[r], E[r^2], E[r^3]$, and using $E[r^4]$ our simulations showed that the output values $p$ very quickly settle into a neighborhood of radius $\varepsilon$. Finally, the fifth moment is used for evaluation purposes only, but it is neither exactly fit (like the first three moments) nor it drives the convergence (as the fourth moment does). Appendix 4 shows an implementation of the EF3M algorithm in Python language.

The solution to the problem subject of this paper is rarely unique when the only reliable moments are the first five. This bothered Pearson (1894), who advised using the sixth moment to choose among results. But relying on noisy high moments for selecting one possible solution out of many valid ones seems quite arbitrary. Indeed, from a Bayesian perspective we would prefer working with a distribution of parameter estimates rather than with a unique value for each. Such approach is made possible by the simplicity (translated into speed) of our EF3M’s algorithm. This allows the researcher to bootstrap the distribution of alternative values for the mixture’s parameters, which can then be used to simulate competing scenarios.

3.2. A NUMERICAL EXAMPLE

In this section we consider an example distribution $D$ formed by mixing Gaussians with an arbitrarily chosen set of parameters $(\mu_1, \mu_2, \sigma_1, \sigma_2, \rho) = (-2,1,2,1,^\text{a})$. The mixture’s moments about the origin $(\bar{E}[r^i])$ and about the mean are given in the left box of Figure 2.

We apply EF3M for $\varepsilon = 10^{-4}$, $\lambda = 5$, $\omega = \frac{1}{2}$. This implies searching within the range $\mu_2 \in [0.7,7.9629]$ by trying 10,000 uniform (equidistant) partitions. We can then repeat this exercise another 10,000 times, and study the distribution of the parameter estimates.

[FIGURE 2 HERE]

For a given output $(\mu_1, \mu_2, \sigma_1, \sigma_2, \rho)$ of the algorithm, we can compare:

a) The difference between the first five moments $E[r^i]$ implied from $(\mu_1, \mu_2, \sigma_1, \sigma_2, \rho)$ and the moments $E[r^i]$ of $D$,

b) The differences between $(\mu_1, \mu_2, \sigma_1, \sigma_2, \rho)$ and $(\bar{\mu}_1, \bar{\mu}_2, \bar{\sigma}_1, \bar{\sigma}_2, \bar{\rho})$, i.e. between $(\mu_1, \mu_2, \sigma_1, \sigma_2, \rho)$ and $(-2,1,2,1,^\text{a})$.

The right two boxes in Figure 2 show the average errors over the simulation. The middle box gives the errors in the first five moments and the rightmost box shows the errors in estimating the mixture parameters. The results show that recovered parameters are generally very close to the mixture parameters from $D$. Figure 3 is a histogram showing with what frequency various estimates of $\mu_1$ occur as outputs of EF3M, in this particular example. Most of the “errors” in Figure 2 are due to the existence of an alternate solution for $\mu_1$ around $\mu_1 \approx -1.56$.

---

2 In each repetition we use the same $\mu_2$, however the values of $\rho$ differ between runs, as they are drawn from a uniform $U(0,1)$ distribution.
Figure 3 illustrates the fact that, as discussed earlier, there is not a unique mixture that matches the first (and only reliable) moments. However, faced with the prospect of having to use unreliable moments in order to be able to pick one solution, we prefer bootstrapping the distribution of possible mixture's parameters that are consistent with the reliable moments. Our approach is therefore representative of the indetermination faced by the researcher. Section 4 will illustrate how that indetermination can be injected into the experiments, thus enriching simulations with a multiplicity of scenarios.

3.3. MONTE CARLO SIMULATIONS

In the previous section we illustrated the performance of 10,000 runs of EF3M over a particular choice of $([\hat{\mu}_1, \hat{\mu}_2, \hat{\sigma}_1, \hat{\sigma}_2, p])$. The results raise the issue of whether the promising performance is related to the properties of the vector $(-2,1,2,1,\lambda)$, or whether the algorithm behaves well in general. To test this we randomly change $([\mu_1, \mu_2, \sigma_1, \sigma_2, p])$ by drawing them from the uniform distributions with boundaries $-\lambda \leq \mu_1 \leq 0 \leq \mu_2 \leq 1, 0 \leq \sigma_1 \leq 1, 0 \leq \sigma_2 \leq 1, 0 \leq p \leq 1$.

3.4. CONVERGENCE OF THE FIFTH MOMENT

We have argued that EF3M's approach is representative of the typical problem faced by most researchers fitting a mixture: Using the first four moments, for which we have some degree of confidence and theoretical interpretation, to find the distribution of the mixture's parameters. As a consequence of using only four moments, we must start with a guess $\mu_2$ and a random seed $p$. Although the fourth moment is used for the convergence of $p$, $\mu_2$ was not re-estimated in each iteration.

However, in those cases where the researcher has some confidence in the fifth moment's estimate, we could use the fourth moment to re-estimate $\mu_2$ and the fifth moment to re-estimate $p$. In this way, no parameter remains constant across the iterations, which has the further advantage of accelerating the speed of convergence. This second variant of EF3M is very similar to the first one, as it can be seen next:

1. $\mu_2 = \hat{E}[r]$.
2. A random seed for $p$ is drawn from a $U(0,1)$ distribution, $0 < p < 1$.
3. Sequentially estimate:
   a. $\mu_1$: Eq. (26).
b. \( \sigma^2 \): Eq. (26). If the estimate of \( \sigma^2 < 0 \), go to Step 8.

c. \( \sigma^2 \): Eq. (26). If the estimate of \( \sigma^2 < 0 \), go to Step 8.

4. Adjust the guess for \( \mu_2 \): Eq. (27). If imaginary root, go to Step 8.

5. Adjust the guess for \( p \): Eq. (28). If invalid probability, go to Step 8.

6. Loop to Step 3 until \( p \) converges within a tolerance level \( \epsilon \).

7. Store \((\mu_1, \mu_2, \sigma_1, \sigma_2, p)\) and the corresponding \( \bar{E}[r^i], i = 1, \ldots, 5 \).

8. Add \( \delta \) to \( \mu_2 \) and loop to Step 2 until \( \mu_2 = \bar{E}[r] + \lambda \sqrt{\bar{E}[r^2] - (\bar{E}[r])^2} \).

9. **Optional tiebreak**: Among all stored results, we can select the \((\mu_1, \mu_2, \sigma_1, \sigma_2, p)\) for which \( \omega(\bar{E}[r^4] - E[r^4])^2 + (1 - \omega)(\bar{E}[r^5] - E[r^5])^2 \) is minimal.

Appendix 3 details the relations used in this second variant of the algorithm. Steps 2 to 6 are represented in Figure 5. Note that, although we are re-estimating the value of our guesses of both \( p \) and \( \mu_2 \) during the algorithm, our initial guesses for \( \mu_2 \) are still uniformly spaced in our search interval. Thus, this second variant of the EF3M algorithm only requires one additional step (4) and a modification of the equation used in step 5. As it can be seen in Appendix 4, both variants of the EF3M algorithm can be implemented in the same code, with a single line setting the difference.

**[FIGURE 5 HERE]**

4. **A PRACTICAL APPLICATION: PORTFOLIO OVERSIGHT**

Investment styles are not immutable, but rather evolve, as prompted by technological and computational advances, among other factors (Lopez de Prado (2011)). We began our paper recalling the evolutionary motivation behind many statistical methods, and in the case of mixtures in particular. A parallel can be drawn between the slow pace at which species adapt to an environment, creating new distinct species out of a once homogeneous genetic pool, and the slow changes that take place over time within a fund. Darwinian arguments can be applied with regards to a fund's, or even a Portfolio Manager's (PM) struggle for survival in a competitive financial environment. Although a fund's or PM's investment style may evolve so slowly that those changes will be undetectable in the short-run, the track record will accumulate evidence of the "evolutionary divergence" taking place. Sometimes this divergence will occur as a PM attempts to adapt her style to prevail in a certain environment, or as an environmental change affects a style's performance. An example of the former is a new technology giving an edge to some market participants, and an example for the latter is when the rest of the market adopts that successful technology over time, thus suppressing its competitive advantage.

A hedge fund's "portfolio oversight" department assesses the operational risk associated with individual PMs, identifies desirable traits and monitors the emergence of undesirable ones. The decision to fund a PM is typically informed by her track record. If her recent returns deviate from the track record used to inform the funding decision, the portfolio oversight department must detect it. This is distinct from the role of risk manager, which is dedicated to assessing the possible losses under a variety of scenarios. For example, even if a PM is running risks below her authorized limits, she may not be taking the bets she was expected to, thus delivering a
performance inconsistent with her track record (and funding). The risk department may not notice anything unusual regarding that PM, however the portfolio oversight department is charged with policing and detecting such situation. A track record can be expressed in terms of its moments, thus the task of overseeing a PM can be understood as detecting an inconsistency between the PM's recent returns and her "approved" track record.

[FIGURE 6 HERE]

More specifically, suppose that we invest in a PM with a track record characterized by IID returns with moments listed in Figure 6. Because we have little to no knowledge regarding her investment process, we cannot be certain about how a number of evolving factors (replacement of PMs, variations to the investment process, market conditions, technological changes, financial environment, etc.) may be altering the distribution that governs those returns. We need to determine a probability that a sequence of returns is consistent with a pre-existing track record, which will inform our decision to re-allocate or possibly redeem our investment. Generally stated, at what point do we have information sufficient to assess whether a sequence of observations significantly departs from the original distribution?

This question can be reformulated in the following manner:

1. **Assumption:** Suppose that the returns of a portfolio manager are drawn from a time-invariant process, such as a process that is independent and identically distributed (IID).

2. **Data:** We are given
   a. A sequence of returns, \( \{r_t\} \), for \( t=1, \ldots, T \) (testing set).
   b. A reference distribution, based on a sample of returns available prior to \( t=l \) (training set), or some prior knowledge.

3. **Goal:** We would like to determine the probability at \( t \) that the cumulative return up to \( t \) is consistent with that reference distribution.

A first possible solution could entail carrying out a generic Kolmogorov-Smirnov test in order to determine the distance between the reference (or track) and post-track distributions. Being a nonparametric test, this approach has the drawback that it might require impractically large data sets for both distributions.

A second possible solution would be to run a structural break test, in order to determine at what observation \( t \) the observations are no longer being drawn from the reference distribution, and are coming from a different process instead. Standard structural break tests include CUSUM, Chow, Hartley, etc. However, a divergence from the reference distribution is not necessarily the result of a structural break or breaks. In our experience, a portfolio manager's style evolves slowly over time, by gradually transitioning from one set of strategies to another, in an attempt to adapt better to the investment environment -just as a species adapts to a new environment in order to maximize its chances of survival. As the new set of strategies emerge and become more prominent, the old set of strategies does not cease to exist. Therefore, there may not be a clean structural break that these tests could identify.

We propose a faster, more robust and less computationally intensive approach. The method consists of: i) applying the EF3M for matching the track record's moments, ii) simulating path
scenarios consistent with the matched moments, iii) deriving a distribution of scenarios based on that match and iv) evaluate what percentile of the distribution corresponds with the PM's recent performance. Note that there is nothing in the EF3M algorithm that takes any time structure into account, which might be present in the reference and/or target data sets. The reason is, the portfolio manager's returns are assumed to be drawn from a time-invariant process, such as a process that is independent and identically distributed (IID), which is the standard assumption used by capital allocation methodologies. If the process is not time-invariant, and as a result the post-track process significantly diverges from the track process, it is the goal of this approach to bring that situation to the attention of the portfolio oversight officer.

An important feature of EF3M is its ability to estimate a distribution of the possible mixture parameters of our data using information on the reliable moments. Step 2 simulates a path scenario for each output and step 3 uses this distribution on mixture parameters to get a cumulative distribution of returns at a given horizon. Thus at time \( t \) we can ask what percentile a given cumulative return corresponds to, relative to a collection of simulations corresponding to all of the outputs of the EF3M algorithm (step 4). The results allow us to determine difference percentiles associated with each drawdown and each time under the water.

4.1. ESTIMATING THE DISTRIBUTION OF MIXTURE'S PARAMETERS

Our procedure starts using the EF3M algorithm to search for parameters that give mixtures whose moments closely match those of the track record's non-Normal distribution. For this particular exercise, the first variant of the EF3M algorithm will be applied, though the second variant would work equally well given reliable information about the fifth moment. Using the first four moments given in Figure 6, we have run the EF3M algorithm 100,000 times and obtained a distribution of parameter estimates for a mixture of two Gaussians. Figure 7 displays the moments' estimation errors \((E[r'] - E[r''])\) and average fitted parameters \((\mu_1, \mu_2, \sigma_1, \sigma_2, \rho)\).

4.2. SIMULATING PERFORMANCE PATHS ON THE PARAMETERS' DISTRIBUTION

Let's define the cumulative return from \( t-h \) to \( t \), denoted \( R_{t-h} \), as

\[
R_{t-h} = \prod_{i=t-h+1}^{t} (1 + rd, h = 1, \ldots, t)
\]

By making \( h=t \), we are computing each cumulative return looking back the full available post-track sample (an increasing window). We would like to simulate paths of cumulative returns \((R_{t-t})\) consistent with the observed moments of simple (non-cumulative) returns (77). A Monte Carlo simulation of \( R_{t-t} \) can be computed by making random draws of \( r_i \). But which of the mixture solutions should we use? One option is to pick one of the \((\mu_1, \mu_2, \sigma_1, \sigma_2, \rho)\), e.g. the mode of the five-dimensional distribution of parameter estimates computed earlier. The problem with that option is that there are several valid combinations of parameters, some more likely than

[FIGURE 7 HERE]
others. Figure 8 plots the pdf for a mixture that delivers the same moments as stated in Figure 6. We cannot however postulate any particular parameter values to characterize the true ex-ante distribution, as there are multiple combinations able to deliver the observed moments.

[FIGURE 8 HERE]

A better approach consists in running one Monte Carlo path \( R_{t,i} \) for each of the 100,000 solutions estimated earlier. Note that by associating an entire Monte Carlo path to the output from each run of our EF3M algorithm, we are implicitly giving higher weight to some outputs than others. This is due to the fact the outputs occur with different multiplicities. Outputs with high multiplicity are more heavily weighted than low multiplicity outputs. For example, the output corresponding to \( \mu_1 = -2.03 \) in Figure 3 would occur over 1,400 times and thus would be weighted heavily in the aggregated data about the Monte Carlo simulations corresponding to the moments given in Section 3.2.

4.3. THE DISTRIBUTION OF CUMULATIVE RETURNS

For observed return \( \{ r_t; 1 \leq t \leq T \} \) we can compare each \( R_{t,i} \) with the expected returns predicted by our simulations. More precisely, the simulation we described gives us an approximation to the cumulative distribution \( CDF_t: \mathbb{R} \to [0,1] \), where \( CDF_t(x) \) is the probability that the return on the portfolio from our original distribution \( D \) is less than or equal to \( x \). By collecting our 100,000 simulated \( R_{t,i} \) for a given \( t \), we can derive an approximation to its \( CDF_t \).

The cumulative distribution functions are consistent with the observed moments on simple returns and incorporate information about a variety of likely mixtures. The next step is to determine the different percentiles associated with each drawdown level and time under the water. Figure 9 plots various percentiles for each \( CDF_t \). For example, with a 99% confidence, drawdowns of more than 5% from any given point after 6 observations would not be consistent with the ex-ante distribution of track record returns. Furthermore, even if the loss does not reach 5%, a time under the water beyond one year is highly unlikely (2.5% probability), thus it should alert the investor regarding the possibility that the track record’s moments (and its Sharpe ratio in particular) are inconsistent with the current performance.

[FIGURE 9 HERE]

4.4. PROBABILITY OF DIVERGENCE

Finally, we are in a position to define the Probability of Divergence, \( PD_t \), updated with every new observation, as

\[
PD_t \left( R_{t,i} \right) = 2 \left| CDF_t(R_{t,i}) - \frac{1}{2} \right|
\]

We interpret this number as follows: At time \( t \), \( R_{t,i} \) is the total cumulative rate of return from observation 1 to \( t \). Applying \( CDF_t \) to the number \( R_{t,i} \) give us (our best approximation to) the percentile rank of \( R_{t,i} \). In particular if \( R_{t,i} \) is exactly the median predicted return, \( PD_t \left( R_{t,i} \right) = 0 \),

This (non-Normal) \( CDF_t \) is on the cumulative returns, not the simple returns.
0. More generally, if $PD_t \left( R_{t,t} \right) = \alpha$, then $R_{t,t}$ is either in the bottom or top $1 - \alpha$ proportion of the predicted returns, depending on the sign of $CDF_t(R_{t,t}) - \frac{1}{2}$. Viewed in this light, $PD_t \left( R_{t,t} \right)$ measures the proportional departure from the median of our simulated returns.

Figure 10 plots 1,000 returns generated from a mixture of Gaussians with moments matching those in Figure 6, namely $(\mu_1, \mu_2, \sigma_1, \sigma_2, \rho) = (-0.025, 0.015, 0.02, 0.01, 0.1)$. PD may sporadically reach high levels, without becoming extreme permanently. What would happen if draws from the first Gaussian become more likely? For example, if $p=0.2$ instead of $p=0.1$, the mixture's distribution would become more negatively skewed and fat-tailed. As Figure 11 evidences, that situation is distinct from the approved track-record, and PD slowly but surely converges to 1.

[FIGURE 10 HERE]

[FIGURE 11 HERE]

Figure 12 presents an example computed on a sequence of 1,000 returns distributed IID Normal that match the mixture's mean and variance, i.e. $N(\mu, \sigma^2) = N(1.10E, 02.2, 74E - 04)$. PD approaches 1, although the model cannot completely discard the possibility that these returns in fact were drawn from the reference mixture.

[FIGURE 12 HERE]

Figure 13 presents another example computed on a sequence of 1,000 returns distributed IID Normal with a mean half the mixture's and the same variance as the mixture, i.e. $N(\mu, \sigma^2) = N(0.5, 02.74E - 04)$. PD quickly converges to 1, as the model recognizes that those Normally distributed draws do not resemble the mixture's simulated paths.

[FIGURE 13 HERE]

As measured above, an increase in the probability of divergence may not always be triggered by a change in the style, but in the way the style fits to changing market conditions. That distinction may be more of a philosophical disquisition, because either cause of an increase in the probability of departure (change of style or change of environment) should be brought up to the attention of the portfolio oversight officer, and invite a review of the capital allocated to that portfolio manager or strategy.

4.5. CROSS-VALIDATION

Suppose that the portfolio oversight officer sets a threshold $PD^*$, above which the probability of departure is deemed to be unacceptably high. Further suppose that $T$ observations are available out-of-sample (i.e., not used in the EF3M estimation of the mixture's parameters), and that $PD_T \left( R_{T,T} \right) > PD^*$. Should the large enough for estimating the five moments with reasonable

---

4 We are thankful to the referee for suggesting this Section.
confidence, it is possible to cross-validate the result that divergence has occurred, following these steps:

1. We divide the sample of observations into two samples: In-sample (IS) and out-of-sample (OOS). We assume that both samples are long enough for providing accurate estimates of five moments.
   a. IS: The training set, used to estimate the set of mixture parameters \((\mu_1, \mu_2, \sigma_1, \sigma_2, \rho)\).
   b. OOS: The testing set, used to calculate \(PD_t(R_{t,t})\), using the fitted parameters \((\mu_1, \mu_2, \sigma_1, \sigma_2, \rho)\).
2. Apply the EF3M algorithm OOS, to compute the mixture's parameter estimates on the testing set. We denote these parameters \((\mu_1, \mu_2, \sigma_1, \sigma_2, \rho)^{OOS}\), to distinguish them from the set of parameters IS, \((\mu_1, \mu_2, \sigma_1, \sigma_2, \rho)\).
3. Using \((\mu_1, \mu_2, \sigma_1, \sigma_2, \rho)^{OOS}\), compute \(PD_{IS}^{OOS}(R_{t,t})\) on the IS data.
4. Assess whether \(PD_{IS}^{OOS}(R_{t,t}) > PD^*\). If that is the case, the divergence has been cross-validated. If not, additional evidence may be required, in the form of a longer \(T\).

5. EXTENSIONS
A first possible extension of this approach would consist in allowing for any number of constituting distributions, not only two. However, that would require fitting a larger number of higher moments, which we have advised against on theoretical and empirical grounds. Also, if the divergence is caused by two or more new distributions, our PD statistic is expected to detect that situation as well, since it is able to detect the more challenging case of only one emerging style.

A second possible extension would mix multivariate Gaussian distributions. An advantage of doing so would be that we could directly track down which PMs are the source of a fund's divergence, however that would come at the cost of again having to use higher moments to fit the additional parameters. The source of the divergence can still be investigated by running this univariate procedure on subsets of PMs.

A third possible extension would involve modeling mixtures of other parametric distributions beyond the Gaussian case. That is a relatively simple change for the most common functional forms, following the same algebraic strategy presented in the Appendix.

6. CONCLUSIONS
In this paper we have described a method of evaluating the probability that a PM's returns correspond to a reference distribution (denoted Probability of Divergence), which answers a critical concern of portfolio oversight. Our method gives investors the ability to assess the representativeness of a PM's track record very early in her post-track observations. It is based on an algorithm for finding the parameters determining a mixture of Gaussians just from the first four or five moments of a given mixture \(D\). Determining these moments is the only interaction the algorithm has with the underlying data.
Accordingly, we have devised the EF3M algorithm, which exactly matches the first three moments (on which the researcher usually has greatest confidence), reserving the fourth moment for guiding the convergence of the mixture probability. That the algorithm converges to a solution based on the first four moments is consistent with a theoretical understanding of their meaning. The fourth moment is closely approximated but not exactly matched, because of its sampling error. In a second variant of the EF3M algorithm, we also allow the fifth moment to lead the convergence of the algorithm, should the researcher be confident in that moment's estimate.

The decomposition of a mixture of Gaussians into its component distributions is rarely unique when the only reliable inputs are the first four or five moments. Rather than searching for a unique solution, we advocate computing a distribution of probability for the fitted parameters. This is a Bayesian-like approach, by which we would simulate a large variety of scenarios consistent with the probable values of the mixture's parameters. This approach is made possible thanks to the relative simplicity (translated into speed) of our EF3M's algorithm. Monte Carlo experiments confirm the validity of our method.

Originally inspired by Galton and Pearson's "Mathematical Theory of Evolution", mixtures of Gaussians are nowadays widely used in a number of scientific applications. The problem of fitting the characteristic parameters for a mixture of two Gaussians has received a number of solutions over the last 120 years. We have identified several scenarios under which MM, ML and EM algorithms may not fully address the problems faced by many researchers, particularly in the field of Quantitative Finance. MM algorithms present the disadvantage that the solution is impacted by the sample error of the fourth moment. Besides, requiring a fifth moment introduces the problem of basing our solution on a moment for which the researcher typically has no theoretical interpretation. ML and EM algorithms also suffer the criticism of sampling error and theoretical interpretation of the moments used, besides getting trapped on local minima and increased computational intensity as a function of sample size.

An analogue can be made between: (a) the slow pace at which species adapt to an environment, which often results in the emergence of a new distinct species out of a once homogeneous genetic pool, and (b) the slow changes that take place over time within a fund, mutating its investment style. A fund's track record provides a sort of genetic marker, which we can use to identify mutations. This has motivated our use of a biometric procedure to detect the emergence of a new investment style within a fund's track record. In doing so, we answer the question: "What is the probability that a particular PM's performance is departing from the reference distribution used to allocate her capital?" Overall, we believe that EF3M is well suited to answer this critical question.
APPENDICES

A.1. HIGHER MOMENTS OF A MIXTURE OF m NORMAL DISTRIBUTIONS

Let \( z \) be a random variable distributed as a standard normal, \( z \sim \mathcal{N}(0,1) \). Then, \( \eta = \mu + \sigma \zeta \sim \mathcal{N}(\mu, \sigma^2) \), with characteristic function:

\[
0_{\eta}(s) = E[e^{is\eta}] = E[e^{is\eta}]E[e^{is\zeta}] = e^{is\mu} e^{\frac{1}{2}is^2\sigma^2} \quad (3)
\]

Let \( r \) be a random variable distributed as a mixture of \( m \) normal distributions, \( r \sim \mathcal{D}(\mu_1, ..., \mu_m, \sigma_1, ..., \sigma_m, p_1, ..., p_m) \), with \( \sum_{j=1}^{m} p_j = 1 \). Then:

\[
\vartheta_r(s) = E[e^{i\sigma r}] = e^{\sum_{j=1}^{m} p_j (is\mu_j - \frac{1}{2}s^2\sigma_j^2)} \quad (4)
\]

The \( k \)-th moment centered about zero of any random variable \( x \) can be computed as:

\[
E[x^k] = \left. \frac{d^k \vartheta_r(s)}{ds^k} \right|_{s=0} \quad (5)
\]

We can use the characteristic function to compute the first five moments about the origin (or centered about zero) in the case of a mixture of \( m \) Gaussians as:

\[
E[r] = \sum_{j=1}^{m} p_j \mu_j \quad (6)
\]

\[
E[r^2] = \sum_{j=1}^{m} p_j (\sigma_j^2 + \mu_j^2) \quad (7)
\]

\[
E[r^3] = \sum_{j=1}^{m} p_j (3\sigma_j^2 \mu_j + \mu_j^3) \quad (8)
\]

\[
E[r^4] = \sum_{j=1}^{m} p_j (3\sigma_j^4 + 6\sigma_j^2 \mu_j^2 + \mu_j^4) \quad (9)
\]

\[
E[r^5] = \sum_{j=1}^{m} p_j (15\sigma_j^4 \mu_j + 10\sigma_j^2 \mu_j^3 + \mu_j^5) \quad (10)
\]

A.1.1. FROM MOMENTS ABOUT ZERO TO MOMENTS ABOUT THE MEAN

We can use the first moments about the origin (Eqs. 6-10) together with Newton’s binomium (Eq. 11) to derive the moments about the mean (Eqs. 12-16):
\[ E[(r - E[r])^k] = \sum_{j=0}^{k} (-1)^j \binom{k}{j} (E[r])^j E[r^{k-j}] \]  

(11)

\[ E[r - E[r]] = 0 \]  

(12)

\[ E[(r - E[r])^2] = E[r^2] - (E[r])^2 \]  

(13)

\[ E[(r - E[r])^3] = E[r^3] - 3E[r^2]E[r] + 2(E[r])^3 \]  

(14)


(15)

\[ E[(r - E[r])^5] = E[r^5] - 5E[r^4]E[r] + 10E[r^3](E[r])^2 - 10E[r^2](E[r])^3 + 4(E[r])^5 \]  

(16)

**A.1.2. FROM MOMENTS ABOUT THE MEAN TO MOMENTS ABOUT ZERO**

We have computed the moments about the mean from the moments about the origin. Using Eqs. (12)-(16), the reverse transformation can be carried out easily:

\[ E[r^k] = (-1)^{k+1}(E[r])^k \]  

(17)

\[ E[r^2] = E[(r - E[r])^2] + (E[r])^2 \]  

(18)


(19)

\[ E[r^4] = E[(r - E[r])^4] + 4E[r^3]E[r] - 6E[r^2](E[r])^2 + 3(E[r])^4 \]  

(20)

\[ E[r^5] = E[(r - E[r])^5] + 5E[r^4]E[r] - 10E[r^3](E[r])^2 + 10E[r^2](E[r])^3 - 4(E[r])^5 \]  

(21)

**A.2. EF3M CONVERGENCE USING THE 4TH MOMENT**

For a given \( \mu_2 \), we would like to find the \((\mu_1, \sigma_1, \sigma_2)\) that match the observed \((E[r], E[r^2], E[r^3], E[r^4])\), with \(p_1\) approximating \(E[r^4]\).

Let \(p = p_1 = 1 - p_2\). With knowledge of the first four non-centered moments of the mixture, \(E[r], E[r^2], E[r^3], E[r^4]\), we can define the following relations among the mixture’s parameters. If the known moments are centered, the non-centered moments can be readily computed from Eqs. (10)-(14).

From Eq. (6), we insert our observation \(E[r]\) to derive
Likewise, from Eq. (7) we obtain

\[ \sigma_T^d = \frac{\bar{E}[r^2] - \sigma_2^2 - \mu_2^2}{P} + \sigma_2^2 + \mu_2^2 - \mu_1^2 \]  

(23)

Inserting Eq. (23) in Eq. (8) leads to

\[ \gamma_4^2 = \frac{\bar{E}[r^3] + 2\rho \mu_1^3 + (p - 1)\mu_2^3 - 3\mu_1 (\bar{E}[r^2] + \mu_2^2(p - 1))}{3(1 - \rho)(\mu_2 - \mu_1)} \]  

(24)

For a seed \((\mu_2, p)\), these relations give us the \(\mu_1, \sigma_1, \sigma_2\) that match the first three moments. An algorithm can then be created to approximate (without exactly matching) the fourth moment by re-estimating \(p\). For that, we need a new relationship, which can be derived from Eq. (9)

\[ p = \frac{\bar{E}[r^4] - 3\sigma_2^4 - 6\sigma_2^2\mu_2^2 - \mu_2^4}{3(\sigma_1^4 - \sigma_2^4) + 6(\sigma_1^2\mu_1^2 - \sigma_2^2\mu_2^2) + \mu_1^4 - \mu_2^4} \]  

(25)

Because we don't have any relationship to re-estimate \(\mu_2\), that parameter remains fixed through every iteration of the algorithm. A fifth moment would be needed to allow for \(\mu_2\)'s convergence, as described in the next Section.

A.3. EF3M CONVERGENCE USING THE \(\gamma_4^d\) MOMENT

We will start by using some of the relationships identified earlier. In particular, for an initial \((\mu_2, p)\)

\[ \mu_1 = \frac{\bar{E}[r] - (1 - \rho)\mu_2}{p} \]  

\[ \sigma_1^2 = \frac{\bar{E}[r^2] - \sigma_2^2 - \mu_2^2}{P} + \sigma_2^2 + \mu_2^2 - \mu_1^2 \]  

(26)

\[ \gamma_4^2 = \frac{\bar{E}[r^3] + 2\rho \mu_1^3 + (p - 1)\mu_2^3 - 3\mu_1 (\bar{E}[r^2] + \mu_2^2(p - 1))}{3(1 - \rho)(\mu_2 - \mu_1)} \]  

These are the same relations that exactly match the first three moments. Now we can use the fourth moment to re-estimate \(\mu_2\), thus allowing it to converge. From Eq. (9),

\[ \mu_2 = \left[ -3\sigma_2^2 \pm \left(6\sigma_2^4 + \frac{\bar{E}[r^4]}{1 - p}(3\sigma_1^4 + 6\sigma_1^2\mu_1^2 + \mu_1^4) \right)^{1/2} \right]^{1/2} \]  

(27)
but we only need to evaluate the "+" from "+", because $\sigma_z^2 > 0$.

Eq. (10) allows us to use the fifth moment to lead $p$’s convergence,

$$p = \frac{E[r^5] \cdot b}{a - b}$$

(28)

with

$$a = 15\sigma_1^4 \mu_1 + 10\sigma_1^2 \mu_2 + \mu_5^4$$

$$b = 15\sigma_2^4 \mu_1 + 10\sigma_2^2 \mu_2 + \mu_5^4$$

(29)

Unlike in the previous case, this solution incorporates a relationship to re-estimate $\mu_2$ in each iteration. We are still matching the first three moments, with the difference that now moments fourth and fifth drive the convergence of our initial seeds, $(\mu_2, p)$.

A.4. EF3M IMPLEMENTATION IN PYTHON

Both variants of the EF3M algorithm are implemented in the following code. For the first variant, comment the line `parameter s=iter5(mu2,p1_self.moments)` and leave uncommented the line `parameter s=iter4(mu2,p1_self.moments)`. Do the reverse for the second variant.

```python
#!/usr/bin/env python
# EF3M class for the exact fit of a mixture of two Gaussians
# On 20120217 by MLDP <lopezdeprado@lbl.gov>

import random

#-------------------------------------------
# Define the problem here
moments=[0,7,2,6,0,4,25,-59.8] # about the origin
epsilon=10**-.5
factor=5 # this is the 'lambda' referred in the paper

#-------------------------------------------
# This is the mixture's class
class M2N:
    def __init__(self,moments):
        self.moments=moments
        self.parameters=[0 for i in range(5)]
        self.error=sum([moments[i]**2 for i in range(len(moments))])

    def fit(self,mu2,epsilon):
        p1=random.random()
        numiter=0
        while True:
            numiter+=1
            try:
                #parameters=iter4(mu2,p1_self.moments) # for the first variant
                parameters=iter5(mu2,p1_self.moments) # for the second variant
                except:
```

21
def binomialCoeff(n, k):
    return

    moments = get_moments(parameters)
    error = sum([(self.moments[i] - moments[i])**2 for i in range(len(moments))])
    if error < self.error:
        self.parameters = parameters
        self.error = error
    if abs(p1 - parameters[4]) < epsilon:
        return
    p1 = parameters[4]
    m2 = parameters[1]  # for the 5th moment's convergence

    def get_moments(parameters):
        m1 = parameters[0] + (1 - parameters[4]) * parameters[1]
        m2 = parameters[2] + parameters[0]**2 + (1 - parameters[4]) * (parameters[3]**2 + parameters[1]**2)
        m3 = parameters[4] * (3 * parameters[2]**2 + parameters[0]**3) + 
            (1 - parameters[4]) * (3 * parameters[3]**2 + parameters[1]**3)
        m4 = parameters[4] * (6 * parameters[2]**4 + 6 * parameters[0]**4) + 
            parameters[0]**4 + (1 - parameters[4]) * (3 * parameters[3]**4 + 6 * parameters[1]**4)
        m5 = parameters[4] * (15 * parameters[2]**5 + 10 * parameters[0]**5) + 
            parameters[0]**5 + (1 - parameters[4]) * (15 * parameters[3]**5) + 
            parameters[1]**5
        return [m1, m2, m3, m4, m5]

    def iter4(mu2, p1, moments):
        mu1 = moments[0] - (1 - p1) * mu2
        sigma2 = ((moments[2] - 2 * p1 * mu1**2 - (1 - p1) * mu2**2 - 3 * mu1 * (moments[1] + mu2**2) - 
                    (1 - p1)) * (3 * (1 - p1) * (mu2 - mu1)))**(1.5)
        sigma1 = ((moments[1] - sigma2**2 - mu2**2) / p1 + sigma2**2 + mu2**2 - mu1**2)**(1.5)
        p1 = (moments[3] - 3 * sigma2**2 - 4 * sigma2**2 * mu2**2 - 2 * mu2**2 - mu1**2) / 
             (3 * sigma1**4 + 6 * sigma1**2 * sigma2**2 + mu1**4 - mu2**4)
        return [mu1, mu2, sigma1, sigma2, p1]

    def iter5(mu2, p1, moments):
        mu1 = moments[0] - (1 - p1) * mu2
        sigma2 = ((moments[2] - 2 * p1 * mu1**2 - (1 - p1) * mu2**2 - 3 * mu1 * (moments[1] + mu2**2) - 
                    (1 - p1)) / (3 * (1 - p1) * (mu2 - mu1)))**(1.5)
        sigma1 = ((moments[1] - sigma2**2 - mu2**2) / p1 + sigma2**2 + mu2**2 - mu1**2)**(1.5)
        a = 6 * sigma2**4 + (moments[3] - p1 * (3 * sigma1**4 + 6 * sigma1**2 * sigma2**2 + mu1**4 - mu2**4)) / 
           (1 - p1)**5
        mu2 = (a - 3 * sigma2**2)**.5
        a = 15 * sigma1**4 * mu1 + 10 * sigma1**2 * mu1**3 + mu1**5
        b = (a - 15 * sigma1**4 * mu2 + 10 * sigma1**2 * mu2**3 + mu2**5)
        p1 = (moments[4] - b) / (a - b)
        return [mu1, mu2, sigma1, sigma2, p1]

    def binomialCoeff(n, k):
        # Number of combinations of n over k
A.5. FITTING A MIXTURE TO AN OBSERVED VARIANCE AND KURTOSIS

Let \( r \) be a random variable distributed as a mixture of 2 Normal distributions, \( r \sim D(\mu_1, \mu_2, \sigma_1, \sigma_2, p_1, p_2) \), with \( p_1 + p_2 = 1 \) and 4th population moment about the origin \( E[r^4] \). Given some observed moments \( \bar{E}[r^2], \bar{E}[r^4] \), we would like to estimate the symmetric mixture \( (E[r^3] = 0) \) centered about zero \( (E[r] = 0) \) such that its population moments match \( E[r^2] = \bar{E}[r^2] \) and \( E[r^4] = \bar{E}[r^4] \).

Rewrite \( p_2 = l - p_1 \). We have five free parameters \((\mu_1, \mu_2, \sigma_1, \sigma_2, p_1)\) to match only four moments \((\bar{E}[r] = 0, \bar{E}[r^2] = \bar{E}[r^2], E[r^3] = 0, E[r^4] = \bar{E}[r^4])\). A mixture of two Gaussians has mean \( E[r] = p_1 \mu_1 + (1 - p_1) \mu_2 \) and a third moment about the origin \( E[r^3] = p_1 (3 \sigma_1^2 \mu_1 + \mu_1^3) + (1 - p_1) (3 \sigma_2^2 \mu_2 + \mu_2^3) \). Thus, \( \mu_1 = \mu_2 = 0 \) meets our requirement that \( E[r] = E[r^3] = \)}
We still have three free parameters ($\sigma_1, \sigma_2, P_t$) to match the two remaining moments ($E[r^2] = \tilde{E}[r^2], E[r^4] = \tilde{E}[r^4]$).

From Eqs. (7) and (9), this particular problem reduces to the system

$$\tilde{E}[r^2] = \rho_1 \sigma_1^2 + (1 - p_1) \sigma_2^2$$

$$\frac{1}{3} \tilde{E}[r^4] = p_1 \sigma_1^4 + (1 - p_1) \sigma_2^4$$

We find solutions in

$$\sigma_2^2 = \tilde{E}[r^2] \pm \left( \frac{p_1}{1 - p_1} \left( \frac{1}{3} \tilde{E}[r^4] - \left( \tilde{E}[r^2] \right)^2 \right) \right)^{1/2} \quad (31)$$

$$\sigma_1^2 = \frac{1}{p_1} \left( \tilde{E}[r^2] - (1 - p_1) \sigma_2^2 \right)$$

but because the system is symmetric in $\sigma_1$ and $\sigma_2$, it suffices to evaluate the "+" in "±". In that way, $\sigma_2 \geq \sigma_1$. A pending question is, what is the appropriate value for $p_1$? From the above equations we find that, in order to find roots in the real domain, an additional condition is

$$\tilde{E}[r^2] > (1 - p_1) \sigma_2^2$$

which, after replacing $\sigma_2^2$ with its solution, leads to

$$p_1 > 1 - \frac{3(\tilde{E}[r^2])^2}{\tilde{E}[r^4]} \quad (33)$$

and obviously $\frac{\tilde{E}[r^4]}{(\tilde{E}[r^2])^2} \geq 3$, so that $0 \leq p_1 \leq 1$. Putting all pieces together, for any $0 < \delta < 1$, a feasible solution is given by

$$p_1 = 1 + (\delta - 1) \frac{3(\tilde{E}[r^2])^2}{\tilde{E}[r^4]}$$

$$\sigma_2^2 = \tilde{E}[r^2] \pm \left( \frac{p_1}{1 - p_1} \left( \frac{1}{3} \tilde{E}[r^4] - \left( \tilde{E}[r^2] \right)^2 \right) \right)^{1/2} \quad (34)$$

$$\sigma_1^2 = \frac{1}{p_1} \left( \tilde{E}[r^2] - (1 - p_1) \sigma_2^2 \right)$$
FIGURES

Figure 1 - Algorithm's flow diagram (with four moments)

<table>
<thead>
<tr>
<th>Moments</th>
<th>Origin</th>
<th>Mean</th>
<th>Errors</th>
<th>Average</th>
<th>StDev</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.7000</td>
<td>0.0000</td>
<td>1</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>2</td>
<td>2.6000</td>
<td>2.1100</td>
<td>2</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>3</td>
<td>0.4000</td>
<td>-4.3740</td>
<td>3</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>4</td>
<td>25.0000</td>
<td>30.8037</td>
<td>4</td>
<td>-0.0096</td>
<td>0.0220</td>
</tr>
<tr>
<td>5</td>
<td>-59.8000</td>
<td>-153.5857</td>
<td>5</td>
<td>0.0021</td>
<td>0.0228</td>
</tr>
</tbody>
</table>

Figure 2 - Moments (left), their estimation errors (center) and departure of the recovered parameters from the original parameters (right)

Figure 3 - Marginal distribution of probability of the $\mu_1$ parameter
### Table: Errors and Deviation

<table>
<thead>
<tr>
<th>Errors</th>
<th>Average</th>
<th>StDev</th>
<th>Parameter</th>
<th>Average</th>
<th>StDev</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.0000</td>
<td>0.0000</td>
<td>Mul</td>
<td>-0.0397</td>
<td>0.2605</td>
</tr>
<tr>
<td>2</td>
<td>0.0000</td>
<td>0.0000</td>
<td>Mu2</td>
<td>-0.0259</td>
<td>0.2698</td>
</tr>
<tr>
<td>3</td>
<td>0.0000</td>
<td>0.0000</td>
<td>Sigma</td>
<td>-0.0283</td>
<td>0.1242</td>
</tr>
<tr>
<td>4</td>
<td>-0.0012</td>
<td>0.0130</td>
<td>Sigma2</td>
<td>0.0036</td>
<td>0.1117</td>
</tr>
<tr>
<td>5</td>
<td>0.0003</td>
<td>0.0109</td>
<td>Probl</td>
<td>-0.0315</td>
<td>0.1877</td>
</tr>
</tbody>
</table>

**Figure 4** - Estimation error statistics (left) and departure of the recovered parameters from the original parameters (right)

### Figure 5 - Algorithm’s flow diagram (with 5th moment)

### Table: Moments

<table>
<thead>
<tr>
<th>Moments</th>
<th>Origin</th>
<th>Mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.10E-02</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>3.95E-04</td>
<td>2.74E-04</td>
</tr>
<tr>
<td>3</td>
<td>2.53E-06</td>
<td>-7.85E-06</td>
</tr>
<tr>
<td>4</td>
<td>4.31E-07</td>
<td>5.63E-07</td>
</tr>
<tr>
<td>5</td>
<td>-7.48E-09</td>
<td>-3.28E-08</td>
</tr>
</tbody>
</table>

**Figure 6** - Moments from the ex-ante distribution

### Table: Errors and Estimated Parameters

<table>
<thead>
<tr>
<th>Errors</th>
<th>Average</th>
<th>StDev</th>
<th>Parameter</th>
<th>Average</th>
<th>StDev</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-0.0245</td>
<td>0.0027</td>
<td>Mul</td>
<td>-0.0245</td>
<td>0.0027</td>
</tr>
<tr>
<td>2</td>
<td>0.0150</td>
<td>0.0001</td>
<td>Mu2</td>
<td>0.0150</td>
<td>0.0001</td>
</tr>
<tr>
<td>3</td>
<td>0.0201</td>
<td>0.0009</td>
<td>Sigma</td>
<td>0.0201</td>
<td>0.0009</td>
</tr>
<tr>
<td>4</td>
<td>0.0100</td>
<td>0.0002</td>
<td>Sigma2</td>
<td>0.0100</td>
<td>0.0002</td>
</tr>
<tr>
<td>5</td>
<td>0.1026</td>
<td>0.0144</td>
<td>Probl</td>
<td>0.1026</td>
<td>0.0144</td>
</tr>
</tbody>
</table>

**Figure 7** - Moments estimation errors (left) and estimated parameters (right) after 100,000 runs of EF3M
Figure 8 - Example of mixture of two Gaussians consistent with the above moments

\((\mu_1, \mu_2, \sigma_1, \sigma_2, \rho) = (-0.025, 0.015, 0.02, 0.01, 0.1)\)
Figure 9 – Performance for various confidence bands
Figure 10 - Returns and Probability of Divergence for draws from 
\((\mu_1, \mu_2, \sigma_1, \sigma_2, \rho) = (-0.025, 0.015, 0.02, 0.01, 0.1)\)
Figure 11 - Returns and Probability of Divergence for draws from
\((\mu_1, \mu_2, \sigma_1, \sigma_2, \rho) = (-0.025, 0.015, 0.02, 0.01, 0.2)\)
Figure 12 - Returns and Probability of Divergence for
$N(\mu, \sigma^2) = N(1.10 \times 10^{-2}, 2.74 \times 10^{-4})$
Figure 13 - Returns and Probability of Divergence for $N(\mu, \sigma^2) = N(5.5 \times 10^{-3}, 2.74 \times 10^{-4})$
REFERENCES

An Open-Source Implementation of the Critical-Line Algorithm for Portfolio Optimization

David H. Bailey 12 and Marcos Lopez de Prado 13,*

1 Lawrence Berkeley National Laboratory, 1 Cyclotron Road, Berkeley, CA 94720, USA; E-Mail: dbailey@lbl.gov
2 Department of Computer Science, University of California, Davis, CA 95616, USA
3 Hess Energy Trading Company, 1185 Avenue of the Americas, New York, NY 10036, USA

* Author to whom correspondence should be addressed; E-Mail: lpezdeprado@lbl.gov; Tel.: +1-212-536-8370; Fax: +1-203-552-8643.

Received: 29 January 2013; in revised form: 8 March 2013 / Accepted: 18 March 2013 / Published: 22 March 2013

Abstract: Portfolio optimization is one of the problems most frequently encountered by financial practitioners. The main goal of this paper is to fill a gap in the literature by providing a well-documented, step-by-step open-source implementation of Critical Line Algorithm (CLA) in scientific language. The code is implemented as a Python class object, which allows it to be imported like any other Python module, and integrated seamlessly with pre-existing code. We discuss the logic behind CLA following the algorithm's decision flow. In addition, we developed several utilities that support finding answers to recurrent practical problems. We believe this publication will offer a better alternative to financial practitioners, many of whom are currently relying on generic-purpose optimizers which often deliver suboptimal solutions. The source code discussed in this paper can be downloaded at the authors’ websites (see Appendix).

Keywords: portfolio selection; quadratic programming; portfolio optimization; constrained efficient frontier; turning point; Kuhn-Tucker conditions; risk aversion
1. Introduction

Since the work of Markowitz [1], portfolio optimization has become one of the most critical operations performed in investment management. In Modern Portfolio Theory, this operation consists in computing the Efficient Frontier, defined as the set of portfolios that yield the highest achievable mean excess return (in excess of the risk-free rate) for any given level of risk (measured in terms of standard deviation). This portfolio optimization problem receives two equivalent formulations: (i) Minimizing the portfolio's standard deviation (or variance) subject to a targeted excess return or (ii) Maximize the portfolio's excess return subject to a targeted standard deviation (or variance). Because the solution to both formulations in the same, we will focus on (i) for the remainder of the paper.

Every financial firm is constantly faced with the problem of optimizing a portfolio. Wealth managers determine optimal holdings in order to achieve a strategic goal. Mutual funds attempt to beat a benchmark by placing active bets, based on some tactical superior knowledge with regard to the future expected returns or their covariance. Multi-manager hedge funds must allocate capital to portfolio managers, based on their performance expectations. Portfolio managers allocate capital to a variety of bets, incorporating their views about the future state of the economy. Risk managers may compute the portfolio that delivers the best hedge against a position that cannot be liquidated, etc. The current size of the global asset management industry is estimated to be in excess of US $58 trillion at year-end 2011. Thus, robust portfolio optimization software is a necessity of the first order [2].

Most practitioners are routinely faced with the problem of optimizing a portfolio subject to inequality conditions (a lower and an upper bound for each portfolio weight) and an equality condition (that the weights add up to one). There is no analytic solution to this problem, and an optimization algorithm must be used. Markowitz [3,4] developed a method for computing such a solution, which he named the "critical line algorithm" or CLA. Wolfe [5] developed a Simplex version of CLA to deal with inequality constraints on linear combinations of the optimal weights. These more general constraints make Wolfe's algorithm more flexible. However, the standard portfolio optimization problem does not require them, making CLA the approach favored by most practitioners.

Given the importance of CLA, one would expect a multiplicity of software implementations in a wide range of languages. Yet, we are aware of only one published source-code for CLA, by Markowitz and Todd [6]. This is written in Excel's Microsoft Visual Basic for Applications (VBA-Excel), and it is the descendant of a previous implementation in an experimental programming language called EAS-E [7]. Steuer et al. [8] point out the inconveniences of working with VBA-Excel, most notably its low computational performance and its limitation to covariance matrices with a maximum order of $(256 \times 256)$. In addition, the VBA-Excel spreadsheet has to be manually adjusted for different problems, which prevents its industrial use (Kwak [9] explains that VBA-Excel implementations are ubiquitous in the financial world, posing a systemic risk. Citing an internal JP Morgan investigation, he mentions that a faulty Excel implementation of the Value-at-Risk model may have been partly responsible for the US $6 billion trading loss suffered by JP Morgan in 2012, popularly known as the "London whale" debacle). Hence, it would be highly convenient to have the source code of CLA in a more scientific language, such as C++ or Python.

A number of authors seem to have implemented CLA in several languages, however their code does not appear to be publicly available. For example, Hirschberger et al. [10] mention their implementation
of CLA in Java. These authors state: "For researchers intending to investigate mid- to large-scale portfolio selection, good, inexpensive and understandable quadratic parametric programming software, capable of computing the efficient frontiers of problems with up to two thousand securities without simplifications to the covariance matrix, is hardly known to be available anywhere". This insightful paper discusses the computational efficiency of their CLA implementation, and makes valuable contributions towards a better understanding of CLA's properties. However, it does not provide the source code on which their calculations were based.

Niedermayer et al. [11] implemented CLA in Fortran 90, and concluded that their algorithm was faster than Steuer et al. [8] by a factor of 10,000, on a universe of 2000 assets. These authors also state that "no publicly available software package exists that computes the entire constrained minimum variance frontier". Like other authors, they discuss the performance of their implementation, without providing the actual source code.

To our knowledge, CLA is the only algorithm specifically designed for inequality-constrained portfolio optimization problems, which guarantees that the exact solution is found after a given number of iterations. Furthermore, CLA does not only compute a single portfolio, but it derives the entire efficient frontier. In contrast, gradient-based algorithms will depend on a seed vector, may converge to a local optimum, are very sensitive to boundary constraints, and require a separate run for each member of the efficient frontier. The Scipy library offers an optimization module called optimize, which bears five constrained optimization algorithms: The Broyden-Fletcher-Goldfarb-Shanno method (BFGS), the Truncated-Newton method (TNC), the Constrained Optimization by Linear Approximation method (COBYLA), the Sequential Least Squares Programming method (SLSQP) and the Non-Negative Least Squares solver (NNLS). Of those, BFGS and TNC are gradient-based and typically fail because they reach a boundary. COBYLA is extremely inefficient in quadratic problems, and is prone to deliver a solution outside the feasibility region defined by the constraints. NNLS does not cope with inequality constraints, and SLSQP may reach a local optimum close to the original seed provided. Popular commercial optimizers include AMPL, NAG, TOMLAB and Solver. To our knowledge, none of them provide a CLA implementation. Barra offers a product called Optimizer, which "incorporates proprietary solvers developed in-house by MSCI's optimization research team". The details of these algorithms are kept confidential, and documentation is only offered to customers, so we have not been able to determine whether this product includes an implementation of CLA. The lack of publicly available CLA software, commercially or open-source, means that most researchers, practitioners and financial firms are resorting to generic linear or quadratic programming algorithms that have not been specifically designed to solve the constrained portfolio optimization problem, and will often return suboptimal solutions. This is quite astounding, because as we said most financial practitioners face this problem with relatively high frequency.

The main goal of this paper is to fill this gap in the literature by providing a well-documented, step-by-step open-source implementation of CLA in a scientific language (All code in this paper is provided "as is", and contributed to the academic community for non-business purposes only, under a GNU-GPL license. Users explicitly renounce to any claim against the authors. The authors retain the commercial rights of any for-profit application of this software, which must be pre-authorized in written by the authors). We have chosen Python because it is free, open-source, and it is widely used in the scientific community in general, and has more than 26,000 extension modules currently available
(To be precise, our code has been developed using the EPD 7.3 product (Enthought Python Distribution), which efficiently integrates all the necessary scientific libraries). An additional reason is that the Python language stresses readability. Its "pseudocode" appearance makes it a good choice for discussion in an academic paper. Because the procedure published in [11] seems to be the most numerically efficient, our implementation will follow closely their analysis. We have solved a wide range of problems using our code, and ensured that our results exactly match those of the VBA-Excel implementation in [6].

CLA was developed by Harry Markowitz to optimize general quadratic functions subject to linear inequality constraints. CLA solves any portfolio optimization problem that can be represented in such terms, like the standard Efficient Frontier problem. The posterior mean and posterior covariance derived by Black-Litterman [12] also lead to a quadratic programming problem, thus CLA is also a useful tool in that Bayesian framework. However, the reader should be aware of portfolio optimization problems that cannot be represented in quadratic form, and therefore cannot be solved by CLA. For example, the authors of this paper introduced in [13] a Sharpe ratio Efficient Frontier framework that deals with moments higher than 2, and thus has not a quadratic representation. For that particular problem, the authors have derived a specific optimization algorithm, which takes skewness and kurtosis into account.

The rest of the paper is organized as follows: Section 2 presents the quadratic programming problem we solve by using the CLA. Readers familiar with this subject can go directly to Section 3, where we will discuss our implementation of CLA in a class object. Section 4 expands the code by adding a few utilities. Section 5 illustrates the use of CLA with a numerical example. Section 6 summarizes our conclusions. Results can be validated using the Python code in the Appendix.

2. The Problem

Consider an investment universe of $n$ assets with observations characterized by a $(n \times l)$ vector of means $\mu$ and a $(n \times n)$ positive definite covariance matrix $\Sigma$. The mean vector $\mu$ and covariance matrix $\Sigma$ are computed on time-homogeneous invariants, \textit{i.e.}, phenomena that repeat themselves identically throughout history regardless of the reference time at which an observation is made. Time-homogeneity is a key property that our observations must satisfy, so that we do not need to re-estimate $\mu, \Sigma$ for a sufficiently long period of time. For instance, compounded returns are generally accepted as good time-homogeneous invariants for equity, commodity and foreign-exchange products. In the case of fixed income products, changes in the yield-to-maturity are typically used. In the case of derivatives, changes in the rolling forward at-the-money implied volatility are usually considered a good time-homogeneous invariant (see [14] for a comprehensive discussion of this subject). The numerical example discussed in Section 5 is based on equity products, hence the mean vector $\mu$ and covariance matrix $\Sigma$ are computed on compounded returns.

Following [11], we solve a quadratic programming problem subject to linear constraints in inequalities and one linear constraint in equality, and so we need some nomenclature to cover those inputs:

- $N = \{1, 2, \ldots, n\}$ is a set of indices that number the investment universe.
- $\omega$ is the $(n \times l)$ vector of asset weights, which is our optimization variable.
is the (nxl) vector of lower bounds, with \( \omega_i \geq l_i, \forall i \in N \).
• \( u \) is the (nxl) vector of upper bounds, with \( \omega_i \leq u_i, \forall i \in N \).
• \( F \subseteq N \) is the subset of free assets, where \( l_i < \omega_i < u_i \). In words, free assets are those that do not lie on their respective boundaries. \( F \) has length \( 1 \leq k \leq n \).
• \( B \subseteq N \) is the subset of weights that lie on one of the bounds. By definition, \( B \cup F = N \).

Accordingly, we can construct \( \mu \) and \( \Sigma \) as a block array and matrix.

\[
\Sigma = \begin{bmatrix} \Sigma_F & \Sigma_{FB} \\ \Sigma_{BF} & \Sigma_B \end{bmatrix}, \quad \mu = \begin{bmatrix} \mu_F \\ \mu_B \end{bmatrix}, \quad \omega = \begin{bmatrix} \omega_F \\ \omega_B \end{bmatrix}
\]

(1)

where \( \Sigma_F \) denotes the \((k \times k)\) covariance matrix among free assets, \( \Sigma_B \) the \((n - k)\times(n - k)\) covariance matrix among assets lying on a boundary condition, and \( \Sigma_{FB} \) the \((k \times (n - k))\) covariance between elements of \( F \) and \( B \), which obviously is equal to \( \Sigma_{BF} \) (the transpose of \( \Sigma_{BF} \)) since \( \Sigma \) is symmetric. Similarly, \( \mu_F \) is the \((k \times l)\) vector of means associated with \( F \), \( \mu_B \) is the \((n - k)\times l\) vector of means associated with \( B \), \( \omega_F \) is the \((k \times l)\) vector of weights associated with \( F \), \( \omega_B \) is the \((n - k)\times l\) vector of weights associated with \( B \).

The solution to the unconstrained problem ("Unconstrained problem" is a bit of a misnomer, because this problem indeed contains two linear equality constraints: Full investment (the weights add up to one) and target portfolio mean. What is meant is to indicate that no specific constraints are imposed on individual weights) consists in minimizing the Lagrange function with respect to the vector of weights \( \omega \) and the multipliers \( \gamma \) and \( \lambda \):

\[
i \left[ \omega, \gamma, \lambda \right] = \frac{1}{2} \omega' \Sigma \omega - \gamma (\omega' l_n - 1) - \lambda (\omega' \mu - \mu_p)
\]

(2)

where \( l_n \) is the \((n \times l)\) vector of ones and \( \mu_p \) is the targeted excess return. The method of Lagrange multipliers applies first order necessary conditions on each weight and Lagrange multiplier, leading to a linear system of \( n + 2 \) conditions. See [14] for an analytical solution to this problem.

As the constrained problem involves conditions in inequalities, the method of Lagrange multipliers cannot be used. One option is to apply Karush-Kuhn-Tucker conditions. Alternatively, we can "divide and conquer" the constrained problem, by translating it into a series of unconstrained problems. The key concept is that of turning point. A solution vector \( \omega^* \) is a turning point if in its vicinity there is another solution vector with different free assets. This is important because in those regions of the solution space away from turning points the inequality constraints are effectively irrelevant with respect to the free assets. In other words, between any two turning points, the constrained solution reduces to solving the following unconstrained problem on the free assets.

\[
i \left[ \omega, \gamma, \lambda \right] = -\frac{1}{2} \omega_F' \Sigma_F \omega_F + \frac{1}{2} \omega_F' \Sigma_{FB} \omega_B + \frac{1}{2} \omega_B' \Sigma_{BF} \omega_F + \frac{1}{2} \omega_B' \Sigma_B \omega_B - \gamma (\omega_F' 1_F + \omega_B' 1_B - 1) - \lambda (\omega_F' \mu_F + \omega_B' \mu_B - \mu_p)
\]

(3)

where \( \omega_B \) is known and does not change between turning points. Markowitz [3] focused his effort in computing the optimal portfolio at each turning point is because the efficient frontier can be simply derived as a convex combination between any two neighbor turning points (In the unconstrained case, this is sometimes referred to as the "two mutual funds theorem". In the constrained case, it still holds between any two neighbor turning points, because as we argued earlier, between them the constrained
problem reduces to an unconstrained problem on the free assets). Hence, the only remaining challenge is the determination of the turning points, to which end we dedicate the following section.

3. The Solution

We have implemented CLA as a class object in Python programming language. The only external library needed for this core functionality is Numpy, which in our code we instantiate with the shorthand `np`. The class is initialized in Snippet 1. The inputs are:

- *mean*: The \((nxl)\) vector of means.
- *covar*: The \((nxn)\) covariance matrix.
- *IB*: The \((nxl)\) vector that sets the lower boundaries for each weight.
- *uB*: The \((nxl)\) vector that sets the upper boundaries for each weight.

Implied is the constraint that the weights will add up to one. The class object will contain four lists of outputs:

- *w*: A list with the \((nxl)\) vector of weights at each turning point.
- *\(\lambda\)*: The value of \(\lambda\) at each turning point.
- *\(\gamma\)*: The value of \(\gamma\) at each turning point.
- *\(F\)*: For each turning point, a list of elements that constitute \(F\).

**Snippet 1.** CLA initialization.

```python
class CLA:
    def __init__(self, mean, covar, IB, uB):
        # Initialize the class
        self.mean = mean
        self.covar = covar
        self.IB = IB
        self.uB = uB
        self.w = []  # solution
        self.l = []  # lambdas
        self.g = []  # gammas
        self.f = []  # free weights
```

The key insight behind Markowitz’s CLA is to find first the turning point associated with the highest expected return, and then compute the sequence of turning points, each with a lower expected return than the previous. That first turning point consists in the smallest subset of assets with highest return such that the sum of their upper boundaries equals or exceeds one. We have implemented this search for the first turning point through a structured array. A structured array is a Numpy object that, among other operations, can be sorted in a way that changes are tracked. We populate the structured array with items from the input *mean*, assigning to each a sequential *id* index. Then we sort the structured array in descending order. This gives us a sequence for searching for the first free asset. All weights are initially set to their lower bounds, and following the sequence from the previous step, we
move those weights from the lower to the upper bound until the sum of weights exceeds one. The last iterated weight is then reduced to comply with the constraint that the sum of weights equals one. This last weight is the first free asset, and the resulting vector of weights the first turning point. See Snippet 2 for the actual implementation of this initialization operation.

**Snippet 2. Algorithm initialization.**

```python
def initAlgo(self):
    # Initialize the algo
    #1) Form structured array
    a = np.zeros((self.mean.shape[0]), dtype=[('id', int), ('mu', float)])
    b = [self.mean[i][0] for i in range(self.mean.shape[0])] # dump array into list
    a[:,] = zip(range(self.mean.shape[0]), b) # fill structured array
    #2) Sort structured array
    b = np.sort(a, order='mu')
    #3) First free weight
    i,w = b.shape[0], np.copy(self.fB)
    while sum(w) < 1:
        i+=1
        w[b[i][0]] = self.mub[b[i][0]]
        w[b[i][0]] += 1 - sum(w)
    return [b[i][0], w]
```

The transition from one turning point to the next requires that one element is either added to or removed from the subset of free assets, $F$. Because $\lambda$ and $\omega\mu$ are linearly and positively related, this means that each subsequent turning point will lead to a lower value for $\lambda$. This recursion of adding or removing one asset from $F$ continues until the algorithm determines that the optimal expected return cannot be further reduced. In the first run of this iteration, the choice is simple: $F$ has been initialized with one asset, and the only option is to add another one ($F$ cannot be an empty set, or there would be no optimization). Snippet 3 performs this task.

**Snippet 3. Determining which asset could be added to $F$.**

```python
#2) case b): Free one bounded weight
l_out = None
if len(f) < self.mean.shape[0]:
b = self.getB(f)
for i in b:
covarF, covarFB, meanF, wB = self.getMatrices(f + [i])
covarF_inv = np.linalg.inv(covarF)
l bi = self.computeLambda(covarF_inv, covarFB, meanF, wB, meanF.shape[0] - 1, \self.w[-1][i])
if (self.l[-1] == None or l < self.l[-1]) and l > l_out:
l_out, i_out = l,i
```

097
In this part of the code, we search within \( B \) for a candidate asset \( i \) to be added to \( F \). That search only makes sense if \( B \) is not an empty set, hence the first if. Because \( F \) and \( B \) are complementary sets, we only need to keep track of one of them. In the code, we always derive \( B \) from \( F \), thanks to the functions \texttt{getB} and \texttt{diffLists}, detailed in Snippet 4.

**Snippet 4.** Deriving \( B \) from \( F \).

```python
def getB(self, f):
    return self.diffLists(range(self.mean.shape[0]), f)
def diffLists(self, list1, list2):
    return list(set(list1) - set(list2))
```

Snippet 3 invokes a function called \texttt{getMatrices}. This function prepares the necessary matrices to determine the value of \( \lambda \) associated with adding each candidate \( i \) to \( F \). In order to do that, it needs to reduce a matrix to a collection of columns and rows, which is accomplished by the function \texttt{reduceMatrix}. Snippet 5 details these two functions.

**Snippet 5.** Preparing \( \Sigma_\rho, \Sigma_{\rho B}, \mu_\rho, \omega_B \).

```python
def getMatrices(self, f):
    # Slice covarF, covarFB, covarB, meanF, meanB, wF, wB
    covarF = self.reduceMatrix(self.covar, f, f)
    meanF = self.reduceMatrix(self.mean, f, [0])
    b = self.getB(f)
    covarFB = self.reduceMatrix(self.covar, f, b)
    wB = self.reduceMatrix(self.w[-1], b, [0])
    return covarF, covarFB, meanF, wB

def reduceMatrix(self, matrix, listX, listY):
    # Reduce a matrix to the provided list of rows and columns
    if len(listX) == 0 or len(listY) == 0:return
    matrix_ = matrix[:, listY[0]:listY[0] + 1]
    for i in listY[1:]:
        a = matrix[:, i:i + 1]
        matrix_ = np.append(matrix_, a, 1)
    matrix_ = matrix_[listX[0]:listX[0] + 1, :]
    for i in listX[1:]:
        a = matrix_[i:i+1, :]
        matrix_ = np.append(matrix_, a, 0)
    return matrix_
```
Using the matrices provided by the function getMatrices, λ can be computed as:

\[
\lambda = \frac{1}{C} \left[ (1 - 1_n \cdot \omega_B + 1_k \Sigma_F^{-1} \Sigma_{FB} \omega_B) (\Sigma_F^{-1} \Sigma_k) - (1_k \Sigma_F^{-1} \Sigma_k) (b_i + (\Sigma_F^{-1} \Sigma_{FB} \omega_B)) \right]
\]

with

\[
C = -(1_k \Sigma_F^{-1} \Sigma_k) (\Sigma_F^{-1} \mu_F) + (1_k \Sigma_F^{-1} \Sigma_k) (\Sigma_F^{-1} \Sigma_k)
\]

\[
b_i = \begin{cases} 
  u_i & \text{if } C_i > 0 \\
  l_i & \text{if } C_i < 0 
\end{cases}
\]

A proof of these expressions can be found in [11]. Equation (4) is implemented in function computeLambda, which is shown is Snippet 6. We have computed some intermediate variables, which can be re-used at various points in order to accelerate the calculations. With the value of λ, this function also returns b which we will need in Snippet 7.

**Snippet 6.** Computing λ.

```python
def computeLambda(self, covarF_inv, covarFB, meanF, wB, i, bi):
    #1) C
    onesF = np.ones(meanF.shape)
    c1 = np.dot(np.dot(onesF.T, covarF_inv), onesF)
    c2 = np.dot(covarF_inv, meanF)
    c3 = np.dot(np.dot(onesF.T, covarF_inv), meanF)
    c4 = np.dot(covarF_inv, onesF)
    c = -c1*c2[i] + c3*c4[i]
    if c == 0:return
    #2) b
    if type(bi) == list:bi = self.computeBt(c, bi)
    #3) Lambda
    if wB == None:
        # All free assets
        return float((c4[i] - c1*bi)/c), bi
    else:
        onesB = np.ones(wB.shape)
        l1 = np.dot(onesB.T, wB)
        l2 = np.dot(covarF_inv, covarFB)
        l3 = np.dot(l2, wB)
        l2 = np.dot(onesF.T, l3)
        return float(((1 - l1 + l2)*c4[i] - c1*(bi + l3[i])/c), bi)
```
Snippet 7. Deciding between the two alternative actions.

```python
#4) decide lambda
    if l_in > l_out:
        self.l.append(l_in)
        f.remove(i_in)
        w[i_in] = bi_in # set value at the correct boundary
    else:
        self.l.append(l_out)
        f.append(i_out)
    covarF, covarFB, meanF, wB = self.getMatrices(f)
    covarF_inv = np.linalg.inv(covarF)
```

In Snippet 3, we saw that the value of $\lambda$ which results from each candidate $i$ is stored in variable $i$. Among those values of $i$, we find the maximum, store it as $l_out$, and denote as $i_out$ our candidate to become free. This is only a candidate for addition into $F$, because before making that decision we need to consider the possibility that one item is removed from $F$, as follows.

After the first run of this iteration, it is also conceivable that one asset in $F$ moves to one of its boundaries. Should that be the case, Snippet 8 determines which asset would do so. Similarly to the addition case, we search for the candidate that, after removal, maximizes $\lambda$ (or to be more precise, minimizes the reduction in $\lambda$, since we know that $\lambda$ becomes smaller at each iteration). We store our candidate for removal in the variable $i_in$, and the associated $\lambda$ in the variable $l_in$.

Snippet 8. Determining which asset could be removed from $F$.

```python
#1) case a): Bound one free weight
    l_in = None
    if len(f) > 1:
        covarF, covarFB, meanF, wB = self.getMatrices(f)
        covarF_inv = np.linalg.inv(covarF)
        j = 0
        for i in f:
            l, bi = self.computeLambda(covarF_inv, covarFB, meanF, wB, j, [self.LB[i], self.UB[i]])
            if l > l_in: l_in, i_in, bi_in = l, i, bi
            j += 1
```

All auxiliary functions used in Snippet 8 have been discussed earlier. At this point, we must take one of two alternative actions: We can either add one asset to $F$, or we can remove one asset from $F$. The answer is whichever gives the greater value of $\lambda$, as shown in Snippet 7. Recall that in Snippet 6 the function `computeLambda` had also returned $b_1$. This is the boundary value that we will assign to a formerly free weight (now a member of $B$).
We finally know how to modify $F$ in order to compute the next turning point, but we still need to compute the actual turning point that results from that action. Given the new value of $\lambda$, we can derive the value of $\gamma$, which together determine the value of the free weights in the next turning point, $\omega_p$.

$$7 = -\lambda \left( \frac{1}{k_F} \Sigma_F^{-1} \mu_F \right) - 1 \left( \frac{1 - \gamma \Sigma_F^{-1} \Sigma_{FB} \omega_B}{1 \left( \frac{1}{k_F} \Sigma_F^{-1} \mu_F \right)} \right)$$

$$\omega_F = -\Sigma_F^{-1} \Sigma_{FB} \omega_B + \gamma \Sigma_F^{-1} 1_k + \lambda \Sigma_F^{-1} \mu_F$$  \hspace{1cm} (5)

Equation (5) is evaluated by the function $\text{computeW}$, which is detailed in Snippet 9.

**Snippet 9.** Computing the turning point associated with the new $F$.

```python
def computeW(self, covarF_inv, covarFB, meanF, wB):
    #1) compute gamma
    onesF = np.ones(meanF.shape)
    g1 = np.dot(np.dot(onesF, meanF), meanF)
    g2 = np.dot(np.dot(onesF, meanF), onesF)
    if wB == None:
        g, w1 = float(-self[[-1]]*g1/g2 + 1/g2), 0
    else:
        onesB = np.ones(wB.shape)
        g3 = np.dot(onesB.T, wB)
        g4 = np.dot(covarF_inv, covarFB)
        w1 = np.dot(g4, wB)
        g4 = np.dot(onesF.T, w1)
        g = float(-self[[-1]]*g1/g2 + (1 - g3 + g4)/g2)
    #2) compute weights
    w2 = np.dot(covarF_inv, onesF)
    w3 = np.dot(covarF_inv, meanF)
    return -w1 + g*w2 + self[[-1]]*w3, g
```

Again, we are computing some intermediate variables for the purpose of re-using them, thus speeding up calculations. We are finally ready to store these results, as shown in Snippet 10. The last line incorporates the exit condition, which is satisfied when $\lambda = 0$, as it cannot be further reduced.

**Snippet 10.** Computing and storing a new solution.

```python
#5) compute solution vector
    wF, g = self.computeW(covarF_inv, covarFB, meanF, wB)
    for i in range(len(f)):w[i] = wF[i]
    self.w.append(np.copy(w))  # store solution
    self.g.append(f)
    self.f.append(f[i])
    if self./[-1] = = 0: break
```
The algorithm then loops back through Snippets 3-10, until no candidates are left or the highest lambda we can get is negative (We also depart here from [11], who keep searching for turning points with negative $\lambda$). When that occurs, we are ready to compute the last solution, as follows. Each turning point is a minimum variance solution subject to a certain target portfolio mean, but at some point we must also explicitly add the global minimum variance solution, denoted Minimum Variance Portfolio. This possibility is not considered by [11], but it is certainly a necessary portfolio computed by [6]. The analysis would be incomplete without it, because we need the left extreme of the frontier, so that we can combine it with the last computed turning point. When the next $\lambda$ is negative or it cannot be computed, no additional turning points can be derived. It is at that point that we must compute the Minimum Variance portfolio, which is characterized by $\lambda = 0$ and a vector $\mu_{\text{F}}$ of zeroes. Snippet 11 prepares the relevant variables, so that the function $\text{compute}\ W$ returns that last solution.


```python
if (l_in == None or l_in < 0) and (l_out == None or l_out < 0):
    #3) compute minimum variance solution
    self.l.append(0)
    covarF, covarFB, meanF, wB = self.getMatrices(f)
    covarF_inv = np.linalg.inv(covarF)
    meanF = np.zeros(meanF.shape)
```

4. A Few Utilities

The CLA class discussed in Section 3 computes all turning points plus the global Minimum Variance portfolio. This constitutes the entire set of solutions, and from that perspective, Section 3 presented an integral implementation of Markowitz's CLA algorithm. We think that this functionality can be complemented with a few additional methods designed to address problems typically faced by practitioners.

4.1. Search for the Minimum Variance Portfolio

The Minimum Variance portfolio is the leftmost portfolio of the constrained efficient frontier. Even if it did not coincide with a turning point, we appended it to $\text{self.}\ w$, so that we can compute the segment of efficient frontier between the Minimum Variance portfolio and the last computed turning point. Snippet 12 exemplifies a simple procedure to retrieve this portfolio: For each solution stored, it computes the variance associated with it. Among all those variances, it returns the squared root of the minimum (the standard deviation), as well as the portfolio that produced it. This portfolio coincides with the solution computed in Snippet 11.
Snipet 12. The search for the Minimum Variance portfolio.

def getMinVar(self):
    # Get the minimum variance solution
    var = []
    for w in self.w:
        a = np.dot(np.dot(w.T, self.covar), w)
        var.append(a)
    return min(var)**.5, self.w[var.index(min(var))]

4.2. Search for the Maximum Sharpe Ratio Portfolio

The turning point with the maximum Sharpe ratio does not necessarily coincide with the maximum Sharpe ratio portfolio. Although we have not explicitly computed the maximum Sharpe ratio portfolio yet, we have the building blocks needed to construct it. Every two neighbor turning points define a segment of the efficient frontier. The weights that form each result segment from the convex combination of the turning points at its edges.

For a $a \in [0, 1]$, $\omega = a\omega_0 + (1-a)\omega_1$ gives the portfolios associated with the segment of the efficient frontier delimited by $\omega_0$ and $\omega_1$. The Sharpe ratio is a strongly unimodal function of $\lambda$ (see [15] for a proof). We know that $\lambda$ is a strictly monotonic function of $\alpha$, because $\lambda$ cannot increase as we transition from $\omega_0$ to $\omega_1$. The conclusion is that the Sharpe ratio function is also a strongly unimodal function of $\alpha$. This means that we can use the Golden Section algorithm to find the maximum Sharpe ratio portfolio within the appropriate segment (The Golden Section search is a numerical algorithm for finding the minimum or maximum of a real-valued and strictly unimodal function. This is accomplished by sequentially narrowing the range of values inside which the minimum or maximum exists. Avriel [16] introduced this technique, and [17] proved its optimality). Snippet 13 shows an implementation of such algorithm. The goldenSection function receives as arguments:

- obj: The objective function on which the extreme will be found.
- a: The leftmost extreme of search.
- b: The rightmost extreme of search.
- **kargs: Keyworded variable-length argument list.


def goldenSection(self, obj, a, b, **kargs):
    # Golden section method. Maximum if kargs['minimum'] == False is passed
    from math import log,ceil
    tol, sign, args = 1.0e-9, 1, None
    if 'minimum' in kargs and kargs['minimum'] == False: sign = -1
    if 'args' in kargs: args = kargs['args']
    numIter = int(ceil(-2.078087*log(tol/abs(a - b))))
    r = 0.618033989
The extremes of the search are delimited by the neighbor turning points, which we pass as a keyworded variable length argument \textit{(kargs)}. \textit{kargs} is a dictionary composed of two optional arguments: "\textit{minimum}\" and "\textit{args}". Our implementation of the Golden Section algorithm searches for a minimum by default, however it will search for a maximum when the user passes the optional argument "\textit{minimum}\" with value \textit{False}. "\textit{args}\" contains a non-keyworded variable-length argument, which (if present) is passed to the objective function \textit{obj}. This approach allows us to pass as many arguments as the objective function \textit{obj} may need in other applications. Note that, for this particular utility, we have imported two additional functions from Python's \textit{math} library: \textit{log} and \textit{ceil}.

We ignore which segment contains the portfolio that delivers the global maximum Sharpe ratio. We will compute the local maximum of the Sharpe ratio function for each segment, and store the output. Then, the global maximum will be determined by comparing those local optima. This operation is conducted by the function \textit{getMaxSR} in Snippet 14. \textit{evalSR} is the objective function (\textit{obj}) which we pass to the \textit{goldenSection} routine, in order to evaluate the Sharpe ratio at various steps between \(\omega_0\) and \(\omega_1\). We are searching for a maximum between those two turning points, and so we set \textit{kargs} = \{'minimum':\textit{False}, 'args':(\textit{w0}, \textit{wl})\}.

\begin{verbatim}
c = 1.0 - r
# Initialize
x1 = r*a + c*b; x2 = c*a + r*b
f1 = sign*obj(x1, *args); f2 = sign*obj(x2, *args)
# Loop
for i in range(numIter):
  if f1 > f2:
    a = x1
    x1 = x2; f1 = f2
    x2 = c*a + r*b; f2 = sign*obj(x2, *args)
  else:
    b = x2
    x2 = x1; f2 = f1
    x1 = r*a + c*b; f1 = sign*obj(x1, *args)
if f1 < f2:return x1, sign*f1
else:return x2, sign*f2
\end{verbatim}

\textbf{Snippet 14.} The search for Maximum Sharpe ratio portfolio.

\begin{verbatim}
def getMaxSR(self):
# Get the max Sharpe ratio portfolio
#1) Compute the local max SR portfolio between any two neighbor turning points
w sr = [], []
for i in range(len(self.w) - 1):
  w0 = np.copy(self.w[i])
  w1 = np.copy(self.w[i + 1])
  kargs = {'minimum':False, 'args':(w0, wl)}
  evalSR = self.evalSR(kargs)
  w sr.extend([w0, w1, evalSR])
return w sr
\end{verbatim}
4.3. Computing the Efficient Frontier

As argued earlier, we can compute the various segments of the efficient frontier as a convex combination between any two neighbor turning points. This calculation is carried out by the function `efFrontier` in Snippet 15. We use the Numpy function `linspace` to uniformly partition the unit segment, excluding the value 1. The reason for this exclusion is, the end of one convex combination coincides with the beginning of another, and so the uniform partition should contain that value 0 or 1, but not both. When we reach the final pair of turning points, we include the value 1 in the uniform partition, because this is the last iteration and the resulting portfolio will not be redundant. `efFrontier` outputs three lists: Means, Standard Deviations and the associated portfolio weights. The number of items in each of these lists is determined by the argument `points`.

**Snippet 15. Computing the Efficient Frontier.**

```python
import numpy as np

def efFrontier(self, points):
    # Get the efficient frontier
    mu, sigma, weights = [], [], []
    a = np.linspace(0, 1, points/len(self.w))[:-1] # remove the 1, to avoid duplications
    b = range(len(self.w) - 1)
    for i in b:
        w0, w1 = self.w[i], self.w[i + 1]
        if i == b[-1]: a = np.linspace(0, 1, points/len(self.w)) # include the 1 in the last iteration
        for j in a:
            w = w1*j + (1 - j)*w0
            weights.append(np.copy(w))
            mu.append(np.dot(w.T, self.mean)[0, 0])
            sigma.append(np.dot(np.dot(w.T, self.covar), w)[0, 0]**.5)
        return mu, sigma, weights
```
5. A Numerical Example

We will run our Python implementation of CLA on the same example that accompanies the VBA-Excel implementation by [6]. Table 1 provides the vector of expected returns and the covariance matrix of returns. We set as boundary conditions that $0 \leq \omega_i \leq 1, \forall i = 1, ..., n$, and the implicit condition that $\sum_{i=1}^{n} \omega_i = 1$.

Table 1. Covariance Matrix, Vector of Expected Returns and boundary conditions.

<table>
<thead>
<tr>
<th>L Bound</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>U Bound</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Mean</td>
<td>1.175</td>
<td>1.19</td>
<td>0.396</td>
<td>1.12</td>
<td>0.346</td>
<td>0.679</td>
<td>0.089</td>
<td>0.73</td>
<td>0.481</td>
<td>1.08</td>
</tr>
<tr>
<td>Cov</td>
<td>0.4075516</td>
<td>0.0317584</td>
<td>0.0313639</td>
<td>0.9063047</td>
<td>0.0518392</td>
<td>0.0313639</td>
<td>0.0313639</td>
<td>0.194909</td>
<td>0.0313639</td>
<td>0.0313639</td>
</tr>
</tbody>
</table>

Snippet 16 shows a simple example of how to use the CLA class. We have stored the data in a csv file, with the following structure:

- Row 1: Headers
- Row 2: Mean vector
- Row 3: Lower bounds
- Row 4: Upper bounds
- Row 5 and successive: Covariance matrix

Snippet 16. Using the CLA class.

```python
def main():
    import numpy as np
    import CLA
    #1) Path
    path='H:/PROJECTS/Data/CLA_Data.csv'
    #2) Load data, set seed
    headers=open(path,'r').readline().split(';')[:-1]
data=np.genfromtxt(path,delimiter=';',skip_header=1) # load as numpy array
mean=np.array(data[:1]).T
lB=np.array(data[1:2]).T
uB=np.array(data[2:3]).T
covar=np.array(data[3:]);
```
The key lines of code are:

- **cla = CLA.CLA(mean, covar, IB, uB)**: This creates a CLA object named `cla`, with the input parameters read from the csv file.
- **cla.solve()**: This runs the `solve` method within the CLA class (see the Appendix), which comprises all the Snippets listed in Section 3 (Snippets 1-1).

Once the `cla.solve()` method has been run, results can be accessed easily:

- **cla.w** contains a list of all turning points.
- **cla.l** and **cla.g** respectively contain the values of $\lambda$ and $\gamma$ for every turning point.
- **cla.f** contains the composition of $F$ used to compute every turning point.

Table 2 reports these outputs for our particular example. Note that sometimes an asset may become free, and yet the turning point has the weight for that asset resting precisely at the same boundary it became free from. In that case, the solution may seem repeated, when in fact what is happening is that the same portfolio is the result of two different $F$ sets.
Table 2. Return, Risk, \( \lambda \) and composition of the 10 turning points.

<table>
<thead>
<tr>
<th>CP Num</th>
<th>Return</th>
<th>Risk</th>
<th>Lambda</th>
<th>X(1)</th>
<th>X(2)</th>
<th>X(3)</th>
<th>X(4)</th>
<th>X(5)</th>
<th>X(6)</th>
<th>X(7)</th>
<th>X(8)</th>
<th>X(9)</th>
<th>X(10)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.190</td>
<td>0.952</td>
<td>58.303</td>
<td>0.000</td>
<td>1.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>2</td>
<td>1.180</td>
<td>0.546</td>
<td>4.174</td>
<td>0.649</td>
<td>0.351</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>3</td>
<td>1.160</td>
<td>0.417</td>
<td>1.946</td>
<td>0.434</td>
<td>0.231</td>
<td>0.000</td>
<td>0.335</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>4</td>
<td>1.111</td>
<td>0.267</td>
<td>0.165</td>
<td>0.127</td>
<td>0.072</td>
<td>0.000</td>
<td>0.281</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.520</td>
</tr>
<tr>
<td>5</td>
<td>1.108</td>
<td>0.265</td>
<td>0.147</td>
<td>0.123</td>
<td>0.070</td>
<td>0.000</td>
<td>0.279</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.006</td>
<td>0.000</td>
<td>0.521</td>
</tr>
<tr>
<td>6</td>
<td>1.022</td>
<td>0.230</td>
<td>0.056</td>
<td>0.087</td>
<td>0.050</td>
<td>0.000</td>
<td>0.224</td>
<td>0.000</td>
<td>0.174</td>
<td>0.000</td>
<td>0.030</td>
<td>0.000</td>
<td>0.435</td>
</tr>
<tr>
<td>7</td>
<td>1.015</td>
<td>0.228</td>
<td>0.052</td>
<td>0.085</td>
<td>0.049</td>
<td>0.000</td>
<td>0.220</td>
<td>0.000</td>
<td>0.180</td>
<td>0.000</td>
<td>0.031</td>
<td>0.006</td>
<td>0.429</td>
</tr>
<tr>
<td>8</td>
<td>0.973</td>
<td>0.220</td>
<td>0.037</td>
<td>0.074</td>
<td>0.044</td>
<td>0.000</td>
<td>0.199</td>
<td>0.026</td>
<td>0.198</td>
<td>0.000</td>
<td>0.033</td>
<td>0.028</td>
<td>0.398</td>
</tr>
<tr>
<td>9</td>
<td>0.950</td>
<td>0.216</td>
<td>0.031</td>
<td>0.068</td>
<td>0.041</td>
<td>0.015</td>
<td>0.188</td>
<td>0.034</td>
<td>0.202</td>
<td>0.000</td>
<td>0.034</td>
<td>0.034</td>
<td>0.383</td>
</tr>
<tr>
<td>10</td>
<td>0.803</td>
<td>0.205</td>
<td>0.000</td>
<td>0.037</td>
<td>0.027</td>
<td>0.095</td>
<td>0.126</td>
<td>0.077</td>
<td>0.219</td>
<td>0.030</td>
<td>0.036</td>
<td>0.061</td>
<td>0.292</td>
</tr>
</tbody>
</table>

We can also access the utilities described in Section 4. For instance, the instruction \( \mu, \sigma, \text{weights} = \text{cla.efFrontier}(100) \) in Snippet 16 computes 100 points of the Efficient Frontier, and plots them using the auxiliary function in Snippet 17. If the optional argument \( \text{pathChart} \) is not used, the chart is plotted, but if a path is provided, the chart is saved as a file in that destination.

Snippet 17. The efficient frontier.

def plot2D(x, y, xLabel = "", yLabel = "", title = ", pathChart = None):
    import matplotlib.pyplot as plt
    fig = plt.figure()
    ax = fig.add_subplot(1, 1, 1) # one row, one column, first plot
    ax.plot(x, y, color = 'blue')
    ax.set_xlabel(xLabel)
    ax.set_ylabel(yLabel, rotation = 90)
    plt.xticks(rotation = 'vertical')
    plt.title(title)
    if pathChart == None:
        plt.show()
    else:
        plt.savefig(pathChart)
    plt.clf() # reset plt
    return

Figure 1 plots the efficient frontier, using the \texttt{plot2D} function provided in Snippet 17. Figure 2 plots the Sharpe ratio as a function of risk. Visually, the maximum Sharpe ratio portfolio is located around a risk of 0.2, and reaches a level close to 4.5. This can be easily verified by running the instruction \( \text{sr, w}_sr = \text{cla.getMaxSRQ} \), which returns a Sharpe ratio of 4.4535 for a portfolio with risk 0.2274. Similarly, running the instruction \( \text{mv, w}_mv = \text{cla.getMinVarQ} \) reports a Minimum Variance portfolio with a risk of 0.2052.
VBA-Excel's *double* data type is based on a modified IEEE 754 specification, which offers a precision of 15 significant figures ([18]). Our Python results exactly match the outputs obtained when using the implementation of [6], to the highest accuracy offered by VBA-Excel.

6. Conclusions

Portfolio optimization is one of the problems most frequently encountered by financial practitioners. Following Markowitz [1], this operation consists in identifying the combination of assets that maximize the expected return subject to a certain risk budget. For a complete range of risk budgets, this gives rise to the concept of Efficient Frontier. This problem has an analytical solution in the absence of inequality constraints, such as lower and upper bounds for portfolio weights.

In order to cope with inequality constraints, [3,4] introduced the Critical Line Algorithm (CLA). To our knowledge, CLA is the only algorithm specifically designed for linear inequality-constrained quadratic portfolio optimization problems, which guarantees that the exact solution is found after a given number of iterations. Furthermore, CLA does not only compute a single portfolio, but it derives
the entire efficient frontier. In the context of portfolio optimization problems, this approach is clearly more adequate than generic-purpose quadratic programming algorithms.

Given these two facts that portfolio optimization is a critical task, and that CLA provides an extremely efficient solution method, one would expect a myriad of software implementations to be available. Yet, to our surprise, we are only aware of one open-source, fully documented implementation of CLA, by [6]. Unfortunately, that implementation was done in VBA-Excel. This requires manual adjustments to a spreadsheet, which inevitably narrows its applicability to small-scale problems. Consequently, we suspect that most financial practitioners are resorting to generic-purpose quadratic programming software to optimize their portfolios, which often delivers suboptimal solutions.

The main goal of this paper is to fill this gap in the literature by providing a well-documented, step-by-step open-source implementation of CLA in a scientific language. The code is implemented as a Python class object, which allows it to be imported like any other Python module, and integrated seamlessly with pre-existing code. Following the explanation provided in this paper, our class can also be easily translated to other languages, such as C, C++ or Fortran-90. We have discussed the logic behind CLA following the algorithm's decision flow. In addition, we have developed several utilities that facilitate the answering of recurrent practical problems. Our results match the output of [6] at the highest accuracy offered by Excel.

Appendix

A.1. Python Implementation of the Critical Line Algorithm

This Python class contains the entirety of the code discussed in Sections 3 and 4 of the paper. Section 4 presents an example of how to generate objects from this class. The following source code incorporates two additional functions:

- `purgeNumErrQ`: It removes turning points which violate the inequality conditions, as a result of a near-singular covariance matrix $\Sigma_F$.
- `purgeExcessQ`: It removes turning points that violate the convex hull, as a result of unnecessary drops in $\lambda$.

The purpose of these two functions is to deal with potentially ill-conditioned matrices. Should the input matrices and vectors have good numerical properties, these numerical controls would be unnecessary. Since they are not strictly a part of CLA, we did not discuss them in the paper. This source code can be downloaded at [19] or [20].

**Snippet 18.** The CLA Python class.

```python
#!/usr/bin/env python
# On 20130210, v0.2
# Critical Line Algorithm
# by MLdP <lopezdeprado@lbl.gov>
import numpy as np
```
class CLA:
    def __init__(self, mean, covar, lb, ub):
        # Initialize the class
        self.mean = mean
        self.covar = covar
        self.lb = lb
        self.ub = ub
        self.w = []  # solution
        self.l = []  # lambdas
        self.g = []  # gammas
        self.f = []  # free weights

    def solve(self):
        # Compute the turning points, free sets and weights
        f, w = self.initAlg0()
        self.w.append(np.copy(w))  # store solution
        self.l.append(None)
        self.g.append(None)
        self.f.append(f)

        while True:
            # 1) case a): Bound one free weight
            l_in = None
            if len(f) > 1:
                covarF, covarFB, meanF, wB = self.getMatrices(f)
                covarF_inv = np.linalg.inv(covarF)
                j = 0
                for i in f:
                    l, bi = self.computeLambda(covarF_inv, covarFB, meanF, wB[j, [self.lb[i], self.ub[i]]])
                    if l > l_in:
                        l_in = l
                        bi_in = i
                        j += 1
            # 2) case b): Free one bounded weight
            l_out = None
            if len(f) < self.mean.shape[0]:
                b = self.getB(f)
                for i in b:
                    covarF, covarFB, meanF, wB = self.getMatrices(f + [i])
                    covarF_inv = np.linalg.inv(covarF)
                    l, bi = self.computeLambda(covarF_inv, covarFB, meanF, wB, meanF.shape[0] - 1, \
                        self.w[-1][i])
                    if (self.f[-1] == None or l < self.f[-1]) and l > l_out:
                        l_out = l
                        i_out = i
if (l_in==None or l_in<0) and (l_out==None or l_out<0):
    #3) compute minimum variance solution
    self.l.append(0)
    covarF,covarFB,meanF,wB=self.getMatrices(f)
    covarF_inv=np.linalg.inv(covarF)
    meanF=np.zeros(meanF.shape)
else:
    #4) decide lambda
    if l_in>l_out:
        self.l.append(l_in)
        f.remove(l_in)
        w[i_in]=bi_in # set value at the correct boundary
    else:
        self.l.append(l_out)
        f.append(l_out)
        covarF,covarFB,meanF,wB=self.getMatrices(f)
    covarF_inv=np.linalg.inv(covarF)

#5) compute solution vector
wF,g=self.computeW(covarF_inv,covarFB,meanF,wB)
for i in range(len(w)):w[i]=wF[i]
self.w.append(np.copy(w)) # store solution
self.g.append(g)
self.f.append(f[i])
if self.f[-1]==0:break

#6) Purge turning points
self.purgeNumErr(t=10e-10)
self.purgeExcess()

def initAlgo(self):
    # Initialize the algo

#1) Form structured array
a=np.zeros((self.mean.shape[0]),dtype=[('id','int'),('mu','float')])
b=[self.mean[i][0] for i in range(self.mean.shape[0])] # dump array into list
a[:,1]=zip(range(self.mean.shape[0]),b) # fill structured array

#2) Sort structured array
b=np.sort(a,order='mu')

#3) First free weight
i,w=b.shape[0],np.copy(self.IB)
while sum(w)<1:
    i+=1
    w[b[i][0]]=self.uB[b[i][0]]

w[b[i][0]]+=1-sum(w)
return [b[i][0]], w
#

# def computeBi(self,c,bi):
    #if c<0:
    #    bi=b[i][0]
    #if c<0:
    #    bi=b[i][0]
    #return bi
#

# def computeW(self,covarF_inv,covarFB,meanF,wB):
#  #1) compute gamma
onesF=np.ones(meanF.shape)
    g1=np.dot(np.dot(onesF.T,covarF_inv),meanF)
g2=np.dot(np.dot(onesF.T,covarF_inv),onesF)
    if wB==None:
        g,w1=float(-self.l[-1]*g1/g2+1/g2),0
    else:
        onesB=np.ones(wB.shape)
g3=np.dot(onesB,T,wB)
g4=np.dot(covarF_inv,covarFB)
w1=np.dot(g4,wB)
g4=np.dot(covarF,T,w1)
g=float(-self.l[-1]*g1/g2+(1-g3+g4)/g2)
#
#2) compute weights
w2=np.dot(covarF_inv,onesF)
w3=np.dot(covarF_inv,meanF)
return -w1+g*w2+self.l[-1]*w3,g
#

# def computeLambda(self,covarF_inv,covarFB,meanF,wB,i,bi):
#  #1) C
onesF=np.ones(meanF.shape)
c1=np.dot(np.dot(onesF.T,covarF_inv),onesF)
c2=np.dot(covarF_inv,meanF)
c3=np.dot(onesF.T,covarF_inv),meanF)
c4=np.dot(covarF_inv,onesF)
c=-c1*c2[i]+c3*c4[i]
    if c==0:return
#2) bi
    if type(bi)==list:bi=self.computeBi(c,bi)
#3) Lambda
    if wB==None:
        # All free assets
```python
return float((c4[i]-c1*bi)/c),bi
else:
onesB=np.ones(wB.shape)
11=np.dot(nesB.T,wB)
l2=np.dot(covarF_inv,covarFB)
l3=np.dot(l2,wB)
l2=np.dot(nesF.T,l3)
return float(((1-l1+l2)*c4[i]-c1*(bi+l3[i]))/c),bi

#---------------------------------------------
def getMatrices(self,f):
    # Slice covarF,covarFB,covarB,meanF,meanB,wF,wB
    covarF=self.reduceMatrix(self.covar,f,f)
    meanF=self.reduceMatrix(self.mean,f,[0])
    b=self.getB(f)
    covarFB=self.reduceMatrix(self.covar,f,b)
    wB=self.reduceMatrix(self.w[-1],b,[0])
    return covarF,covarFB,meanF,wB

#---------------------------------------------
def getB(self,f):
    return self.diffLists(range(self.mean.shape[0]),f)

#---------------------------------------------
def diffLists(self,list1,list2):
    return list(set(list1)-set(list2))

#---------------------------------------------
def reduceMatrix(self,matrix,listX,listY):
    # Reduce a matrix to the provided list of rows and columns
    if len(listX)==0 or len(listY)==0:return
    matrix_=matrix[:,listY[0]:listY[0]+1]
    for i in listY[1:]:
        a=matrix[:,i:i+1]
        matrix_=np.append(matrix_,a,1)
        matrix__=matrix__[listX[0]:listX[0]+1,:]
        for i in listX[1:]:
            a=matrix_[i:i+1,:]
            matrix__=np.append(matrix__,a,0)
        return matrix__

#---------------------------------------------
def purgeNumErr(self, tol):
    # Purge violations of inequality constraints (associated with ill-conditioned covar matrix)
    i=0
    while True:
        if i==len(self.w):break
```
w = self.w[i]
for j in range(w.shape[0]):
        del self.w[i]
        del self.l[i]
        del self.g[i]
        del self.f[i]
        break
i += 1

#---------------------------------------------------------------------------------

def purgeExcess(self):
    # Remove violations of the convex hull
    i, repeat = 0, False
    while True:
        if repeat is False: i += 1
        if i == len(self.w) - 1: break
        w = self.w[i]
        mu = np.dot(w, T, self.mean)(0, 0)
        j, repeat = i + 1, False
        while True:
            if j == len(self.w): break
            w = self.w[j]
            mu_ = np.dot(w, T, self.mean)(0, 0)
            if mu < mu_:
                del self.w[i]
                del self.l[i]
                del self.g[i]
                del self.f[i]
                repeat = True
                break
            else:
                j += 1

#---------------------------------------------------------------------------------

def getMinVar(self):
    # Get the minimum variance solution
    var = []
    for w in self.w:
        a = np.dot(np.dot(w, T, self.covar), w)
        var.append(a)
    return min(var)**.5, self.w[var.index(min(var))]

#---------------------------------------------------------------------------------

def getMaxSR(self):
# Get the max Sharpe ratio portfolio
#1) Compute the local max SR portfolio between any two neighbor turning points
w_sr sr=[]
for i in range(len(self.w)-1):
w0=np.copy(self.w[i])
w1=np.copy(self.w[i+1])
kargs={‘minimum’:False,’args’:(w0,w1)}
a,b=self.goldenSection(self.evalSR,0,1,**kargs)
w_sr.append(a*w0+(1-a)*w1)
sr.append(b)
return max(sr),w_sr[sr.index(max(sr))]

def evalSR(self,a,w0,w1):
    # Evaluate SR of the portfolio within the convex combination
    w=a*w0+(1-a)*w1
    b=np.dot(w.T,self.mean)[0,0]
c=np.dot(np.dot(w.T,self.covar),w)[0,0]**.5
    return b/c

def goldenSection(self,obj,a,b,**kargs):
    # Golden section method. Maximum if kargs[‘minimum’]==False is passed
    from math import log,ceil
tol,sign,argv=1.0e-9,1,None
    if ‘minimum’ in kargs and kargs[‘minimum’]==False:sign=-1
    if ‘args’ in kargs:argv=kargs[‘args’]
    numIter=int(ceil(-2.078087*log(tol/abs(b-a))))
r=0.618033989
c=1.0-r
    # Initialize
    x1=r*a+c*b;x2=c*a+r*b
    f1=sign*obj(x1,*argv);f2=sign*obj(x2,*argv)
    # Loop
    for i in range(numIter):
        if f1>f2:
            a=x1
            x1=x2;f1=f2
            x2=c*a+r*b;f2=sign*obj(x2,*argv)
        else:
            b=x2
            x2=x1;f2=f1
            x1=r*a+c*b;f1=sign*obj(x1,*argv)
        if f1<=f2:return x1,sign*f1
Acknowledgments

Supported in part by the Director, Office of Computational and Technology Research, Division of Mathematical, Information, and Computational Sciences of the U.S. Department of Energy, under contract number DE-AC02-05CH11231.

We would like to thank the Editor-in-Chief of Algorithms, Kazuo Iwama (Kyoto University), as well as two anonymous referees for useful comments. We are grateful to Hess Energy Trading Company, our colleagues at CIFT (Lawrence Berkeley National Laboratory), Attilio Meucci (Kepos Capital, New York University), Riccardo Rebonato (PIMCO, University of Oxford) and Luis Viceira (Harvard Business School).

References


© 2013 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution license (http://creativecommons.Org/licenses/by/3.0/).
BALANCED BASKETS:
A NEW APPROACH TO TRADING AND HEDGING RISKS

David H. Bailey
Complex Systems Group Leader - Lawrence Berkeley National Laboratory
dhbailey@lbl.gov

Marcos Lopez de Prado
Head of Global Quantitative Research - Tudor Investment Corporation
and
Research Affiliate - Lawrence Berkeley National Laboratory
lopezdeprado@lbl.gov

The Journal of Investment Strategies, Fall 2012
http://www.risk.net/type/journal/source/journal-of-investment-strategies

We are grateful to Tudor Investment Corporation, Robert Almgren (Quantitative Brokers, NYU), Jose Blanco (UBS), Sid Browne (Guggenheim Partners), Peter Carr (Morgan Stanley, NYU), David Easley (Cornell Univ.), Matthew Foreman (Univ. of California, Irvine), Ross Garon (S.A.C. Capital Advisors), Paul Glasserman (Columbia Univ.), Robert Jarxow (Cornell Univ.), David Leinweber (Lawrence Berkeley National Laboratory), Yin Luo (Deutsche Bank), Attilio Meucci (Kepos Capital, NYU), Maureen O'Hara (Cornell Univ.), Riccardo Rebonato (PIMCO, Univ. of Oxford), Luis Viceira (HBS), as well as two anonymous referees.

Supported in part by the Director, Office of Computational and Technology Research, Division of Mathematical, Information, and Computational Sciences of the U.S. Department of Energy, under contract number DE-AC02-05CH11231.
BALANCED BASKETS:
A NEW APPROACH TO TRADING AND HEDGING RISKS

ABSTRACT

A basket is a set of instruments that are held together because its statistical profile delivers a desired goal, such as hedging or trading, which cannot be achieved through the individual constituents or even subsets of them. Multiple procedures have been proposed to compute hedging and trading baskets, among which balanced baskets have attracted significant attention in recent years. Unlike Principal Component Analysis (PCA) style of methods, balanced baskets spread risk or exposure across their constituents without requiring a change of basis. Practitioners typically prefer balanced baskets because their output can be understood in the same terms for which they have developed an intuition.

We review three methodologies for determining balanced baskets, analyze the features of their respective solutions and provide Python code for their calculation. We also introduce a new method for reducing the dimension of a covariance matrix, called Covariance Clustering, which addresses the problem of numerical ill-conditioning without requiring a change of basis.

Keywords: Trading baskets, hedging baskets, equal risk contribution, maximum diversification, subset correlation.

JEL Classifications: C01, C02, C61, D53, G11.
1. INTRODUCTION
A basket is a set of instruments that are held together because its statistical profile delivers a desired goal, such as hedging a risk or trading it, which cannot be achieved through the individual constituents or even subsets of them. Portfolio managers build trading baskets that translate their views of the markets into actual financial bets, while hedging their exposure to other risks they have no view on. Market makers build hedging baskets that allow them to offset the risk derived from undesired inventory. Quantitative researchers form hedging baskets as a mean to study, replicate or reverse-engineer the factors driving the performance of a security, portfolio or hedge fund (Jaeger (2008)).

Multiple procedures have been proposed to compute hedging and trading baskets. Balanced baskets have attracted significant attention in recent years because, unlike PCA-style methods (see Litterman and Scheinkman (1991), Moulton and Seydoux (1998), for example), they spread risk or exposure across its constituents without requiring a change of basis. A change of basis is problematic because the basket's solution is expressed in terms of the new basis (a linear combination of tradable instruments), which may not be intuitive in terms of the old basis. Practitioners typically prefer balanced baskets for this reason.

In this paper we will differentiate between the goal of hedging and the goal of trading. In the first instance, the basket is formed to reduce the investor's risk or exposure to any of its legs, or any subset of them. In the second instance, the investor would like to acquire risk or exposure to each and every of its legs (or subsets of them) in a balanced way. Although hedging baskets may appear to be the opposite of trading baskets, both concepts are intimately related and both can be computed using similar procedures.

Lopez de Prado and Leinweber (2012) reviewed the literature on hedging methods. Among the methods they studied are Equal-Risk Contribution (ERC), Maximum Diversification Ratio (MDR) and Mini-Max Subset Correlation (MMSC). The three are static (time-invariant) methods that attempt to balance the risk or exposure of the basket among its constituents. MDR solves a "hedging" problem, while MMSC and ERC can be applied to solve both, a "hedging" as well as a "trading" problem.

Maillard, Roncalli and Teiletche (2009) and Demey, Maillard and Roncalli (2010) gave a formal definition of ERC. Previous descriptions can be found in Qian (2005, 2006) under the term "risk parity", Booth and Fama (1992), and earlier authors. This procedure attempts to balance the contribution of risk for each of the basket's legs. Empirical studies of ERC's performance against alternative weighting schemes can be found in Neukirch (2008), DeMiguel, Garlappi and Uppal (2009) and Hurst, Johnson and Ooi (2010). Most authors impose the constraints that all weights must be positive and add up to one, because they have in mind an asset allocation application. Our analysis is free of such constraints because we would like to discuss the problem of constructing a basket in general terms, rather than focusing on a particular use. This concept's popularity is illustrated by the many institutional asset managers offering ERC-weighted funds: PanAgora Asset Management, Bridgewater Associates, AQR Capital, Aquila Capital, Invesco, First Quadrant, Putnam Investments, ATP, Barclays Global Investors, Mellon Capital Management, State Street Global Advisors, ... to cite only a few.
MDR was proposed by Choueifaty and Coignard (2008) and Choueifaty, Froidure and Reynier (2011). Their goal is to maximize a "diversification ratio" that effectively balances the basket's exposure to each leg, measured in terms of correlation. Like in the ERC case, these authors also incorporate constraints characteristic of an asset allocation framework, which we will obviate in this study for the sake of generality.

MMSC was introduced by Lopez de Prado and Leinweber (2012). This procedure balances the exposure of the basket, not only to each leg (like MDR) but also to any subset of legs. The motivation is to reduce the basket's vulnerability to structural breaks, i.e. when a subset receives a shock that does not impact the rest of the basket. In a basket of two instruments, MMSC coincides with MDR, since the only subsets are the legs themselves. Furthermore, we will see that when only two instruments are considered, ERC, MDR and MMSC give the same solution. However, the three procedures exhibit substantial differences whenever we are dealing with baskets of more than two instruments.

The three procedures are theoretically sound. The purpose of this study is not to invalidate or criticize any of them, but to evidence the differences and properties associated with each solution. A second goal of this paper is to provide efficient algorithms for the calculation of ERC, MDR and MMSC. Hundreds of billions of dollars are invested using balanced basket approaches (particularly ERC), and yet no optimization algorithm can be found in the academic literature. A third, ancillary goal, is to provide a procedure for reducing the dimension of a covariance matrix to a number that makes these methodologies computationally feasible. We believe that our Covariance Clustering method has important applications for the management of risks in large portfolios of highly correlated instruments or funds. Given the analytical and algorithmic nature of this paper, an empirical study of the absolute and relative performance of balanced baskets over the past years is beyond its scope. Such study would merit an extensive and monographic discussion.

The rest of the paper is organized as follows: Section 2 discusses the hedging problem in a two-dimensional framework. Section 3 evidences the qualitative difference between working in two dimensions and dealing with three or more. Section 4 extends our "hedging" analysis to the problem of computing "trading baskets." Section 5 summarizes our conclusions. Appendix 1 derives a numerical procedure for the calculation of ERC baskets. Appendix 2 presents a codification of that algorithm in Python. Appendices 3 and 4 do the same in the context of MMSC and MDR baskets. Appendix 5 describes the Covariance Clustering method, and includes its implementation in Python.

2. THE TWO-DIMENSIONAL HEDGING PROBLEM
Suppose that a portfolio manager wishes to hedge her position of 1,000 S&P Midcap 400 E-mini futures contracts (Bloomberg code "FA1 Index") using S&P 500 E-mini futures (Bloomberg code "ESI Index"). The relevant covariance and correlation matrices on daily market value dollar changes ($\Delta P_t$) are shown in Figure 1. As expected, we can appreciate a high and positive codependence between these two products, with an estimated correlation coefficient approaching 0.95.
Given some holdings \( \omega \) and a covariance matrix \( V \) of market value dollar changes, we can compute the variance of market value changes of the resulting basket \( B \) of \( n \) constituents as

\[
\sigma_{\Delta B}^2 = \omega' V \omega = \sum_{i=1}^{n} \sum_{j=1}^{n} \omega_i \sigma_{i,j} \omega_j
\]

where \( \sigma_{i,j} \) is the covariance between \( \Delta P_i \) and \( \Delta P_j \). One option would be to compute the vector \( \omega_{Mv} \) that minimizes the basket’s variance,

\[
\omega_{Mv} = \arg \min_{\omega} \sigma_{\Delta B}^2
\]

for which a general solution can be found in Lopez de Prado and Leinweber (2012). In our particular case, \( n=2 \) and our holdings of "FA1 Index" are set fix to be \( h_1 = 1,000 \), so the only free holding is \( \omega_2 \). Deriving,

\[
\frac{\partial \sigma_{\Delta B}^2}{\partial \omega_i} = 2 h_i \sigma_{i,2} + 2 \omega_2 \sigma_i^2
\]

\[
\frac{\partial^2 \sigma_{\Delta B}^2}{\partial \omega_2^2} = 2 \sigma_2^2 > 0
\]

Because \( \frac{\partial^2 \sigma_{\Delta B}^2}{\partial \omega_2^2} > 0 \), applying the first order condition \( \frac{\partial \sigma_{\Delta B}^2}{\partial \omega_2} = 0 \) will lead to a minimum at the point \( \omega_{Mv} = h_1 \left( 1, -\frac{\sigma_{1,2}}{\sigma_2^2} \right) \approx (1000, -1467.85) \), with \( \sigma_{\Delta B} \approx 299,709.02 \). The solution matches the coefficients of an Ordinary Least Squares (OLS) regression, which should have been expected, since the regression’s objective function coincides with Eq. (2) for the case \( n=2 \).

Scherer (2010) and Clarke, de Silva and Thorley (2011) offer practical examples of OLS and Minimum Variance (MV) solutions. Although ubiquitous, the solutions provided by these procedures exhibit a few undesirable traits. For example, note that if we had switched contracts, the alternative second holding would have been \( \tilde{\omega}_2 = -\frac{x_2}{\sqrt{\sigma_2^2}} = -\frac{\sigma_{1,2}}{\sigma_2^2} \), and \( \tilde{\omega}_2 \neq -\frac{1}{\omega_2} \) unless \( \sigma_1 = \sigma_2 \). So the ordering of the instruments introduces some arbitrariness to this solution. Also, the risk contributed by each of the basket’s constituents is not equal. In order to evaluate that, we need first to formalize the concept of Contribution to Risk.

### 2.1. CONTRIBUTION TO RISK

The marginal contribution to the basket’s variance from a leg \( i \) is

\[
\frac{\partial \sigma_{\Delta B}}{\partial \omega_i} = \frac{1}{2} \left( \sigma_{\Delta B} \right)^{-1/2} \sigma_{i,\Delta B} - \frac{\sigma_{\Delta B} \Sigma_i \omega_i \sigma_{i,l} / \sigma_{\Delta B}}{\omega_i \sigma_{\Delta B}}
\]
where $A_S = \Delta P_i \omega_i$ and $\sigma_{\Delta B, \Delta S_i}$ is the covariance between changes in the basket's dollar value and changes in the $z$th leg's dollar value. Because volatility is a homogeneous function of degree 1, Euler's theorem allows us to write

$$\sigma_{\Delta B} = \sum_{i=1}^{n} \omega_i \frac{\partial \sigma_{\Delta B}}{\partial \omega_i} = \frac{1}{\sigma_{\Delta B}} \sum_{i=1}^{n} [V \omega \circ \omega]_i$$

(5)

where $\circ$ represents a Hadamard product. We can define Contribution to Risk (CtR) as

$$CtR_i = \frac{\partial \sigma_{\Delta B}}{\sigma_{\omega_i}} \frac{\omega_i}{\sigma_{\Delta B}} \frac{[V \omega \circ \omega]_i}{\sigma_{\Delta B}} = \frac{\omega_i \sigma_{i} \sum_{j=1}^{n} \omega_j \sigma_{j} \rho_{i,j}}{\sigma_{\Delta B} \sigma_{\Delta B}} = \frac{\sigma_{\Delta B, \Delta S_i}}{\sigma_{\Delta B}^2}$$

(6)

and obviously $\sum_{i=1}^{n} CtR_i = 1$. One hedging option would be to estimate the Equal-Risk Contribution (ERC) vector, such that $CtR_i = CtR_j, \forall i, j$. In our $n=2$ case, that occurs when

$$h_1^2 \sigma_1^2 + h_1 \sigma_1 \omega_2 \sigma_2 \rho_{1,2} = \omega_2^2 \sigma_2^2 + \omega_2 \sigma_2 h_1 \sigma_1 \rho_{1,2}$$

(7)

which leads to $\omega_{ERC} \approx (1000, -1552.48)$. Unlike in the OLS solution, the ERC solution is not affected by the order of the instruments. The risk of ERC's hedging basket is $\sigma_{\Delta B} \approx 303,879.05$. That is very close to the minimum risk of the MV-OLS solution, with the advantage that with ERC we get $CtR_1 = CtR_2 = \frac{1}{2}. That is not the case of $\omega_{MV} \approx (1000, -1467.85)$, for which we obtain $CtR_1 = 1$ and $CtR_2 = 0$. So although the MV-OLS solution has reduced risk from 920,304.74 to a minimum 299,709.02, the first leg is still responsible for the entirety of the risk. In order to understand why, we need to introduce the concept of Correlation to Basket (CtB).

### 2.2. CORRELATION TO BASKET

The correlation of each constituent to the overall basket can be computed as

$$\sigma_{\Delta S_i, \Delta S_j} = \omega_i \sigma_i \omega_j = \omega_i \sigma_i \omega_j \sigma_{j, i}$$

(8)

$$\sigma_{\Delta B, \Delta S_i} = \sum_{j=1}^{n} \sigma_{\Delta S_i, \Delta S_j} = \omega_i \sigma_i \sum_{j=1}^{n} \omega_j \sigma_{j, i} \rho_{i, j}$$

(9)

$$CtB_i = \rho_{\Delta B, \Delta S_i} = \frac{\sigma_{\Delta B, \Delta S_i}}{\sigma_{\Delta B} \sigma_{\Delta S_i}} = Sn[\omega_i] \frac{\sum_{j=1}^{n} \omega_j \sigma_{j, i} \rho_{i, j}}{\sigma_{\Delta B}} = Sn[\omega_i] \rho_{\Delta B, i}$$

(10)
where \( \sigma_{\Delta S_i} = \sqrt{\sigma_{\Delta S_i}^2} = 1\omega_i |\sigma_i| \). The factor \( Sgn[\omega_i] \) arises from simplifying \( \frac{\omega_i}{\sqrt{\omega_i^2}} \). Lopez de Prado and Leinweber (2012) show how to compute a general hedging basket such that \( CtBi = CtB_i Vi,j \), which the authors call MMSC (for Mini-Max Subset Correlation). In our particular \( n=2 \) case, we expect the hedge to be \( \omega_2 < 0 \) since \( p_{i,2} > 0 \), so the equal exposure is achieved at

\[
\begin{align*}
  h_1 \sigma_1 + \omega_2 \sigma_2 \rho_{1,2} &= -\omega_2 \sigma_2 - h_1 \sigma_1 \rho_{1,2} \\
  \omega_2 &= -h_1 \frac{\sigma_1}{\sigma_2}
\end{align*}
\]

which means that the MMSC and ERC solutions coincide when \( n=2 \). There is a connection between \( CtR \) and \( CtB \), as evidenced by Eq. (6) and Eq. (10), specifically,

\[
CtR_i = \frac{\sigma_{\Delta S_i}}{\sigma_{\Delta B}} CtBi = \frac{1\omega_i |\sigma_i|}{\sigma_{\Delta B}} CtBi
\]

From this expression we deduce that an ERC basket will match the MMSC solution when \( 1\omega_i |\sigma_i| = |\omega_j| |\sigma_j| \). This condition is clearly met when \( n=2 \) (see Eq. 11), however that is not generally true when \( n>2 \).

For \( \omega_{MV} \approx (1000, -1552.48) \), we obtain \( CtB_1 = CtB_2 \approx 0.17 \), while for \( \omega_{MMSC} \approx (1000, -1467.85) \) we obtained \( CtB_1 \approx 0.33 \) and \( CtB_2 = 0 \). So what the MV-OLS solution has done is to add an "ESI Index" leg that is orthogonal to the basket. The risk from \( \omega_{MV} \) is slightly lower (in fact, the lowest), however it is still concentrated in the first leg (\( CtR_1 = 1 \)). These are good reasons for favouring MMSC or ERC solutions over MV or OLS.

### 2.3. SPECTRAL DECOMPOSITION

A change of basis will allow us to understand geometrically the previous analysis. Because a covariance matrix \( V \) is squared and symmetric, its eigenvector decomposition delivers a set of real-valued orthonormal vectors that we can use to plot the MV-OLS and the MMSC-ERC solutions. Applying the Spectral Theorem,

\[
V = WAW
\]

where \( \Lambda \) is the eigenvalues matrix, \( W \) is the eigenvectors matrix, and \( W \) denotes its transpose. \( \Lambda \) is a squared diagonal matrix and the columns of \( W \) are orthogonal to each other, i.e. \( W = W'1 \), with unit length. For convenience, we reorder columns in \( W \) and \( \Lambda \) so that \( A_{i,j} \geq A_{i,j}', V_j > i \). The \( z \)th principal component is defined by a portfolio with the holdings listed in the \( z \)th column of \( W \). Looking at the above equation, \( \Lambda \) can then be interpreted as the covariance matrix between the principal components characterized by the columns of \( W \). The factor loadings vector \( f_{\omega} = W'\omega \) gives us the projection of \( \omega \) into this new orthogonal basis. This can be verified from \( \sigma_{BB}^2 = \omega'V\omega = \omega'WAW'\omega = \omega'N_\omega \). The product \( f_I = W'1 \), where \( I \) represents the identity matrix, gives us the directions of the old axes in the new basis.
Going back to our original example, 

\[
W = \begin{bmatrix}
0.84646 & -0.53245 \\
0.53245 & 0.84646
\end{bmatrix}, \quad \Lambda = \begin{bmatrix}
1171422.026 & 0 \\
0 & 26946.176
\end{bmatrix}
\]  \quad (14)

For the initial position \( \omega_0 = [1000, 0] \), the factor loadings are \( f_{\omega_0} \approx [846.46203, -532.44908] \). It becomes evident that \( \omega_0 \) is not well hedged, because the \( f_{M_0} \) vector is not pointing in the direction of the second orthogonal component, which is the one with least variance. If we adopted the MV-OLS solution, the resulting basket would take the direction \( f_{MMV} \approx \begin{bmatrix} 64,90733 \\ -1774.9273 \end{bmatrix} \) in the orthonormal axes. That is a noticeable improvement over \( \omega_0 \), as \( \omega_{MMV} \) is shown to be much closer to the second orthogonal component, thus it remains little directional risk in terms of these two instruments ("FA1 Index" and "ESI Index"). And yet the MMSC-ERC solution is less directional, with \( f_{\omega_{MMSC}} \approx \begin{bmatrix} 19.84512 \\ -1846.56505 \end{bmatrix} \).

The first component is typically associated with market risk, of which \( \omega_{MMSC} \) exhibits the least. The MMSC basket is almost completely associated with spread risk, which is best captured by the second component. Lopez de Prado and Leinweber (2012) show how to compute in general a hedging portfolio with zero exposure to the first components. For the \( n=2 \) case, it means that

\[
f_{\omega} = W' \begin{bmatrix}
\omega_1 \\
\omega_2
\end{bmatrix} = \begin{bmatrix}
0 \\
\frac{\sigma_{\Delta B}}{\sqrt{\Lambda_{2,2}}}
\end{bmatrix}
\]

(15)

and re-scaling to meet the condition that \( \omega_1 = h_1 \), we obtain that \( \omega_{PCA} = h_1 \left( I_{-\frac{\omega_0 \cdot \omega}{\|\omega_0\|^2}} \right) \). In our example, this means that \( \omega_{PCA} \approx (1000, -1589.75) \). Then, \( f_{\omega_{PCA}} \approx \begin{bmatrix} 0 \\ -1878.11386 \end{bmatrix} \), which indicates that we are perfectly hedged against the most volatile of the orthogonal components (typically known as the "market component"), as expected. Figure 2 graphs these vectors.

[FIGURE 2 HERE]

The columns of \( W \) (which characterize the Principal Components) represent linear combinations of the securities, which happen to be orthogonal to each other, sorted descending in terms of the variance associated with that direction. In this PCA approach, we are hedged against all risk coming from the first \( n-1 \) components (with the largest variance), and fully concentrated on the \( n-th \) component (with the smallest variance). One caveat of the PCA approach is that the interpretation of the \( n-th \) component is not necessarily intuitive in terms of the basket constituents. For example, in the ERC approach we know that each leg contributes equal risk,
and in the MMSC approach that the basket had equal exposure to each leg. That direct connection between the solution and the basket constituents is missing in the PCA approach, because of the change of basis. This is generally perceived as a drawback, particularly in highly-dimensional problems. MMSC is an appealing alternative to PCA because MMSC searches for a basket as orthogonal as possible to the legs, without requiring a basis change (like PCA). So although MMSC’s solution is close to PCA’s, it can still be linked intuitively to the basket’s constituents. Understanding how this is done beyond the two-dimensional framework requires us to introduce the concept of subset correlation.

3. HEDGING BEYOND TWO DIMENSIONS
In the previous section we argued in favor of MMSC and ERC over alternative hedging approaches such as MV or PCA. The problem with MV’s solution was that risk contribution (measured in terms of CtR) and exposure (measured in terms of CtB) of the basket’s constituents is typically unbalanced. The problem with PCA was that the solution cannot be immediately (or for that matter, intuitively) understood in terms of the basket’s constituents. In this section we will examine the distinct features of MMSC and ERC beyond the two-dimensional case.

The previous example was simplified by the fact that we were only considering two dimensions. As we will see next, there is a qualitative difference between working in two dimensions and working with three or more. Suppose that a portfolio manager wishes to hedge her position of 1,000 S&P Midcap 400 E-mini futures contracts ("FA1 Index") using two other instruments: S&P 500 E-mini futures ("ESI Index") and DJAI E-Mini futures ("DM1 Index"). The relevant covariance and correlation matrices of daily market value (dollar) changes are shown in Figure 3, with the corresponding eigenvectors and eigenvalues matrices.

[FIGURE 3 HERE]

As expected, we can appreciate a high and positive codependence between these three products. It is of course desirable to hedge our position using instruments highly correlated to it (about 0.95 correlation against "ESI Index", and 0.91 correlation against "DM1 Index"). Unfortunately, that comes at the cost of having to deal with a similarly high correlation between the hedging instruments (about 0.98 correlation between "ESI Index" and "DM1 Index"). This poses a problem because there may be an overlap between the hedges, which was not present in the two-dimensional case.

3.1. THE EQUAL-RISK CONTRIBUTION (ALIAS, RISK PARITY) SOLUTION
We will illustrate this last point by computing the ERC basket, which solves the problem (recall Eq. (6))

$$\omega_{ERC} = \left\{ \omega_i \left| \frac{\sigma_{\Delta B, \Delta i}}{\sigma_{\Delta B}} = \frac{1}{n}, \forall i \right. \right\}$$ (16)

Appendix 1 provides the details of this calculation, for any n dimensions, and Appendix 2 offers an algorithm coded in Python which computes the ERC basket.
Figure 4 reports the results of applying this algorithm to the input variables in Figure 3. A first problem with this result is the uneven correlations to the basket (CtB). The ("ESI Index", "DM1 Index") subset will dominate the performance of the hedge, with its 0.26 correlation to the basket. This could have potentially serious consequences should there exist a correlation break between "ESI Index" and "DM1 Index" on one hand and "FA1 Index" on the other. A second problem is that the solution itself is not unique. Figure 5 presents an alternative solution for which also $CtR_i \approx \frac{1}{3}, \forall i$, with unacceptably high values like $CtB_2 > 0.99$. We would of course reject this alternative solution out of common sense, however it would be better to rely on a procedure that searches for reasonable hedges, if possible with unique solutions.

In conclusion, ERC does not necessarily deliver a unique and balanced (exposure-wise) solution when $n > 2$.

### 3.2. Diversified Risk Parity

Building on an idea of Meucci (2009b), Lohre, Neugebauer and Zimmer (2012) and Lohre, Opfer, and Orszag (2012), proposed a very interesting variation of ERC that they branded "Diversified Risk Parity" (DRP). It computes the allocations such that the contribution to risk from every principal component is equal. In a nutshell, DRP is like ERC, but computed on the principal components instead of the actual instruments. Like PCA, DRP also requires a spectral decomposition, and thus a change of basis. We cannot strictly classify it as a balanced basket, however we will derive its calculation for illustration purposes. Our DRP solution differs from Lohre, Neugebauer and Zimmer (2012) in two aspects: First, we are not imposing the asset allocation constraints (non-negativity, additivity to one), because they are not relevant in a hedging framework. Second (and as a consequence of the first aspect), our solution is analytical, while theirs is numerical.

#### 3.2.1. Analytical Solution

Recall from Section 2.3 that the eigenvectors of a symmetric matrix are orthonormal, thus $W'W = I$ and $\Lambda_{i,j} = 0, \forall i \neq j$. The correlation between any two distinct principal components is $\frac{\Lambda_{i,j}}{\sqrt{\Lambda_{i,i} \Lambda_{j,j}}} = 0, \forall i \neq j$. This implies that, given some factor loadings $f_\omega = W_\omega$, the basket's variance can be simply decomposed as $\sigma_{BP}^2 = \sum_{i=1}^n [f_\omega]_i^2 \Lambda_{i,i}$, where $[f_\omega]_i$ is the $i$th element of the factor loading's vector. We conclude that the contribution to risk of the $i$th principal component is

$$CtR_i = \frac{[f_\omega]_i^2 \Lambda_{i,i}}{\sum_{j=1}^n [f_\omega]_j^2 \Lambda_{j,j}} \tag{17}$$

Suppose that $[f_\omega]_i = \sqrt{\frac{\Lambda_{i,i}}{\Lambda_{i,i}}} \forall i$. Then, $CtR_i = \frac{\Lambda_{i,i} \Lambda_{i,i}}{\Lambda_{i,i} \sum_{j=1}^n \Lambda_{j,j}} = \frac{1}{n}$, which is the ERC solution on factor loadings $f_M$. But $f_M$ are loadings in the new basis (of principal components), and we still
need to derive the holdings in the old basis (of actual instruments). Since \( f_\omega = W' \omega \) and \( W = W^{-1} \), these can be computed as

\[
\omega_{DRP} = W' f_\omega = W f_\omega
\]  

(18)

[FIGURE 6 HERE]

In our particular example, Eq. (18) leads to the solution shown in Figure 6. This result is mathematically correct, and yet looking at the actual holdings proposed, many portfolio managers may have a difficult problem understanding it. \( \bar{CtR}i = \frac{1}{3} \bar{V}i \), however CtR and CtB (in terms of the instruments) exhibit a large concentration on "DM1 Index". Another inconvenience of DRP, which in fact is shared by all PCA-style approaches, is their lack of robustness in the presence of numerically ill-conditioned covariance matrices: A change in just one observation may produce a last eigenvector that spans in a completely different direction, and because DRP equally distributes risk among all principal components, the impact on \( \omega_{DRP} \) will be dramatic. This makes PCA-like baskets in general, and DRP in particular, vulnerable to structural breaks and outliers. In Appendix 5 we present a covariance clustering procedure which addresses this concern.

### 3.2.2. THE DANGER OF "HOLDING" AN EIGENVECTOR IN A RISK-ON/RISK-OFF ENVIRONMENT

Regardless of the practicality of the DRP solution, it is useful for understanding why baskets pointing in the direction of an eigenvector should not be part of a hedging basket, except for the eigenvector associated with the smallest eigenvalue. Denote by \( \sigma(W_i) \) the risk of a portfolio pointing in the direction of the \( i \)-th eigenvector, and by \( \sigma[\hat{\omega}_{DRP}] \) the risk associated with a unit-length vector pointing in the direction of the DRP solution. As we move away from \( \hat{\omega}_{DRP} \) and toward \( W_i \), we have computed the intermediate baskets, \( \omega_a = \alpha W_i + (1 - \alpha) \hat{\omega}_{DRP} \), which can be normalized as \( u_a = \frac{\omega_a}{\|\omega_a\|} \). Figure 7(a) plots the resulting risk, \( \sigma[\hat{\omega}^\alpha] \) for \( \alpha \in [0,1] \) in 100 equally spaced nodes. Risk increases as we approach \( W_1 \) and \( W_2 \), but not \( W_3 \). This is because eigenvectors are the critical points of the Rayleigh quotient \( \omega' V \omega \), where the numerator is the variance of the basket. Consider the optimization program \( \max \omega' V \omega, \text{subject to } \omega' \omega = 1 \), with a Lagrangian \( L(\omega, \lambda) = \omega' V \omega - \lambda (\omega' \omega - 1) \). Applying first order conditions on \( \omega \),

\[
\frac{d L(\omega, \lambda)}{d \omega} = (V + V') \omega - 2 \lambda \omega = 2 V \omega - 2 \lambda \omega = 0 \Rightarrow \lambda = \frac{\omega' V \omega}{\omega' V' \omega}.
\]

But that is precisely the generalized eigenvalue problem. So, finding the largest eigenvalue of \( V, \lambda = \Lambda_{1,1} = V' W_i \), leads us to the maximum, which is achieved by the first eigenvector, \( W_1 \). Furthermore, because \( VW = WA \Rightarrow \Lambda_{i,i} = \frac{w_i' V W_i}{w_i' W_i} \), gives a Rayleigh quotient, all critical points (and extreme values in particular) of this optimization program are derived from computing the eigenvectors of \( V \) with stationary values in \( \Lambda \).

[FIGURE 7(a) HERE]
Because $VW = WA$, the eigenvector makes the covariance matrix behave like a mere scalar. A portfolio that concentrates risk in the direction of a particular eigenvector $W_i$ (except for $i=n$), is investing in a single bet. This means that the investment universe that forms $V$ will not be able to dissipate a "hit" that comes from the direction of $W_i$, a situation particularly dangerous in a risk-on/risk-off environment. Figure 7(b) displays the risk of a portfolio that moves away from DRP and toward $W_1$, before (risk-off) and after (risk-on) a 100% increase in $\sqrt{\Lambda_{1,1}}$ (standard deviation in the direction of $W_1$). As we can see, investors holding the first eigenvector receive the entirety of the shock, and their risk is doubled. That shock would have been greatly dissipated by the investment universe if the investor had held a portfolio closer to DRP, instead of being so exposed to the first eigenvector.

[FIGURE 7(b) HERE]

3.3. THE MAXIMUM-DIVERSIFICATION SOLUTION

Choueifaty and Coignard (2008) compute the vector of holdings that Maximize the Diversification Ratio (MDR), as defined by

$$\omega_{\text{MDR}} = \arg \max_\omega \frac{\sum_{i=1}^n \omega_i \sigma_i}{\sigma_{\Delta B}} \quad (19)$$

This diversification ratio is the ratio of weighted volatilities divided by the basket's volatility, and it is closely related to our Eq. (10). MDR is an intuitive method that penalizes the risk associated with cross-correlations, as they are accounted by the denominator but absent in the numerator of the maximized ratio. Choueifaty, Froidure and Reynier (2011) show that the correlations of each leg to the MDR hedging basket are minimized and made equal.1 Figure 8 presents the MDR solution to our example. $CtB_i \approx 0.08$, $V_i$, and there is no way we can make $CtB$ smaller for all legs. Now $CtR$ values make more sense, because the leg with a negative holding is responsible for almost $\frac{1}{2}$ of the total risk, with the other half going to the legs with a positive holding. Among the "long" legs the risk is not equally spread, because the correlation between "DM1 Index" and "ESI Index" (about 0.98) is greater than the correlation between "FA1 Index" and "ESI Index" (about 0.95). So the MDR result seems intuitive and preferable to the ERC result.

[FIGURE 8 HERE]

However, if we look more closely into this MDR result, we will find something not entirely satisfactory. The problem is, there are subsets of instruments that, combined, exhibit greater correlation to the overall basket. Figure 8 shows that, even though the exposure is perfectly balanced at the leg level, this hedging basket's performance may still be dominated by some groups of legs. In particular, there is an approx. 0.36 correlation between the subset made of ("ESI Index", "DM1 Index") and the overall basket. Like in the ERC case, that could be a source of losses should there be a correlation break between large-cap and mid-cap stocks, as witnessed most recently during the 2008 financial crisis. Furthermore, Choueifaty and Coignard (2008)

---

1 As we will show in the next Section, this is equivalent to a MMSC where only subsets of one instrument are taken into consideration.
acknowledge that the solution may not be unique or robust, particularly with ill-conditioned covariance matrices. Adding some structure to the optimization program would alleviate these problems.

Succinctly, although MDR is to some extent preferable to ERC, it does not address the problems of uniqueness of solution and balanced exposure of subsets of legs to the overall basket.

3.4. THE MINI-MAX SUBSET CORRELATION SOLUTION

We denote subset correlation as the correlation of a subset of instruments to the overall basket. MMSC's goal is to prevent that any leg or subset of legs dominates the basket's performance, as measured by its subset correlations. This additional structure adds the robustness and uniqueness of solution that were missing in ERC and MDR. MMSC baskets are also more resilient to structural breaks, because this approach minimizes the basket's dependency to any particular leg or subset of legs. Suppose for instance that $\Lambda_{1,3}$ rises and as a result the correlation of the basket to those legs and subsets most exposed to the third principal component increases by a function of $\Delta\Lambda_{1,3}$. Because MMSC provided the most balanced exposure, it will generally be the least impacted basket. We will illustrate this point with an example in Section 3.5.

When we are dealing with 2 instruments, the only subsets were the instruments themselves, so the only subset correlations were the CBs. The MMSC solution coincided with the MDR solution. But now that $n=3$, we can compute correlations to 6 subsets (the 3 single legs plus the 3 possible pairs of legs), and we need to distinguish between both procedures. The solution can be characterized as

$$\omega_{MMSC} = \arg\min_\omega \left\{ \max_{i,j} \rho_{AB,\Delta s_i} \right\}$$

where $i=1,\ldots,N$ and $s=1,\ldots,N$.

When $n=3$, suppose that $M_M = \{(1,2,3), (1,2), (1,3), (2,3), (1), (2), (3)\}$. Then, the correlation matrix $\rho_{MM}$ can be written as

$$\rho_{MM} = \begin{pmatrix}
\rho_{11} & \rho_{12} & \rho_{13} \\
\rho_{21} & \rho_{22} & \rho_{23} \\
\rho_{31} & \rho_{32} & \rho_{33}
\end{pmatrix}$$

where $\rho_{ij}$ is the correlation between assets $i$ and $j$.

When $n=4$, the number of subsets increases significantly, and the computation becomes more complex. However, the principle remains the same: to find the subset of instruments that minimizes the maximum correlation to any other subset.

[FIGURE 9 HERE]

$C_R \approx \frac{1}{2}$ seems to point at a concentration of risk in the "FA1 Index" leg. This is an artifact of computing $C_R$ when one of the legs has low correlation to the basket. In this instance, $C_R$ will not be able to accurately split risk among the instruments. From Appendix 1 we know that
\[
\frac{dCtR_i}{\partial \omega_j} = \frac{\omega_i \sigma_i^2}{\sigma_{\alpha B}^2} \left( 1 - 2 \rho_{\alpha B}^2 \sigma_{\alpha B}^2 \right) + \frac{\sigma_{i, B, \Delta B, \Delta S_j}}{\sigma_{\alpha B}} 
\]

(21)

\[
\frac{dCtR_i}{\partial \omega_j} \approx \frac{\omega_i \sigma_i^2}{\sigma_{\alpha B}^2} \quad \text{because} \quad \rho_{\alpha B}^2, S_j^2 > 0, \frac{\sigma_{i}}{\sigma_{\alpha B}} > 1, \text{because this is a hedging basket. Thus, a very small change in the holdings could transfer a substantial amount from } CtR_1 \text{ to } CtR_2, \text{ which is counter-intuitive. We can also see that, despite the high } CtR_1 \text{ value, } CtB_1 \text{ is virtually the same as the correlation of the subset formed by ("FA1 Index", "ESI Index") to the basket, or the correlation of the subset ("ESI Index", "DM1 Index") to the basket. In contrast, CtB is more stable, with }
\]

\[
\frac{dp_{\alpha B} \Delta S_j}{d \omega_j} = \frac{\sigma_{i}}{\sigma_{\alpha B}} \left( 1 - \rho_{\alpha B}^2 \right) 
\]

(22)

Summarily, MMSC provides the basket with the most balanced exposure to any of its legs or subsets of legs. The solution is also unique and more robust, virtue of the 6 conditions imposed on only 2 holdings. The fact that } N > n \text{ as } n \text{ grows provides a competitive advantage for users with access to high performance computing (HPC) facilities, who can deploy a solution unavailable to investors with limited computational power. When } n > 30, \text{ the computations involved would present a challenge for today's supercomputers. However, that obstacle could be surmounted by clustering the legs into highly correlated blocks, as proposed in Appendix 5. A second alternative is to limit the maximum size of the subsets evaluated, which can be done through the parameter } maxSubsetSize \text{ in the Python code provided in Appendix 4.}

3.5. GEOMETRIC INTERPRETATION

As we did in the two-dimensional case, we can understand the previous three-dimensional results in geometric terms. For the initial position \( \omega_0 = \begin{bmatrix} 1000 \\ 0 \\ 0 \end{bmatrix} \), the factor loadings are \( f_{\omega_0} \approx \begin{bmatrix} 774.88384 \\ -620.81023 \\ 118.95250 \end{bmatrix} \). As expected, \( \omega_0 \) is not well-hedged, because the \( f_{\omega_0} \) vector is nowhere close to the orthogonal directions with least variance. ERC is better hedged, with \( f_{\omega_{ERC}} \approx \begin{bmatrix} 26.23808 \\ -558.67818 \\ 5320.06831 \end{bmatrix} \). For MDR, we get \( f_{\omega_{MDR}} \approx \begin{bmatrix} -388.45011 \\ 14.14535 \\ 6287.25928 \end{bmatrix} \), and for MMSC we get \( f_{\omega_{MMSC}} \approx \begin{bmatrix} 33.59253 \\ -691.79627 \\ 4577.41954 \end{bmatrix} \). So it would appear as if, in this particular example, MMSC did a worse job than MDR and ERC. Figure 10 tells us that this is not the case.

\(^2\) See Lopez de Prado and Leinweber (2012) for a proof.
[FIGURE 10 HERE]

The first table reports the factor loadings of the three balanced hedging baskets to the first principal component. Although \( f_{\omega_{\text{MMSC}}}(1) \approx 33.59253 \) is slightly greater than for the other baskets, when it comes to the subsets we find that \( f_{\omega_{\text{MMSC}}} \) is generally the least exposed. This is an important feature because, in the case of a structural break\(^3\), MMSC’s hedging basket will tend to be the least impacted. Figure 10 shows that MMSC is the least exposed to shocks in the direction of any principal component. This is the result of minimizing the correlation of any subset to the overall basket. From Figures 4, 8 and 9, we know that all three balanced baskets had the greatest exposure to subset (2,3). This subset, composed of ("ESI Index", "DM1 Index"), is particularly sensitive to shocks in the direction of the third principal component (see panel 3 in Figure 10). Figure 11 displays the covariance and correlation matrices that result from a shock in the direction of that third principal component, in particular a 25% increase to \( \Lambda_{3,3} \). The differences between both correlation matrices seem negligible, and yet the impact on ERC and MDR is significant. Figure 12 reports how, as a consequence of this structural break, the correlations of each balanced basket to subset (2,3) are impacted. As expected, MMSC is the only basket with a relatively low exposure of around 0.23 (compared to the previous 0.19), while ERC and MDR’s go up to approx. 0.31 and 0.41 respectively.

[FIGURE 11 HERE]

[FIGURE 12 HERE]

4. TRADING BASKETS

As we have seen earlier, a hedging basket attempts to minimize the exposure to any of its constituents. In contrast, a trading basket tries to determine the holdings such that the exposure is maximized. With that difference in mind, the problem is again how to determine a basket with balanced exposures, i.e. that no particular leg or subset of legs is responsible for the overall basket’s performance.

As discussed earlier, MDR’s goal is to maximize diversification by minimizing all CtB. This is consistent with a hedging problem. As originally formulated, it cannot be used to compute a trading basket.

The ERC procedure can sometimes deliver a trading basket by chance. The result reported in Figure 5 happened to be a trading basket, even though we were searching for a hedging basket. This is because ERC does not have a way to control for those two different objectives. ERC may converge to one or the other depending on the initial seed (e.g., a vector of ones in the case of Figure 5). This problem could be circumvented by trying different seeds. As expected, the resulting ERC trading basket for our example is balanced in terms of contribution to risk (CtR).

---

\(^3\) In this context, we speak of a “structural break” with the meaning of a change in the direction of a principal component. This is what occurs if, for example, we stretch the location-dispersion ellipsoid in the direction of an eigenvector. See Meucci (2009a), Chapter 3.
When used to compute a trading basket, MMSC will maximize the minimum subset correlation (as opposed to minimizing the maximum subset correlation). In every iteration, it will push the lowest subset correlation to a higher value, thus raising the average correlation, until there is no way to increase any correlation without reducing another.

$$\omega = \arg\max_{\omega} \{\min|p_{AB,\Delta S_i}|\}$$  \hspace{1cm} (23)

where \(i=1, ..., N\) subsets, \(N = \sum_{i=1}^{n-1} \binom{n}{i} = 2(2^n - 1)\) is the number of subsets, \(\Delta S_i = \sum_{j=1}^{n} \Delta S_j \omega i_j\), and \(\omega_i\) is the vector of holdings of subset \(i\). Figure 13 shows the trading basket computed by MMSC on the same three instruments used in the previous section. MMSC recognizes that a linear combination of "FA1 Index" and "DM1 Index" makes "ESI Index" redundant, thus giving \(\omega_2 \approx 0\). As a result, MMSC spreads risk equally between "FA1 Index" and "DM1 Index", rather than between three instruments (including a redundant position on "ESI Index"), like the ERC approach did. Perhaps more important is that MMSC actively searches for this trading basket, rather than arriving at it thanks to a fortunate seed. Appendix 4 provides the Python code that computes MMSC hedging and trading baskets, depending on the user’s preference.

[FIGURE 13 HERE]

5. CONCLUSIONS

PCA is a good theoretical option for computing hedging baskets. One drawback of the PCA approach is that the interpretation of the \(n\)-th component is not necessarily intuitive in terms of the basket's constituents. This is due to the change of basis involved in PCA's eigen-decomposition of variance, and for \(n>3\) it is usually difficult to visualize the intuition behind the resulting eigenvectors. A case in point was the solution delivered by the DRP basket which, although mathematically correct, defied common sense.

For this reason, practitioners typically rely on approaches that allow them to relate the basket's holdings to the statistical properties of the basket's constituents. Such is the purpose of balanced hedging baskets, which are characterized by spreading risk or exposure across the baskets' constituents, so that the combined risk is not only minimal but also well distributed. Although the solution is not a perfect hedge (i.e., a minimum variance portfolio orthogonal to the main principal components), there is no change of basis involved and therefore the basket can be understood in terms of its constituents.

How well risk is spread is measured by the Contribution to Risk (CtR) from each leg. Similarly, how well exposure is spread is measured by each leg's Correlation to the Basket (CtB). Three methods have been proposed to compute balanced hedging baskets: Equal-Risk Contribution (ERC), Maximum Diversification (MDR) and Mini-Max Subset Correlation (MMSC). All three are theoretically sound and useful procedures, and the purpose of this paper is not to disqualify any of them but to examine the properties of their respective solutions.
ERC computes the vector of holdings such that the CtR of each leg is equal. Absent of constraints, the solution is not unique and it could lead to hedging baskets riskier than any of the individual holdings. This problem can be circumvented by trying alternative seeds, however the procedure *per se* does not control the outcome. Also, although the solution always delivers equal CtR per leg, the CtB could be very high for some legs, evidencing concentration of exposure.

MDR is equivalent to ERC, with the difference that instead of equalizing CtR, it attempts to achieve equal CtB per leg. Solutions tend to be more intuitive and robust, however they are not unique. Another caveat is that, although each leg’s CtB may be equal, there may be subsets of legs that are highly exposed, i.e. they will dominate the overall basket’s performance.

MMSC computes a hedging basket for which not only CtB are as low as possible, but also the correlations of subsets of legs to the overall basket are minimized. The outcome is a basket which performance is not dominated by any of its constituents, individually or in subsets. This feature is important, because it makes the basket more resilient to structural breaks. Since the number of subsets of legs is necessarily greater than the number of legs, the system is over-determined: There may be no solution that equalizes all subset correlations, in which case MMSC computes the Mini-Max approximation. Another advantage of MMSC is that it can also be used to compute trading baskets. These are characterized by a vector of holdings such that the correlation of each leg or subset of legs to the overall basket is maximized (rather than minimized, like in the hedging case).

One caveat of MMSC is that, for very large baskets, the number of subsets can be enormous, and the calculation of its solution may require access to high performance computing (HPC) facilities, such as the NERSC facility at Lawrence Berkeley National Laboratory. We hope that readers will find helpful the numerically efficient algorithms provided in the Appendices.

---

4 Additional details are available at: [http://www.nersc.gov/users/computational-systems/](http://www.nersc.gov/users/computational-systems/)
APPENDICES

A.1. COMPUTING THE ERC BASKET FOR «-INSTRUMENTS

A.1.1. TAYLOR’S EXPANSION

A second-degree Taylor expansion of the CtR function takes the form:

\[
LCtR_i = \frac{\partial CtR_i}{\partial \omega_i} \Delta \omega_i + \frac{1}{2} \left( \frac{\partial^2 CtR_i}{\partial \omega_i^2} \right)^2 (\Delta \omega_i)^2 + \sum_{k=3}^{\infty} \frac{1}{k!} \left( \frac{\partial^k CtR_i}{\partial \omega_i^k} \right) (\Delta \omega_i)^k
\]

(24)

We therefore need to compute an analytical expression for the first and second partial derivatives.

A.1.2. FIRST DERIVATIVE

From Eq. (6) we know that CtR = \( \frac{\partial \sigma_{AB}}{\partial \omega_i} \omega_i \), thus

\[
\frac{\partial CtR_i}{\partial \omega_i} = \frac{\partial}{\partial \omega_i} \left( \frac{g\sigma_{AB}}{\omega_i} \right) = \frac{\partial^2 \sigma_{AB}}{\partial \omega_i^2} \frac{\omega_i}{\sigma_{AB}} + \frac{\partial \left( \frac{\omega_i}{\sigma_{AB}} \right)}{\partial \omega_i} \frac{\sigma_{AB}}{\sigma_{AB}}
\]

(25)

We already know that \( \frac{\partial \sigma_{AB}}{\partial \omega_i} = \frac{\sigma_{AB,\Delta S_i}}{\omega_i \sigma_{AB}} \) from Eq. (4). We need to compute \( \frac{\partial^2 \sigma_{AB}}{\partial \omega_i^2} \) and \( \frac{\partial \left( \frac{\omega_i}{\sigma_{AB}} \right)}{\partial \omega_i} \). From Lopez de Prado and Leinweber (2012) we also know that \( \frac{\partial \sigma_{AB}}{\partial \omega_i} = \sigma_i \rho_{\Delta B,\Delta S_i} \) and

\[
\frac{\partial \sigma_{AB}}{\partial \omega_i} = \frac{\sigma_{AB}}{\sigma_{AB}} \left( 1 - \rho_{\Delta B,\Delta S_i}^2 \right)
\]

(26)

We therefore conclude that

\[
\frac{\partial CtR_i}{\partial \omega_i} = \frac{\omega_i \sigma_i^2}{\sigma_{AB}^2} \left( 1 - \rho_{\Delta B,\Delta S_i}^2 \right) + \left( \frac{1}{\sigma_{AB}^2} - \frac{1}{\sigma_{AB}^2} + \frac{\sigma_i \rho_{\Delta B,\Delta S_i}}{\sigma_{AB}^2} \right) \sigma_i \rho_{\Delta B,\Delta S_i}
\]

(28)
A.1.3. SECOND DERIVATIVE

Next, we will obtain the expression for

\[
\frac{d^2 CtR_i}{d\omega_i^2} = \frac{\partial}{d\omega_i} \left[ \frac{\omega_i}{\sigma_{\Delta B}^2} \right] \sigma_i^2 (1 - 2\rho_{\Delta B,\Delta S_i}^2) - 4\rho_{\Delta B,\Delta S_i} \sigma_i^2 \frac{\omega_i}{\sigma_{\Delta B}^2} + \frac{\partial \rho_{\Delta B,\Delta S_i}}{d\omega_i} \omega_i \sigma_i^2 + \frac{\partial \rho_{\Delta B,\Delta S_i}}{d\omega_i} \sigma_i
\]  

(29)

We need to determine the analytical expressions for \( \frac{\partial \omega_i}{\sigma_{\Delta B}^2} \) and \( \frac{\partial \rho_{\Delta B,\Delta S_i}}{d\omega_i} \).

\[
a \left[ \frac{\rho_{\Delta B,\Delta S_i}}{\sigma_{\Delta B}^2} \right] = \frac{1}{\sigma_{\Delta B}^2} - 2\frac{\omega_i}{\sigma_{\Delta B}^2} \sigma_i^2 \frac{\omega_i}{\sigma_{\Delta B}^2} - \frac{\partial \rho_{\Delta B,\Delta S_i}}{d\omega_i} \sigma_i^2 \frac{\partial \rho_{\Delta B,\Delta S_i}}{d\omega_i} \omega_i \sigma_i^2 + \frac{\partial \rho_{\Delta B,\Delta S_i}}{d\omega_i} \sigma_i^2  
\]  

(30)

\[
\frac{\partial \rho_{\Delta B,\Delta S_i}}{d\omega_i} \sigma_{\Delta B}^2 = \frac{\partial \rho_{\Delta B,\Delta S_i}}{d\omega_i} \sigma_{\Delta B}^2 - \frac{\partial \sigma_{\Delta B}}{d\omega_i} \rho_{\Delta B,\Delta S_i} \frac{\partial \sigma_{\Delta B}}{d\omega_i} \sigma_{\Delta B}^2 = \frac{\sigma_i}{\sigma_{\Delta B}^2} (1 - 2\rho_{\Delta B,\Delta S_i}^2)  
\]  

(31)

with the conclusion that

\[
\frac{\partial^2 CtR_i}{d\omega_i^2} = \left( \frac{1}{\sigma_{\Delta B}^2} - 2\frac{\omega_i}{\sigma_{\Delta B}^2} \sigma_i^2 \right) \left( 1 - 2\rho_{\Delta B,\Delta S_i}^2 \right) - 4\rho_{\Delta B,\Delta S_i} \sigma_i \frac{\sigma_i}{\sigma_{\Delta B}^2} \left( 1 - 2\rho_{\Delta B,\Delta S_i}^2 \right) + \frac{\omega_i}{\sigma_{\Delta B}^2} \sigma_i^2 \left( 1 - 2\rho_{\Delta B,\Delta S_i}^2 \right)  
\]  

(32)

A.1.4. STEP SIZE

Finally, assuming \( \sum_{k=3}^{\infty} \frac{1}{k!} \frac{\partial^k CtR_i}{\partial \omega_i^k} (\Delta \omega_i)^k = 0 \), we can replace these derivatives into Taylor’s expansion:

\[
ACtRi = \frac{\partial CtR_i}{\partial \omega_i} \Delta \omega_i + \frac{1}{2} \frac{\partial^2 CtR_i}{\partial \omega_i^2} (\Delta \omega_i)^2  
\]  

(33)

giving us the expression
Let us define

\[ \Delta \omega_i = \left[ \frac{\omega_i \sigma_i^2}{\sigma_{AB}^2} \left( 1 - 2\rho_{\Delta B, \Delta S_i}^2 \right) + \frac{\sigma_i \rho_{\Delta B, \Delta S_i}}{\sigma_{AB}} \right] \Delta \omega_i \]

\[ + \left[ \frac{\sigma_i^2}{\sigma_{AB}^2} \left( 1 - 2\rho_{\Delta B, \Delta S_i}^2 \right) - \rho_{\Delta B, \Delta S_i} \frac{\omega_i \sigma_i^3}{\sigma_{AB}^3} \left( 2 - 3\rho_{\Delta B, \Delta S_i}^2 \right) \right] (\Delta \omega_i)^2 \]  

\[ \text{(34)} \]

Let \( a = -\frac{\sigma_i^2}{\sigma_{AB}^2} \left( 1 - 2\rho_{\Delta B, \Delta S_i}^2 \right) - \frac{\omega_i \sigma_i^3}{\sigma_{AB}^3} \left( 2 - 3\rho_{\Delta B, \Delta S_i}^2 \right) \]

\[ b = \frac{\omega_i \sigma_i^2}{\sigma_{AB}^2} \left( 1 - 2\rho_{\Delta B, \Delta S_i}^2 \right) + \frac{\sigma_i \rho_{\Delta B, \Delta S_i}}{\sigma_{AB}} \]

\[ c = -ACTR_i \text{ } \] 

Then, for \( a \neq 0 \) we will choose the smallest step size (to reduce the error due to Taylor’s approximation, which grows with \( \Delta \omega_i \)):

\[ \Delta \omega_i = \begin{cases} 
-b + \sqrt{b^2 - 4ac} & \text{if } b \geq 0 \\
2a & \text{if } b < 0
\end{cases} \]

\[ \text{(36)} \]

For \( a = 0 \), the solution coincides with a first degree Taylor approximation:

\[ \Delta \omega_i = \frac{c}{b} = ACTR_i \left[ \frac{\omega_i \sigma_i^2}{\sigma_{AB}^2} \left( 1 - 2\rho_{\Delta B, \Delta S_i}^2 \right) + \frac{\sigma_i \rho_{\Delta B, \Delta S_i}}{\sigma_{AB}} \right]^{-1} \]

\[ \text{(37)} \]

**A.2. THE ERC ALGORITHM**

We know that the \( \omega_{ERC} \) basket must verify that all Contributions to Risk are equal: \( CtR_i = \frac{1}{n}, \forall i \). Thus, for any \( i \) we can compute the step size \( \Delta \omega_i \) so that any deviation from that value, \( ACTR_i = \frac{1}{n} - CtR_i \), could be corrected through the expression in Eq. (36) (if \( a \neq 0 \)) or Eq. (37) (if \( a = 0 \)).

The following algorithm computes the \( \{CtR_i\} \) vector, determines for which leg \( i \) the deviation \( \Delta \omega_i \) is greatest, and computes the corresponding \( \Delta \omega_i \) that reduces such deviation in the next iteration. The algorithm stops when either the desired accuracy has been achieved, or iterations exceed a user-designated limit. Figures 14 and 15 show how holdings converge to their optimal values for the examples of ERC hedging and trading baskets discussed in Section 3 and 4.
import numpy as np
import scipy as sp

# ERC basket optimization algo
# On 20120516 by MLDP <lopezdeprado@lbl.gov>

class ERC:
    def __init__(self, covar):
        # Class constructor
        self.covar = covar
        self.weights, self.ctr, self.ctb, self.tErr, self.iter = 0, 0, 0, 0

    def solve(self, precision, seed):
        # Compute the ERC holdings and stats
        w = np.zeros(shape=(seed.shape[0], seed.shape[1]))
        w[:] = seed[:]
        iter, n, grad = 0, len(w), range(3)
        while True:
            # 1) Normalize holdings
            w = w / w[0]
            # 2) Compute Ctb, CIR
            iter += 1
            risk = float(w.T * self.covar * w[0, 0]) ** 0.5
            tErr, mErr, iErr = 0, 0, 0
            ctb, ctr, err = [0 for i in range(n)], [0 for i in range(n)], [0 for i in range(n)]
            for i in range(n):
                for j in range(n):
                    ctb[i] = w[j, 0] * self.covar[i, j] / float(self.covar[i, j]) ** 0.5
                    ctb[i] = ctb[i] * risk * get_Sign(w[i, 0])
                    err[i] = abs(w[i, 0] * self.covar[i, j] ** 0.5 / risk) * ctb[i]
                    err[i] = 1 / n * err[i]
                    tErr += err[i] ** 2
                # Determine which leg to change
                if abs(err[i]) > abs(mErr):
                    mErr = err[i]
                    iErr = i
            # 3) Exit conditions
            tErr = (tErr / n) ** 0.5  # Root mean squared deviation
            if tErr < 10 ** -precision: break
            if iter > 10 ** precision: break
            # 4) CIR's Taylor expansion
            grad[0] = -err[iErr]
            grad[1] = w[iErr, 0] * self.covar[iErr, iErr] / risk ** 2 * (1 - 2 * ctb[iErr]) ** 2 + \n                self.covar[iErr, iErr] ** 0.5 * ctb[iErr] / risk
            grad[2] = self.covar[iErr, iErr] / risk ** 2 * (1 - 2 * ctb[iErr]) ** 2 - \n                ctb[iErr] / w[iErr, 0] * (self.covar[iErr, iErr] ** 0.5 / risk) ** 3 * (2 - 3 * ctb[iErr]) ** 2
            # 5) Step size
            if grad[2] == 0:
                if grad[1] == 0:
                    delta = (-grad[1] + delta) / (2 * grad[2])
                else:
                    delta = (-grad[1] + delta) / (2 * grad[2])
else:
    delta = -grad[0]/grad[1]
    w[iErr,0] = delta
    self.weights[iErr,0,0,0] = delta
    self.Terr, self.iter = w, self.add, self.iter

# Seeds for hedging basket, based on alternating OLS holdings
if name == '_main':
    def get_seed(self, col):
        # Seeds for hedging basket, based on alternating OLS holdings
        a = np.zeros(shape=(self.covar.shape[0], self.covar.shape[1]))
        a[1] = self.covar[1]
        a[0,0] = delete(a, col, 1)
        a[0,1] = delete(a, col, 0)
        a[0,2] = delete(a, col, 0)
        b = self.covar[:, col]
        b[0] = delete(b, col, 0)
        c = a * b
        c = np.insert(c, col, -1, axis=0)
        return c

# Sign of the holding, needed to compute Ctb
if number == 0:
    return 0
if number > 0:
    return 1
if number < 0:
    return -1

# Instantiation

def main():
    # 1) Inputs (covariance, optional seed) --- to be changed by the user
    precision = 5
    covar = np.matrix([846660.0,805351971.515812.899769821,403177.0598353136, \ 515812.899769821,351407.396443653,280150.614979364, \ 403177.0598353136,280150.614979364,232934.832710412])
    for seed in covar.shape[0]:
        seed = np.zeros((covar.shape[0], 1))
        seed[:, 0] = (covar.shape[0], 1)
        if number == 0:
            return 0
        if number > 0:
            return 1
        if number < 0:
            return -1

    # Compute and report solution for alternative seeds
    for i in range(covar.shape[0]):
        print '##### SOLUTION ' + str(i+1) + ' ######'
        print '##### Holdings #####'
        print '##### Ctb #####'
        print '##### Ctb #####'
        print '##### Stats #####'
        print '##### RMSE = + str(erc.iter)'

# Boilerplate
if __name__ == '_main__':
    main()
A.3. COMPUTING THE MMSC BASKETS FOR «-INSTRUMENTS
What follows is a synthesis of some of the results derived in Lopez de Prado and Leinweber (2012). Please refer the reader to that publication for the detailed proofs.

For any two subsets \(i\) and \(j\), we can define the following variables:

\[
a = \frac{1}{2} \sigma_{\Delta S_i}^2 + \rho_{\Delta B,\Delta S_j} + \rho_{\Delta B,\Delta S_i} \left(2 \rho_{\Delta S_i,\Delta S_j} - 3 \rho_{\Delta B,\Delta S_j} \rho_{\Delta B,\Delta S_i}\right)
\]

\[
b = -\frac{\sigma_{\Delta S_i}}{\sigma_{\Delta B}} \left(\rho_{\Delta S_i,\Delta S_j} - \rho_{\Delta B,\Delta S_j} \rho_{\Delta B,\Delta S_i}\right)
\]

\[
c = \Delta \rho_{\Delta B,\Delta S_j}
\]

where \(i=1, \ldots, N\) subsets, \(N = \sum_{i=1}^{N-1} \binom{N}{i} = 2^{N-1} - 1\) is the number of subsets, \(\Delta S_i = \sum_{j=1}^{N} \rho_{\Delta S_i,\Delta S_j}\), and \(\rho_{\Delta S_i,\Delta S_j}\) is the vector of holdings of subset \(i\). Then, for \(a \neq 0\):

\[
\delta_i = \begin{cases} 
-\frac{b + \sqrt{b^2 - 4ac}}{2a} & \text{if } b \geq 0 \\
-\frac{b - \sqrt{b^2 - 4ac}}{2a} & \text{if } b < 0
\end{cases}
\]

And for \(a = 0\):

\[
s_i = c \left( \frac{\sigma_{\Delta B}}{\sigma_{\Delta S_i}^2 \rho_{\Delta S_i,\Delta S_j} - \rho_{\Delta B,\Delta S_j} \rho_{\Delta B,\Delta S_i}} \right)
\]

This result allows us to compute the step size \(S_i\) for subset \(i\) that will change the correlation between subset \(i\) and the basket, \(A P_{\Delta B,\Delta S_j}\). This change in subset \(i\) can be backpropagated to the legs involved in that subset by multiplying the holdings of the legs that form subset \(i\) by \((1 + \delta_i)\). For example, a change in subset \(i\) is backpropagated by setting the leg’s holdings to \(\omega_j = \omega_i(1 + S_i)\), \(j=1, \ldots, n\), for the next iteration (see function get_Backpropagate in Section A.4). By doing so, we can balance the exposure to a subset \(j\), even if it is composed of instruments that are not tradable or subject to constraints.

A.4. THE MMSC ALGORITHM
The hedging basket is determined by minimizing the maximum \(P_{\Delta B,\Delta S_j}\), where \(i=1, \ldots, N\) represents any of the \(N\) subsets of legs (including the \(n\) legs). In each iteration, we identify the subset \(j\) for which \(A P_{\Delta B,\Delta S_j} = \frac{1}{N} \left(\sum_{i=1}^{N} \rho_{\Delta B,\Delta S_i}\right)\) is lowest. This is the subset whose correlation we would like to bring down to the average value. This can be done by changing the
holdings of that subset \( j \) or, if it is not tradable or its holding is constrained, by changing any other subset's holdings \( i \). By reducing our exposure to the subsets with above average correlations, we bring that average down until there is no possibility to keep reducing \( \rho_{B,A_S^j} \) without producing some other \( \rho_{A_B,A_S^j} > \rho_{B,A_S^j} \), at which point the solution has been found.

For a trading basket, the algorithm is essentially the same, with the only difference that at each iteration we identify the subset \( j \) for which \( \Delta \rho_{B,A_S^j} = \frac{1}{N} \sum_{i=1}^{N} \rho_{B,A_S^j} - \rho_{B,A_S^j} \) is highest. This Maximizes the Minimum Subset Correlation (the purpose of a trading basket), as opposed to Minimize the Maximum Subset Correlation (the goal of a hedging basket).

```python
#!/usr/bin/env python
# MMSC basket optimization algo
# On 20120516 by MLDP <lopezdeprado@lbl.gov>
import numpy as np
from scipy import linalg
from itertools import combinations
from math import log

class MMSC:
    def __init__(self,covar,maxSubsetSize):
        # Class constructor
        self.covar=covar
        if maxSubsetSize<=covar.shape[0]:maxSubsetSize=covar.shape[0]-1
        self.subsets=self.get_Subsets(range(covar.shape[0]),maxSubsetSize)
        self.weights,self.err,self.iter=0,0,0
        self.score=0

    def solve(self,precision,seed,hedge=True):
        # Compute the MMSC holdings and stats
        w=np.zeros((seed.shape[0],seed.shape[1]))
        weights=np.zeros((seed.shape[0],seed.shape[1]))
        wf=[]
        iterTotal,iterW,aim,msc,n,grad=0,0,0,1,len(w)-1,len(self.subsets)-1,range(3)
        if hedge==False: msc=0 # For a trading basket
        loop=True
        #1) Iterations
        while loop==True:
            iterTotal+=1
            iterW+=1
            #2) Aim size
            if iterW==1:
                aim=max(1,aim/(1+log(n)))+1
            else:
                aim=min(10**6,aim*(1+log(n)))+1
            wf.append(weights)
            #3) Iterate subsets
            for i in range(N):
                #4) Normalize holdings
                w=w/w[0]
```

24
#5) Compute subset correl matrix
scovar=self.get_SubsetCovar(w)
scorrel=get_Corrrel(scovar)

#6) Determine which leg to change
avg=(sum(scorrel[1:-1, 1:-1])/N)
err=[avg-scorrel[j, 1] for j in range(N)]
iErr=[er for er in err] if iErr<0.0
for j in range(N):
  tErr+=err[1]**2
  if (hedge==True and scovar[j, 1]<scorrel[iErr, 1]) or (hedge==False and scovar[j, 1]<scorrel[iErr, 1])
    weights[j]=weight
    iterW=iter
    msc=scorrel[iErr, 1]

#7) Store new optimum
if (hedge==True and scovar[iErr, 1]<msc) or (hedge==False and scovar[iErr, 1]<msc):
  weights[j]=w[j]
  iterW=iter
  msc=scorrel[iErr, 1]

#8) CTR's Taylor expansion
grad[0]=err[iErr]/aim
grad[1]=-(scovar[iErr, 1]/(scovar[1:-1, 1]**.5*(scorrel[iErr, 1]-scorrel[iErr, 1])
    -scorrel[iErr, 1])
grad[2]=.5*(scovar[iErr, 1]/(scovar[1:-1, 1]*scorrel[iErr, 1]+scorrel[iErr, 1])
    -2*scorrel[iErr, 1]-3*scorrel[iErr, 1]*scorrel[iErr, 1]))

#9) Step size
if grad[2]==0:
  delta=(grad[1]**2-4*grad[2]*grad[0])**.5
  if grad[1]==0:
    delta=(grad[1]+delta)/(2.*grad[2])
  else:
    delta=(grad[1]-delta)/(2.*grad[2])
else:
  delta=grad[0]/grad[1]

#10) Backpropagate subset step back to legs
w=self.get_Backpropagate(w, delta, i)

#11) Exit conditions
tErr=err/N**.5 # Root mean squared deviation
if err<10**-precision:loop=False # If exit by convergence
if iterTotal>10**2:precision:loop=False
scovar=self.get_SubsetCovar(weights)
scorrel=get_Corrrel(scovar)
self.weights=self.get_SubsetCovar(weights)
scorrel=get_Corrrel(scovar)
self.iter, self.iter, self.iter = weights, tErr, iterTotal, scorr

#---------------------------------------------
def get_Seed(self, col, hedge):
  # Seed for basket
  if hedge==True:
    # Seeds for hedging basket, based on alternating OLS holdings
    a=np.zeros(shape=(self.covar.shape[0], self.covar.shape[1]))
    a[:, ]=self.covar[:, ]
    a=delete(a, col, 0)
    a=delete(a, col, 1)
    a=np.linalg.inv(a)
    b=self.covar[1:, col]
    b=delete(b, col, 0)
    c=a*b
    c=np.insert(c, col, 1, axis=0)
    return c
else:
    # Seeds for trading basket, based on the leg's betas to the basket
    a=np.zeros(shape=(self.covar.shape[0],1))
    for b in range(self.covar.shape[0]):
        a[b,0]=self.covar[b,b]/self.covar[0,0]
    return a

#---------------------------------------------
def get_BBackpropagate(self,w,delta,subset):
    # Backpropagates subset delta to the legs
    for leg in self.subsets[subset]:
        w[leg]=w[leg]*(1+delta)
    return w

#---------------------------------------------
def get_Subsets(self,iterable,maxSubsetSize):
    # Generate all subsets up to maxSubsetSize, but including the full set
    subsets=[item for item in combinations(iterable,1)]
    for subsetSize in range(2,maxSubsetSize+1):
        for item in combinations(iterable,subsetSize):
            subsets.append(item)
    for item in combinations(iterable,self.covar.shape[0]):
        subsets.append(item)
    return subsets

#---------------------------------------------
def get_SubsetCovar(self,w):
    # Computes covariances among all subsets
    subsetCovar=np.zeros(shape=(len(self.subsets),len(self.subsets)))
    for i in range(len(self.subsets)):
        for j in range(i,len(self.subsets)):
            for k in self.subsets[i]:
                for l in self.subsets[j]:
                    subsetCovar[i,j]=w[k]*w[l]*self.covar[k,l]
                subsetCovar[j,i]=subsetCovar[i,j]
    return subsetCovar

#---------------------------------------------
def get_Sign(number):
    # Sign of the holding, needed to compute CtB
    if number==0:return 0
    if number>0:return 1
    if number<0:return -1

#---------------------------------------------
def get_Correl(covar):
    # Correl matrix from covar
    correl=np.zeros(shape=(covar.shape[0],covar.shape[1]))
    for i in range(covar.shape[0]):
        for j in range(i,covar.shape[1]):
            correl[i,j]=covar[i,j]*(covar[i,i]*covar[j,j])**.5
        correl[j,i]=correl[i,j]
    return correl

#---------------------------------------------
def main():
    #1) Input parameters --- to be changed by the user
    precision=3
    covar=np.matrix("846960.805351971,515812.899769821,403177.059835136; 515812.899769821,351407.396443653,280150.614979364; 403177.059835136,280150.614979364,232934.832710412")

Figures 16 and 17 show how holdings converge to their optimal values for the examples of MMSC hedging and trading baskets discussed in Section 3 and 4.

This algorithm can also be used to compute the MDR solution, by setting the parameter `maxSubsetSize=1`. This is equivalent to a MMSC optimization where subsets of more than one leg are ignored. `maxSubsetSize` can also be used to skip evaluating correlations for subsets of larger size, which is convenient should \( N \) reach an impracticable order of magnitude.

A.5. COVARIANCE CLUSTERING

The number of subsets follows a power law on the number \( n \) of instruments involved: \( N = \sum_{i=1}^{n-1} \binom{n}{i} = 2(2^{n-1} -1) \), where we exclude the empty set and the full set. For a sufficiently large \( n \), the number of subsets \( N \) to be evaluated per iteration makes MMSC impracticable. But because \( n \) is large, it also becomes more likely that some of the instruments involved are highly correlated. An approach commonly used to reduce the dimension of a problem applies PCA to identify which orthogonal directions add least variance, so that they can be dropped. There are at least three arguments for discarding such procedure in the context of balanced baskets: First, a key reason for favoring balanced baskets was precisely that they did not require a change of basis, so that the solution could be intuitively connected to the original instruments and subsets of them. Second, "dropping" dimensions involves the loss of information, even if minimal. Third, in a capital allocation context, we cannot short funds or portfolio managers.
In this section we propose a new method for reducing the dimension of a covariance matrix without requiring a change of basis or dropping dimensions. The intuition is to identify which tuples of matrix columns point in neighboring directions, in which case they are redundant and can be clustered together. This can be evaluated by carrying out an eigen-decomposition of the matrix and evaluating which columns have the largest loadings in the orthogonal directions that contribute least variance. When such tuples are clustered together they form a new column that is less redundant, thus contributing to a more parsimonious distribution of the variance across the orthogonal directions. The actual clustering is done by recursively aggregating any two columns \((ij)\), applying the property that

\[
\sigma_{i+j,k} = E[(i+j - E[i+j])(k - E[k])] \\
= E[(i - E[i] + j - E[j]X_k - E[k])]
\]

(41)

\[
\sigma_{i+j,i+} = \sigma_i,i+ + \sigma_{i,j} + 2\sigma_{i,j}
\]

(42)

The algorithm identifies what column aggregation minimizes the matrix’s condition number at each iteration. In this way, we reduce \(n\) and, more importantly, \(N\), while making the covariance matrix less singular. One possible procedure would consist in computing, by brute force, all possible clustering outcomes, and determine the one for which the condition number is minimal. That would require \(\sum_{m=1}^{m=n} \binom{n}{i}\) covariance clustering operations, where \(m\) is the number of dimensions reduced. This is a very large number, considering that the reason for clustering was to avoid having to evaluate \(N\) subset correlations per iteration. Applying brute force does not seem to alleviate our computational problem. An alternative clustering strategy would consist in sequentially pairing columns of the covariance matrix, so that at each iteration we minimize the condition number. That strategy only requires \(\sum_{i=1}^{m=n} \binom{n-i+1}{2} \ll \sum_{i=1}^{m=n} \binom{n}{i}\) covariance clustering operations.

More precisely, suppose a covariance matrix \(W\) with elements \(\sigma_{i,j}\) and \(i,j=1,...,n\). We denote the matrix’s \(z\)-th eigenvalue by \(\lambda_z\), and its condition number by \(c = \frac{\lambda_z}{\lambda_1}\). We would like to reduce \(V\)’s order to a more manageable \(n^* = n - m\), where \(2 \leq n^* < n\). The following algorithm clusters \(m\) elements of \(V\) until such requirement is met:

1. If \(n = n^*\), return \(V\) and exit.
2. For each pair \((i,j)\), \(i < j\) of columns of \(V\),
   a. Let \(\tilde{V} = V\) be a copy of \(V\).
   b. Insert in \(\tilde{V}\)
      i. column and row, \(n+1\), with \(k\) elements, \(\sigma_{k,n+1} = \sigma_{k,i} + \sigma_{k,j}\).
      ii. diagonal element \(\sigma_{k,n+1} = \sigma_{i,n+1} + \sigma_{j,n+1} = \sigma_{i,i} + \sigma_{j,j} + 2\sigma_{i,j}\).
   c. Strike down columns and rows \(ij\), giving \(\tilde{V}\) an order \(\tilde{n} = n - 1\).
   d. Compute \(\tilde{\lambda}_z\) for the resulting \(\tilde{V}\).

28
e. Store the value $c_{i,j} = \frac{\max \{\lambda_i\}}{\min \{\lambda_i\}} = \frac{\lambda_i}{\lambda_N}$

3. Determine the $(i^*,j^*) = \arg \max_{i < j \in \mathcal{V}} c_{i,j}$

4. Prepare for the next iteration
   a. Replace $\mathcal{V}$ with the matrix $\tilde{\mathcal{V}}$ which clustered together elements $(i^*, j^*)$.
   b. Set $n = \tilde{n}$.

5. Loop to 1.

The outcome is a clustered covariance matrix with three crucial properties:

1. It has a smaller dimension, which enables calculations to be carried out in a feasible amount of time.
2. It has a smaller condition number, and consequently it is less singular. Financial applications typically require the inversion of the covariance matrix, and near-singular covariance matrices are a major source of numerically unstable results.
3. This clustering of the covariance matrix forms a disjoint-set data structure, whereby each original element will end up in only one cluster. Cluster constituents are equally (and positively) weighted, thus the sum of elements of the covariance matrix is kept constant across iterations. This is a key difference with respect to PCA, where each constituent forms part of each component, and weights are allowed to be negative.

[FIGURE 18 HERE]

A numerical example with illustrate how the algorithm works. Suppose that we are given a covariance matrix of the 88 most liquid futures contracts. Computing a MMSC hedging basket of that dimension seems quite impracticable. Even though no column may be derived as an exact linear combination of the rest, it is very likely that one column spans a vector in a very close angle to another column or a combination of columns. The algorithm above will identify those situations and form clusters of very similar instruments. Figure 18 shows how, beginning with a numerically ill-conditioned covariance matrix of 88 instruments (determinant greater than $10^{305}$, condition number of $486,546.1293$), the covariance matrix is greatly improved by clustering. For example, for $n^* = 20$, the condition number has dropped to a fraction of its original value ($677.41739$). As a result, the covariance matrix will now be less numerically ill-conditioned, and $n$ can be reduced to a value for which the MMSC solution can be computed.

We believe this algorithm will prove useful in many financial applications beyond basket construction, like in capital allocation and portfolio optimization problems. What follows is an implementation in Python code. The user simply needs to adjust the statement `path='E:\HFT\Covariance.csv'` with the path where a covariance matrix is stored in csv format.

```python
#!/usr/bin/env python
# Covariance clustering algo
# On 20120516 by MLdP <lopezdeprado@lbl.gov>

import numpy as np
from itertools import combinations
from scipy import delete

c = np.load('E:\HFT\Covariance.csv')
```

```
class ClusterCov:
    def __init__(self, cov, minSize):
        # Class constructor
        self.cov = cov
        self.minSize = minSize
        self.cn = []
        self.clusters = []

    def solve(self, clusterSize=2):
        # Determine the optimal clustering of the covariance matrix
        size = self.cov.shape[0]
        bundles = []
        while self.cov.shape[0] > self.minSize:
            cn = []
            comb = self.comb(range(self.cov.shape[0]), clusterSize)
            for trial in comb:
                # Copy matrix
                cov = self.cov[::]
                # Bundle
                cov = self.bundle(cov, trial)
                # Compute condition number
                evals = np.linalg.eigvalsh(cov)
                cn_ = max(evals) / min(evals)
                cn.append(cn_)
                # Determine optimal cluster
                sol = self.findSolution(cn)
                if sol <= 0:
                    break
                # Store solution
                self.cn.append(sol)
                bundles.append(comb[comb.index(self.cn[-1])])
                self.cov = self.bundle(self.cov[::], bundles[-1])
                self.clusters = self.getClusters(bundles, size)
                return

    def bundle(self, cov, indices):
        # Bundle the covariance matrix
        diag, a = 0, 0
        # Add clustering column and row
        for i in indices:
            a += cov[:, i]
        # Compute diagonal element
        for i in indices:
            diag += a[i]
        cov = self.expandCov(cov, a, diag)
        # Remove clustered columns and rows
        for i in range(len(indices)):
            cov = delete(cov, indices[i]-1, 0)
            cov = delete(cov, indices[i]-1, 1)
        return cov

    def expandCov(self, cov, array, diag):
        # Expand the covariance matrix
        b = array[::]
def comb(self, iterable, groupSize):
    # Generate all combinations
    comb = [item for item in combinations(iterable, groupSize)]
    return comb

#---------------------------------------------------

def findSolution(self, cn):
    # Find the optimal solution among candidates
    sol = max(cn)
    for i in cn:
        if i < sol and i > 0:
            sol = i
    return sol

#---------------------------------------------------

def getClusters(self, bundles, size):
    # Determine clusters' constituents
    clusters = range(size)
    for i in bundles:
        a = []
        for j in i:
            a.append(clusters[j])
        clusters.append(a)
        for j in range(len(i)):
            clusters.pop(i[j]-j)
    return clusters

#---------------------------------------------------
#---------------------------------------------------

def main():
    # Example using the ClusterCov class
    path = 'E:\HT\Covariance.csv'
    # 0) Parameters
    minSize = 2
    # 1) Load the covariance matrix
    cov = np.genfromtxt(path, delimiter=',')  # numpy array
    # 2) Cluster
    cluster = ClusterCov(cov, minSize)
    cluster.solve()
    # 3) Report results
    print(cluster.cov.shape)
    print(cluster.cov)
    print(cluster.cn)
    print(cluster.clusters)
    return

#---------------------------------------------------
# Boilerplate
if name == '__main__': main()
FIGURES

<table>
<thead>
<tr>
<th>FA1 Index</th>
<th>ESI Index</th>
</tr>
</thead>
<tbody>
<tr>
<td>846960.8</td>
<td>515812.9</td>
</tr>
<tr>
<td>515812.9</td>
<td>351407.4</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>FA1 Index</th>
<th>ESI Index</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.000000</td>
<td>0.945486</td>
</tr>
<tr>
<td>0.945486</td>
<td>1.000000</td>
</tr>
</tbody>
</table>

*Figure 1 - Covariance and Correlation matrices for the proposed hedging problem (two instruments)*

---

We can plot the original position \( (h_1) \) as a vector in a two-dimensional space characterized by the eigen-decomposition of the covariance matrix from Figure 1. This vector lays in a direction very close to the first eigenvector, which is associated with the market risk component. If we hedge "FA1 Index" with "ESI Index" using the MV or OLS procedures, the basket's vector spans in a direction very close to the second eigenvector, which is associated with the spread risk. The MMSC and ERC procedures give a solution very similar to PCA's without requiring a basis change.
Figure 3 - Covariance and Correlation matrices for the proposed hedging problem (three instruments), with the corresponding eigenvectors and eigenvalues matrices

<table>
<thead>
<tr>
<th>INSTRUMENTS</th>
<th>HOLDINGS</th>
<th>CtB</th>
<th>CtR</th>
</tr>
</thead>
<tbody>
<tr>
<td>FA1 Index</td>
<td>846960.8</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>ESI Index</td>
<td>515812.9</td>
<td>0.97</td>
<td>0.33</td>
</tr>
<tr>
<td>DM1 Index</td>
<td>403177.1</td>
<td>0.98</td>
<td>0.33</td>
</tr>
</tbody>
</table>

Subsets 1 2 3 1,2 1,3 2,3
1,2,3 0.135702 0.051251 0.079026 0.156600 0.102101 0.264190

Figure 4 - One possible ERC solution

The second table in Figure 4 reports the correlation of the basket to each subset of legs, where the subset is identified by the legs it is made of. For example, subset (1,3) is composed of "ESI Index" and "DM1 Index".

<table>
<thead>
<tr>
<th>INSTRUMENTS</th>
<th>HOLDINGS</th>
<th>CtB</th>
<th>CtR</th>
</tr>
</thead>
<tbody>
<tr>
<td>FA1 Index</td>
<td>1000</td>
<td>0.97</td>
<td>0.33</td>
</tr>
<tr>
<td>ESI Index</td>
<td>1515</td>
<td>0.99</td>
<td>0.33</td>
</tr>
<tr>
<td>DM1 Index</td>
<td>1885</td>
<td>0.98</td>
<td>0.33</td>
</tr>
</tbody>
</table>

Subsets 1 2 3 1,2 1,3 2,3
1,2,3 0.969597 0.993448 0.980552 0.995030 0.998350 0.992132

Figure 5 - An alternative ERC solution
Although the DRP method is not a balanced basket approach, we report its solution here to illustrate the point that procedures which require a change of basis may yield rather unintuitive solutions.

**Figure 6 - The DRP solution**

<table>
<thead>
<tr>
<th>Instruments</th>
<th>Holdings</th>
<th>CtB</th>
<th>CtR</th>
</tr>
</thead>
<tbody>
<tr>
<td>FA1 Index</td>
<td>1000</td>
<td>-0.49</td>
<td>-0.12</td>
</tr>
<tr>
<td>ES1 Index</td>
<td>18338</td>
<td>-0.62</td>
<td>-1.75</td>
</tr>
<tr>
<td>DM1 Index</td>
<td>-29896</td>
<td>0.77</td>
<td>2.87</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Subsets</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>1,2</th>
<th>1,3</th>
<th>2,3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1,2,3</td>
<td>-0.492806</td>
<td>-0.619724</td>
<td>0.765364</td>
<td>-0.612224</td>
<td>0.778720</td>
<td>0.983147</td>
</tr>
</tbody>
</table>

**Figure 7(a) - Risk as we move away from DRP and towards the eigenvectors**

Risk increases as we approach a basket pointing in the direction of an eigenvector, except in the case of the eigenvector associated with the lowest eigenvalue. This is because eigenvectors are the critical points of the Rayleigh quotient $\frac{\omega'Y\omega}{\omega'\omega}$, where the numerator is the variance of the basket.
Figure 7(b) - Risk as we move away from DRP and towards the first eigenvector, in a risk-on and risk-off environment

Figure 8 - The MDR solution

<table>
<thead>
<tr>
<th>INSTRUMENTS</th>
<th>HOLDINGS</th>
<th>CtB</th>
<th>CtR</th>
</tr>
</thead>
<tbody>
<tr>
<td>FA1 Index</td>
<td>1000</td>
<td>0.08</td>
<td>0.16</td>
</tr>
<tr>
<td>ES1 Index</td>
<td>-4736</td>
<td>0.08</td>
<td>0.49</td>
</tr>
<tr>
<td>DM1 Index</td>
<td>4031</td>
<td>0.08</td>
<td>0.34</td>
</tr>
</tbody>
</table>

Subsets 1, 2, 3, 1,2, 1,3, 2,3

<table>
<thead>
<tr>
<th>Subsets</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>1,2</th>
<th>1,3</th>
<th>2,3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1,2,3</td>
<td>0.075176</td>
<td>0.075176</td>
<td>0.075176</td>
<td>0.142953</td>
<td>0.076735</td>
<td>0.362830</td>
</tr>
</tbody>
</table>
MMSC
INSTRUMENTS HOLDINGS CtB CtR
FA1 Index 1000 0.19 0.50
ES1 Index -3632 0.03 0.22
DM1 Index 2690 0.07 0.28

Subsets 1 2 3 1,2 1,3 2,3
1,2,3 0.185136 0.034164 0.074562 0.185136 0.123218 0.185136

Figure 9 - The MMSC solution

Figure 10 - Factor loadings for all subsets to the three principal components
FA1 Index 846976.4 515717.6 403266.3
ESI Index 515717.6 351987.8 279607.2
DM1 Index 403266.3 279607.2 233443.7

FA1 Index 1.000000 0.944523 0.906912
ESI Index 0.944523 1.000000 0.975424
DM1 Index 0.906912 0.975424 1.000000

Figure 11 - Covariance and Correlation matrices following a structural break (25% increase of A_{3,3})

<table>
<thead>
<tr>
<th>Subsets</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>1,2</th>
<th>1,3</th>
<th>2,3</th>
</tr>
</thead>
<tbody>
<tr>
<td>ERC</td>
<td>0.124551</td>
<td>0.063642</td>
<td>0.091317</td>
<td>0.168826</td>
<td>0.105856</td>
<td>0.311802</td>
</tr>
<tr>
<td>MDR</td>
<td>0.069397</td>
<td>0.085319</td>
<td>0.087978</td>
<td>0.154587</td>
<td>0.083732</td>
<td>0.408664</td>
</tr>
<tr>
<td>MMSC</td>
<td>0.170800</td>
<td>0.047725</td>
<td>0.087045</td>
<td>0.196876</td>
<td>0.124610</td>
<td>0.231720</td>
</tr>
</tbody>
</table>

Figure 12 - Impact of the structural break (25% increase of A_{3,3}) on the exposures (subset correlations) of the previously computed baskets

<table>
<thead>
<tr>
<th>INSTRUMENTS</th>
<th>HOLDINGS</th>
<th>Ctb</th>
<th>Ctr</th>
</tr>
</thead>
<tbody>
<tr>
<td>FA1 Index</td>
<td>1000</td>
<td>0.98</td>
<td>0.50</td>
</tr>
<tr>
<td>ESI Index</td>
<td>0</td>
<td>0.99</td>
<td>0.00</td>
</tr>
<tr>
<td>DM1 Index</td>
<td>1907</td>
<td>0.98</td>
<td>0.50</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Subsets</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>1,2</th>
<th>1,3</th>
<th>2,3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1,2,3</td>
<td>0.976655</td>
<td>0.985343</td>
<td>0.976655</td>
<td>0.976655</td>
<td>1.000000</td>
<td>0.976655</td>
</tr>
</tbody>
</table>

Figure 13 - The MMSC trading basket
This figure illustrates how the ERC algorithm found the optimal hedging basket after the instruments' holdings converged over a number of iterations to a combination such that each leg contributed the same amount of risk.
This figure illustrates how the ERC algorithm found the optimal trading basket after the instruments' holdings converged over a number of iterations to a combination such that each leg contributed the same amount of risk.
This figure illustrates how the MMSC algorithm found the optimal hedging basket after the instruments' holdings converged over a number of iterations to a combination that minimized the maximum subset correlation.
This figure illustrates how the MMSC algorithm found the optimal trading basket after the instruments’ holdings converged over a number of iterations to a combination that maximized the minimum subset correlation.
This figure illustrates how clustering the covariance matrix improves its numerical condition. We can reduce the number of instruments involved in the MMSC calculation to a value $n$ for which the $N$ subset correlations can be computed within a reasonable timeframe.
REFERENCES

DISCLAIMER

The views expressed in this paper are those of the authors and not necessarily reflect those of Tudor Investment Corporation. No investment decision or particular course of action is recommended by this paper.
GENERALIZED OPTIMAL TRADING TRAJECTORIES:
A FINANCIAL QUANTUM COMPUTING APPLICATION

Marcos Lopez de Prado †

This version: March 7, 2015

FIRST DRAFT – DO NOT CITE WITHOUT THE AUTHOR’S PERMISSION

† Senior Managing Director, Guggenheim Partners, New York, NY 10017. Research Affiliate, Computational Research Division, Lawrence Berkeley National Laboratory, Berkeley, CA 94720. E-mail: lopezdeprado@lbl.gov, Website: www.QuantResearch.org

I would like to acknowledge useful comments from David H. Bailey (Lawrence Berkeley National Laboratory), Jose Blanco (Credit Suisse), Jonathan M. Borwein (University of Newcastle), Peter Carr (Morgan Stanley, NYU), Matthew D. Foreman (University of California, Irvine), Phil Goddard (IQBit), Andrew Landon (IQBit), Riccardo Rebonato (PIMCO, University of Oxford), Luis Viceira (HBS) and Jim Qiji Zhu (Western Michigan University).

The statements made in this communication are strictly those of the authors and do not represent the views of Guggenheim Partners or its affiliates. No investment advice or particular course of action is recommended. All rights reserved.
GENERALIZED OPTIMAL TRADING TRAJECTORIES: A FINANCIAL QUANTUM COMPUTING APPLICATION

ABSTRACT

Generalized dynamic portfolio optimization problems have no known closed-form solution. These problems are particularly relevant to large asset managers, as the costs from excessive turnover and implementation shortfall may critically erode the profitability of their investment strategies.

In this brief note we demonstrate how this financial problem, intractable to modern supercomputers, can be reformulated as an integer optimization problem. Such representation makes it amenable to quantum computers.

Keywords: High-performance computing, integer optimization, quantum computing, adiabatic process.

JEL Classification: G0, Gl, G2, G15, G24, E44.
AMS Classification: 91G10, 91G60, 91G70, 62C, 60E.
1. INTRODUCTION
A supercomputer is a mainframe computer able to perform an extremely large number of floating point operations per second (FLOPS). This is generally achieved following one of two approaches. In the first approach, a problem is divided into many small problems that can be solved in parallel. This is the strategy used by distributed computing or hyper-threaded architectures, such as cloud systems or GPUs. The second approach takes advantage of the topological configuration of a system to save time in I/O and other intensive operations. This is the key advantage of computer clusters. Moore's law, which states that the number of transistors on a chip will double approximately every two years, means that a system qualifies as a supercomputer for a relatively short period of time. The TOP500 project keeps track of the 500 fastest supercomputers in the world. As of June of 2014, 233 of these systems are located in the United States, and 76 in China.

Combinatorial optimization problems can be described as problems where there is a finite number of feasible solutions, which result from combining the discrete values of a finite number of variables. As the number of feasible combinations grows, an exhaustive search becomes impractical. The traveling salesman problem is an example of a combinatorial optimization problem that is known to be NP-hard, i.e. the category of problems that are at least as hard as the hardest problems solvable is nondeterministic polynomial time.

What makes an exhaustive search impractical is that standard computers evaluate and store the feasible solutions sequentially. But what if we could evaluate and store all feasible solutions at once? That is the goal of quantum computers. Whereas the bits of a standard computer can only adopt one of two possible states (\{0,1\}) at once, quantum computers rely on qubits, which are memory elements that may hold a linear superposition of both states. In theory, quantum computers can accomplish this thanks to quantum mechanics. A qubit can support currents flowing in two directions at once, hence providing the desired superposition. D-Wave, a commercial quantum computer designer, is planning to produce a 2048-qubit system in the year 2015. This linear superposition property is what makes quantum computers ideally suited for solving NP-hard combinatorial optimization problems.

In this note we will show how a dynamic portfolio optimization problem subject to generic transaction cost functions can be represented as a combinatorial optimization problem, tractable by quantum computers. Unlike Garleanu and Pedersen [2012], we will not assume that the returns are IID Normal. Furthermore, we will maximize the Probabilistic Sharpe Ratio (PSR), thus taking into account our confidence on the risk estimates involved. This problem is particularly relevant to large asset managers, as the costs from excessive turnover and implementation shortfall may critically erode the profitability of their investment strategies.

2. THE OBJECTIVE FUNCTION
Consider a set on assets \( X = \{ x_i \}, i = 1, ..., N \). We do not assume that returns of \( X \) follow a multivariate Normal distribution, or even IID. Our only prerequisite is that returns of \( X \) are ergodic. Their expected values change over \( H \) time horizons, with \( \mu \) representing a \( N \times H \) matrix of forecasted means.
We define a trading trajectory as a \( N \times H \) matrix \( \omega \) that determines the proportion of capital allocated to each of the \( N \) assets over each of the \( H \) horizons. This means that, given a trading trajectory \( \omega \), we can compute a stream of investment returns \( r \), as

\[
r = \text{diag}(\mu^T \omega) - \tau(\omega)
\]

where:

- \( \tau(\omega)_0 = \sum_{i=1}^{TM} c_n \sqrt{|\omega_{n,1} - \eta_1|} \)
- \( \tau(\omega)_h = \sum_{n=1}^{H} c_n \sqrt{|\omega_{n,h} - \eta_n, h-1|} \), for \( h = 1, \ldots, H \).
- \( \omega^i_n \) is the initial holding of instrument \( n, n = 1, \ldots, N \).

\( \tau(\omega) \) is a \( H \times l \) vector of transaction costs. In words, the transaction costs associated with each asset is the sum of the square roots of the changes in capital allocations, re-scaled by an asset-specific factor \( c_n \). Thus, a \( N \times l \) vector \( C \) determines the relative transaction cost across assets.

The Probabilistic Sharpe Ratio (PSR) associated with \( r \) can be computed as

\[
PSR[r] = Z \left[ \frac{\tilde{SR}\sqrt{H} - 1}{\sqrt{1 - \hat{\gamma}_3\tilde{SR} + \hat{\gamma}_4 - \frac{1}{4}\tilde{SR}^2}} \right]
\]

where

- \( Z \) is the CDF of the Standard Normal distribution.
- \( \tilde{SR} = E[r]/V[r]^{-1/2} \) is the estimated Sharpe ratio of \( r \), where \( E[. \] \) is the expectation operator, and \( V[. \] \) is the variance operator.
- \( \hat{\gamma}_3 \) is the estimated skewness of \( r \).
- \( \hat{\gamma}_4 \) is the estimated kurtosis of \( r \).

Note that in this analysis we did not make use of a covariance matrix. The reason is, that information is embedded in the \( H \) dimension of \( \mu \). Furthermore, the objective function already penalizes risk through higher moments, and a covariance matrix would reduce the scope of our approach. Furthermore, alternative \( \mu \) can be simulated incorporating various degrees of noise.

3. THE PROBLEM
We would like to compute the optimal trading trajectory that solves the problem

\[
\max_{\omega} PSR[r]
\]
Note that non-continuous transaction costs are embedded in \( r \). Next, we will show how to calculate solutions without making use of any functional property of the objective function (hence the "generalized" nature of this approach).

4. AN INTEGER OPTIMIZATION APPROACH

The generality of this problem makes is intractable to standard convex optimization techniques. Our solution strategy is to discretize it so that it becomes amenable to integer optimization. This in turn allows us to use quantum computing technology to find the optimal solution.

Suppose that we count with \( K \) units of capital, to be allocated among the \( N \) assets. This is a classic integer partitioning problem studied in number theory and combinatorics, and by Hardy and Ramanujan in particular, see Johansson [2012]. The only difference is, order is relevant to the partition. For example, if \( K=6 \) and \( N=3 \), partitions \((1,2,3)\) and \((3,2,1)\) must be treated as different. This means that we must consider all distinct permutations of each partition (obviously \((2,2,2)\) does not need to be permuted).

An efficient algorithm to generate all partitions is provided by Kelleher and O'Sullivan [2009].

```
#!/usr/bin/env python
# Kelleher-O'Sullivan 2009 partition generating function
def intPart2(n):
    a=[0 for i in range(n+1)]
    k=1
    y=n-1
    while k!=0:
        x=a[k-1]+1
        k=1
        while 2*x<=y:
            a[k]=x
            y-=x
            k+=1
        l=k+1
        while x==y:
            a[k]=x
            a[l]=y
            yield a[:k+2]
            x+=1
            y-=1
            a[k]=x+y
            y=x+y-1
            yield a[:k+1]
```

Snippet 1 - Kelleher-O'Sullivan Integer Partition generating function

If we add all permutations of each partition, and standardize the outputs (filling blanks when an asset does not receive allocation), we can compute the set of all possible realizations of a vector column of \( r \). This is accomplished in Snippet 2.
5. A NUMERICAL EXAMPLE
Following our previous example, for \( K=6 \) and \( N=3 \), each column of \( f(t) \) can adopt one of the following 28 arrays (run the code in Snippet 2):

\[
[[1, 1, 4], [1, 4, 1], [4, 1, 1], [1, 2, 3], [1, 3, 2], [2, 1, 3], [3, 1, 2], [2, 3, 1], [3, 2, 1], [1, 5, 0], [1, 0, 5], [5, 1, 0], [0, 1, 5], [5, 0, 1], [0, 5, 1], [2, 2, 2], [2, 4, 0], [2, 0, 4], [4, 2, 0], [0, 2, 4], [4, 0, 2], [0, 4, 2], [3, 3, 0], [3, 0, 3], [6, 0, 0], [0, 6, 0], [0, 0, 6]]
\]

Since \( f(t) \) has \( H \) columns, there are \( 2^H \) possible trajectory matrices \( f(t) \). For each of those possible trajectories, we can compute \( \tau[\omega] \rightarrow r \rightarrow PSR[r] \). This procedure is highly computationally intensive. Add the possibility of simulating multiple matrices \( \mu \) for various risk scenarios, and the problem is clearly intractable using standard methods. However, our discretization makes it amenable to quantum computers.

6. REFERENCES
ABSTRACT

Execution traders know that market impact greatly depends on whether their orders lean with or against the market. We introduce the OEH model, which incorporates this fact when determining the optimal trading horizon for an order, an input required by many sophisticated execution strategies. From a theoretical perspective, OEH explains why market participants may rationally "dump" their orders in an increasingly illiquid market. OEH is shown to perform better than participation rate schemes and VWAP strategies. We argue that trade side and order imbalance are key variables needed for modeling market impact functions, and their dismissal may be the reason behind the apparent disagreement in the literature regarding the functional form of the market impact function. Our backtests suggest that OEH contributes substantial "execution alpha" for a wide variety of futures contracts. An implementation of OEH is provided in Python language.

Keywords: Liquidity, flow toxicity, broker, VWAP, market microstructure, adverse selection, probability of informed trading, VPIN, OEH.

JEL codes: C02, D52, D53, G14, G23.
1. INTRODUCTION
Optimal execution strategies compute a trajectory that minimizes the shortfall cost of acquiring or disposing of a position in an asset. Well-known contributions to this subject are Perold [1998], Bertsimas and Lo [1998], Almgren and Chriss [2000], and Kissell and Glantz [2003], to cite only a few. These strategies provide an abstract and general framework to model the costs that trading imposes on a particular investor. One drawback of this abstraction is that it does not explicitly define how market impact arises from a trade's perturbation of the liquidity provision process. This paper attempts to cover that gap in the market impact literature, offering an answer to some of its long-standing questions. For example, execution traders know that market impact greatly depends on whether their orders lean with or against the market. The execution strategy literature, however, has not yet incorporated the two variables associated with this phenomenon: order side and market imbalance.

The PIN theory (Easley et al. [1996]) shows that market makers adjust the range at which they are willing to provide liquidity based on their estimates of the probability of being adversely selected by informed traders. Easley, Lopez de Prado and O'Hara [2012] show that, in high frequency markets, this probability can be accurately approximated as a function of the absolute order imbalance (absolute value of buy volume minus sell volume). Suppose that we are interested in selling a large amount of E-mini S&P500 futures in a market that is imbalanced towards sells. Because our order is leaning with previous orders, it reinforces market makers' fears that they are being adversely selected, and that their current (long) inventory will be harder to liquidate without incurring a loss. As a result, market makers will further widen the range at which they are willing to provide liquidity, increasing our order's market impact. Alternatively, if our order were on the buy side, market makers would narrow their trading range as they believe that their chances of turning over their inventory improve, in which case we would experience a lower market impact than with a sell order. Thus, order imbalance and market maker behavior set the stage for understanding how orders fare in terms of execution costs. Our goal in this paper is to apply this theory in the context of market impact and execution strategies.

Besides reducing transaction costs, there are a number of reasons why traders care about not increasing the order imbalance. First, Federal Regulation and Exchange Circulars limit a trader's ability to disrupt or manipulate market activity. A trading strategy that disrupts the market can bring fines, restrictions and sanctions. Second, traders often will have to come back to the market shortly after completing the initial trade. If a trader got great fills in the previous trade at the expense of eroding liquidity, the previous trade's gains may transform into losses on the successive trades. Third, the position acquired will be marked-to-market, so post-trade liquidity conditions will be reflected in the unrealized P&L of the position. Thus, it would be useful to

---

4 We could envision a future in which regulators and exchanges limit the assets under management of an investment firm based on that participant's technology as it relates to the disruption of the liquidity provision process.

5 See, for example, the "stupid algo" rules recently introduced by the Deutsche Borse (discussed in "Superfast traders feel heat as Bourses act", Financial Times, March 5, 2012). The CFTC and SEC have also recently introduced explicit rules relating to algorithmic impact on markets.

6 Even if there are no successive trades, leaving a footprint leaks information that can be recovered by competitors and used against that trader in futures occasions. See Easley et al. [2012c] for examples.

7 If a buyer pushes the mid-price up at the expense of draining liquidity, she may find that the liquidation value of that position implies a loss.
determine the amount of volume needed to "conceal" a trade so that it leaves a minimum footprint on the trading range.

Many optimal execution strategies make the assumption that liquidity cost per share traded is a linear function of trading rate or of the size of the block to be traded. This seems an unrealistic assumption and it is the motivation for Almgren's [2003] study of optimal execution. The model we introduce here also incorporates a nonlinear response in trading cost to size of the trade. We are not aware of other methodologies explicitly taking into account the presence of asymmetric information in determining the optimal execution horizon. In this context, we find that the side of the trade is as important as its size.

Our Optimal Execution Horizon model, which for brevity we denote OEH henceforth, is related to a growing number of recent studies concerned with execution in the context of high frequency markets and tactical liquidity provision. Schied and Schoneborn [2009] point out that the speed by which the remaining asset position is sold can be decreasing in the size of the position, but increasing in the liquidity price impact. Gatheral and Schied [2011] extend the Almgren-Chriss framework to an alternative choice of the risk criterion. Forsyth [2011] formulates the trading problem as an optimal stochastic control problem, where the objective is to maximize the mean-variance tradeoff as measured at the initial time. Bayraktar and Ludkovski [2011] propose an optimal trade execution scheme for dark pools.

OEH does not replace or supersede minimum market impact strategies. On the contrary, it is complementary to them. OEH does not address the question of how to slice the orders (create a "trading schedule"), which has been studied by Hasbrouck and Schwartz [1988], Berkowitz et al. [1988], Grinold and Kahn [1999], Konishi and Makimoto [2001] among others previously cited. Our main concern is with understanding how transaction costs are derived from the impact that a trade has on liquidity providers. There are some connections, however, between OEH and the previous studies. Like earlier models, OEH minimizes the impact on liquidity subject to a timing risk. From the standard VWAP (Madhavan [2002]) to sophisticated nonlinear approaches (Almgren [2003]; Dai and Zhong [2012]), many execution strategies require as an input the trading horizon, which is typically exogenous to those models. One of our contributions is to provide a framework for determining this critical input variable from a new perspective: Informational leakage. In doing so, we stress the importance of modeling the order side and its asymmetric impact on the order imbalance.

The current paper is organized as follows. Section 2 reviews how order imbalance and trading range are related. Section 3 explains why our trading actions leave a footprint on the market makers' trading range. Section 4 incorporates timing risk and risk aversion to our analysis. Section 5 develops the Optimal Execution Horizon (OEH) algorithm that determines the optimal horizon over which to execute a trade. Section 6 presents several stylized cases. Section 7 argues why the apparently irrational behavior of some market participants during the "flash crash" may be explained by OEH. Section 8 compares OEH's performance with that of trading schemes that target a volume participation rate. Section 9 discusses how our model relates to alternative functional forms of the market impact function. Section 10 provides backtests of the performance of EOH, and compares it with a VWAP execution strategy. Section 11 summarizes our conclusions. Appendices 1 provides a proof.
Appendix 2 offers a generalization of our approach. Appendix 3 contains an implementation of Appendices 1 and 2 in Python language. Appendix 4 provides a procedure for estimating market imbalance.

2. TRADING RANGE AND FLOW TOXICITY
We begin by summarizing a standard sequential-trade market microstructure approach to determining the trading range for an asset. In a series of papers, Easley et al. [1992a, 1992b, 1996] demonstrate how a microstructure model can be estimated for individual assets using trade data to determine the probability of information-based trading, PIN. This microstructure model views trading as a game between liquidity providers and traders (position takers) that is repeated over trading periods \( i = 1, \ldots, I \). At the beginning of each period, nature chooses whether an information event occurs. These events occur independently with probability \( \alpha \). If the information is good news, then informed traders know that by the end of the trading period the asset will be worth \( \overline{S} \); and, if the information is bad news, that it will be worth \( \underline{S} \), with \( \overline{S} > \underline{S} \). Good news occurs with probability \( (1 - \delta) \) and bad news occurs with probability \( \delta \). After an information event occurs or does not occur, trading for the period begins with traders arriving according to Poisson processes. During periods with an information event, orders from informed traders arrive at rate \( \mu \). These informed traders buy if they have seen good news, and sell if they have seen bad news. Every period, orders from uninformed buyers and uninformed sellers each arrive at rate \( \epsilon \).

Easley, Kiefer, O'Hara and Paperman [1996] argue that, for the natural case that \( \delta = \frac{1}{2} \), the trading range at which market makers are willing to provide liquidity is

\[
\Sigma = \frac{\alpha \mu}{\alpha \mu + 2 \epsilon} \overline{S} - \underline{S}
\]  

(1)

This trading range gives the market maker’s targeted profit per portfolio turnover. The first term in Eq. (1) is the probability that a trade is information-based. It is known as PIN. Easley, Engle, O’Hara and Wu [2008] show that expected absolute trade imbalance can be used to approximate the numerator of PIN. They demonstrate that, letting \( V^S \) and \( V^B \) represent sell and buy volume, respectively, \( E[|\overline{S} - V^B|] \approx \alpha \mu \) for a sufficiently large \( \mu \). Easley, Lopez de Prado and O’Hara [2011a, 2012a] argue that in volume-time space, PIN can be approximated as

\[
PIN = \frac{e\epsilon u}{\alpha \mu + 2 \epsilon} \approx VPIN = \frac{E[|V^B - V^S|]}{V} = E[O_L]
\]  

(2)

where we have grouped trades in equal volume buckets of size \( V^B + V^S = V = \alpha \mu + 2 \epsilon \), and \( O_L = \frac{y^B y^S}{V} \) represents the order imbalance within \( V \). This volume-time approximation of PIN, known as VPIN, has been found to be useful in a number of settings (see Easley, Lopez de Prado and O’Hara [2012a] or Bethel et al. [2011] for example). In the next section we will show how VPIN’s expectations play a role in modeling the liquidity component of an execution strategy.

For a procedure that can be used to estimate OI, see Easley, Lopez de Prado and O’Hara [2012b].
3. THE LIQUIDITY COMPONENT

For the reasons argued in the introduction, traders are mindful of the footprint that their actions leave on the trading range, $\sum$. According to Eq. (2) the trading range will change due to adjustments on VPIN expectations. In the next two subsections we discuss the impact that our trade has on the order imbalance, how that may affect VPIN expectations and consequently the transaction cost’s liquidity component.

3.1. IMPACTING THE ORDER IMBALANCE

Suppose that we are an aggressive trader for $m$ contracts to be traded in the next volume bucket.\(^9\) For now, we set the bucket size at an exogenously given $V$ (we explain in Section 5 how to optimize this bucket size). Our trade of $m$ contracts will cause the bucket to fill faster than it would otherwise fill. Let us suppose that our trade pushes buys and sells from other traders out of the next bucket at equal rates. Given our knowledge of $\mathbf{0I}$, we let $\overline{\mathbf{V}}^B$ and $\overline{\mathbf{V}}^S$ be our forecasts of buy and sell volume that would occur in the next bucket without our trade.\(^10\) The expected order imbalance over $V$ using our private information about our trade of $m$ contracts can be represented by

$$\overline{\mathbf{OI}} \equiv \frac{\overline{\mathbf{V}}^B - \overline{\mathbf{V}}^S}{V} = \frac{\overline{\mathbf{V}}^B}{V} (V - |m|) - \frac{\overline{\mathbf{V}}^S}{V} (V - |m|) + m = \frac{\overline{\mathbf{V}}^B - \overline{\mathbf{V}}^S}{V} \left(1 - \frac{|m|}{V}\right) + m$$

(3)

where $V \geq m$, $v^B = \frac{\overline{\mathbf{V}}^B}{V}$ represents the forecasted fraction of buy volume in absence of our trade and $\mathbf{0I} \equiv 2v^B - 1$ is the order imbalance in absence of our trade.\(^11\) In the nomenclature above, $\overline{\mathbf{OI}}$ incorporates private information and $\mathbf{0I}$ only public information. If $\frac{|m|}{V} \approx 0$, $\overline{\mathbf{OI}} \approx \mathbf{0I}$. Alternatively, in the extreme case of $\frac{|m|}{V} \approx 1$, we have $|\overline{\mathbf{OI}}| \approx 1$. Generally, the impact of our trade on $\overline{\mathbf{OI}}$ will depend on $m$, but also the $\mathbf{0I}$ that would otherwise occur in the next volume bucket. If, for instance, $V^B < V^S$, a trade of size $m = \frac{(\overline{\mathbf{V}}^S - V^S)v}{v^B - (\overline{\mathbf{V}}^S) + v}$ will make $\overline{\mathbf{OI}} = 0$.

\(^9\) Of course, $m$ must be smaller than or equal to $V$ in order for us to do it within one volume bucket. The meaning of “aggressive trader” is that we determine the timing of the trade (as opposed to being a “passive trader”).

\(^10\) Easley et al. [2012a] present evidence that order imbalance shows persistence over (volume) time. Easley et al. [2012d] present a forecasting model for $v^B$.

\(^11\) For a procedure that can be used to estimate $v^B$, see Easley, Lopez de Prado and O’Hara [2012b].
3.2. INFORMATIONAL LEAKAGE ON THE LIQUIDITY COMPONENT

The previous section explained how trading $m$ contracts interspersed with $V - m$ external volume displaces the expected order imbalance from $0I$ to $\overline{0I}$. Next, we discuss how that displacement triggers an updating of the market makers’ expectations on the order imbalance, and thus VPIN.

Market makers adjust their estimates of VPIN as a result of the information leaked during the trading process. *Ceteris paribus*, if $\left\{m\right\}$ is relatively small, we expect it to convey little information to market makers. For example, if a small trade is executed in block but it is however large enough to fill a small bucket of size $V$, market makers may expect VPIN to remain around forecasted levels rather than jump to 1. We model the expected order imbalance (leaked to market makers) during execution, $\overline{0I}$, as a convex combination of two extreme outcomes: No leakage ($\varphi(|m|) \rightarrow 0$) and complete ($\varphi(|m|) \rightarrow 1$) informational leakage.

\[
\overline{0I} = \varphi(|m|) \left(2^\frac{m}{V} - 1\right) \left(1 - \frac{|m|}{V}\right) + (1 - \varphi(|m|)) \left(2^\frac{\ell}{V} - 1\right)
\]

where $\varphi(.)$ is a monotonic increasing function in $\left\{m\right\}$ with values in the range $(0,1)$. $\overline{0I}$ contains private information to the extent that it has been leaked by $m$. That occurs as a result of the trade’s size ($|m|$). The role of $\varphi(.)$ is to determine the degree by which the effective order imbalance during our execution $\overline{0I}$ impacts the market makers’ expectations on VPIN, and consequently leads to an adjustment of the range at which they provide liquidity, $\Sigma$.

It is critical to understand that the privately known order imbalance ($\overline{0I}$) may differ from the one inferred by the market makers ($\overline{I}$). The reason is, some private information may have been leaked by $m$, but not all. If $\varphi(|m|) \rightarrow 0$, there is no informational leakage, and $\overline{0I} \approx 0I$, regardless of $\overline{0I}$. In this case, it is as if $m$ had not been traded and the effect on VPIN will be imperceptible. However, when the order is so large that $\varphi(|m|) \rightarrow 1$, the leak is complete and the market maker knows as much as the trader, $\overline{0I} \approx \overline{I}$. For instance, an order of 75,000 contracts may dramatically displace the expectation of VPIN, even if blended among a volume 10 times greater (see SEC-CFTC [2010] in connection with the Waddell & Reed order). This displacement of the expectation from $0I$ to $\overline{0I}$ is the footprint left by $m$ in the liquidity provision process. Different footprint specifications could be considered, but that would not conceptually alter the analysis and conclusions presented here.

For simplicity, we have assumed that $E[|0I|] = \varphi|0\}$ because this expectation is based on past, public information. Similarly, $\varphi(|m||\overline{0I}| = \varphi(|m||\overline{0I}|$ because $\varphi(|m||\overline{0I}|$ is precisely the portion of $\overline{0I}$ that has become public due to $m$’s leakage. In this particular model we have not considered other sources of uncertainty in forming VPIN’s expectation, thus $\overline{VPIN} = E[|\overline{0I}| = |\overline{0I}|$. 

\[
\text{\textcopyright 2015/149035} \quad \text{PCT/US2015/023198}
\]
4. THE TIMING RISK COMPONENT
Minimizing the footprint that trading has on $\Sigma$ is an important factor when deciding the execution horizon. But it is not the only factor that matters to a trader. Minimal impact may require slicing a large order into multiple small trades, resulting in a long delay in executing the intended trade. This delay involves a risk, called *timing risk*. To model this timing risk, we introduce a simple, standard model of how the midpoint of the spread evolves. It is useful to view the trading range, $\Sigma$, as being centered on the mid-price, $S$, which moves stochastically as (volume) time passes. We model this process as an exogenous arithmetic random walk

$$\Delta S \equiv \hat{\sigma} \frac{V}{\sqrt{V_\sigma}} \xi$$

where $\xi \sim N(0, I)$ i.i.d. and $V_\sigma$ is the volume used to compute each mid-price change. So $V/V_\sigma$ is the number of price changes recorded in a bucket of size $V$. The standard deviation, $\sigma$, of price changes is unknown, but it can be estimated from a sample of $n$ equal-volume buckets as $\hat{\sigma}$, in which case $\frac{(n-1)\hat{\sigma}^2}{\sigma^2} \approx \chi^2_{n-1}$.

According to this specification, the market loss from a trade of size $m$ can be probabilistically bounded as

$$P \left[ Sgn(m)\Delta S > Z_\lambda \frac{V}{\sqrt{V_\sigma}} \right] = 1 - \lambda$$

where $Sgn(m)$ is the sign of the trade $m$, $\lambda$ is the probability with which we are willing to accept a loss greater than $Z_\lambda \frac{V}{\sqrt{V_\sigma}}$, and $Z_\lambda$ is the critical value from a Standard Normal distribution associated with $\lambda \in (0, \frac{1}{2}]$. $\lambda$ can also be interpreted as a risk aversion parameter, as it modulates the relative importance that we give to timing risk.

The specification above abstracts from any direct impact that $m$ may have on the midpoint price process. As is well known from microstructure research, the private information in our trade can introduce permanent effects on prices. In Appendix 2, we provide a generalization of this component taking into account the possibility that $m$ leaks part of our private information into the mid-price.

5. OPTIMAL EXECUTION HORIZON
We have argued that the impact of our trade of $m$ on $\Sigma$ will depend on the size of $m$ relative to $V$. In this section we take the intended trade, $m$, as fixed and determine the optimal trading volume $V$ in which to hide our trading action $m$ without incurring excessive timing risk. In order to compute this quantity, we define a probabilistic loss $\Pi$ which incorporates the liquidity and timing risk components discussed earlier. The probabilistic loss models how different execution horizons will impact our per unit portfolio valuation after the trade’s completion. This is not an
implementation cost, as that is contingent upon the execution model adopted to slice the parent order into child orders. OEH's goal is to determine the $V^*$ that optimally conceals (in the sense of minimizing $\Pi$) for a given risk aversion $\lambda$) our order $m$, given the prevalent market state ($\varphi[|m|], v^B, [5 - 5], \nu_0, \sigma$), where

$$\Pi = \frac{\varphi[|m|] \left[(2v^B - 1) \left(1 - \frac{|m|}{V}\right) + \frac{|m|}{V}\right] + (1 - \varphi[|m|])(2v^B - 1) |\bar{S} - \bar{S}|}{\sqrt{V_\sigma \delta}}$$

(7)

$$\text{liquidity component}$$

$$-Z_\lambda \sqrt{V_\sigma \delta}$$

(8)

$$\text{timing risk component}$$

with $Z_\lambda \leq 0$, thus the subtraction. The greater $V$, the smaller the impact on the order imbalance, but also the larger the possible change in the center of the trading range. Appendix 1 demonstrates the solution to minimizing $\Pi$, which can be implemented through the following algorithm (see Appendix 3 for an implementation in Python language):

1. If $(2v^B - 1)|m| < m$, try $V_1 = \left(2\varphi[|m|]/((2v^B - 1)|m| - m)/|S - \bar{S}|\right)^{2/3}$ and compute the value of $\bar{O}[V_1]$ associated with $V_1$, $\bar{O}[V_1]$.
   a. If $\bar{O}[V_1] > 0$ and $v^B \leq v^{B+}$, then $V^* = V_1$ is the solution.
   b. If $\bar{O}[V_1] > 0$ and $v^B > v^{B+}$, then $V^* = m$ is the solution.

2. If $(2v^B - 1)|m| > m$, try $V_2 = \left(2\varphi[|m|][m - (2v^B - 1)|m|]/|S - \bar{S}|\right)^{2/3}$ and compute the value of $\bar{O}[V_2]$ associated with $V_2$, $\bar{O}[V_2]$.
   a. If $W[V_2] < 0$ and $v^B \geq v^{B-}$, then $V^* = V_2$ is the solution.
   b. If $\bar{O}[V_2] < 0$ and $v^B < v^{B-}$, then $V^* = \bar{m} = m$ is the solution.

3. If $(2v^B - 1)|m| = m$, then $V_3 = |m|$ is the solution.

4. Else, try $V_4 = \varphi[|m|] \left(|m| - \frac{m}{2v^{B+} - 1}\right)$.
   a. If $v^B \geq v^{B+}$, then $V^* = V_4$ is the solution.
   b. If $v^B < v^{B+}$, then $V^* = \bar{m}$ is the solution.

where

$$\bar{O}[V^*] = \varphi[|m|] \left[\frac{m - (2v^B - 1)|m|}{V^*} + 2v^B - 1\right] + (1 - \varphi[|m|])(2v^B - 1)$$

(8)

$$v^{B+} = \frac{m}{2} + \frac{Z_\lambda \sqrt{|m|}}{4\varphi[|m|](|S - \bar{S}|)}$$

(9)
\[ \bar{\eta} I \in [-1,1] \] is the signed order imbalanced as a result of the informational leak that comes with trading \( m \). \( v^{B-} \), \( v^{B+} \) and \( v^{B=} \) set the boundaries for \( v^B \) in order to meet the constraint that \( V^* \geq \eta m \). The liquidity component is nonlinear in \( m \) and takes into account the side of the trade (leaning against or with the market), which we illustrate with numerical examples in Section 6.

Note that \( V^* \) has been optimized to minimize \( \Pi \) and it will generally differ from the \( V \) used to compute VPIN. We reiterate the point made earlier that various uses of VPIN require different calibration procedures. In this paper, we propose a method for determining the \( V^* \) that minimizes the probabilistic loss \( \Pi \), not the \( V \) that maximizes our forecasting power of \( v^B \) or short-term toxicity-induced volatility.

In practice, \( V^* \) could be estimated once before submitting the first child order and re-estimated again before submitting every new slice. By doing so, we incorporate the market's feedback into the model: If our initial slices have a greater (or smaller) impact than expected on the liquidity provision, we could adjust in real-time. That is, we assume that the values observed will remain constant through the end of the liquidation period, and determine the statically optimal strategy using those values. As the input parameters change, we could recompute the stationary solution. Almgren's [2009] shows that this "rolling horizon" approach, although not dynamically optimal, provides reasonably good solutions to a related problem.

6. NUMERICAL EXAMPLES

In this section, we develop numerical examples using the following state variables: \( \delta = 1,000, V_o = 10,000, m = 1,000, [\bar{S} - \underline{S}] = 10,000, \lambda = 0.05 \) and \( \phi[\eta m] \rightarrow 1 \). We want to find the optimal horizon to buy our desired amount of 1,000, recognizing that the optimal strategy must take account of both liquidity costs and timing risk costs. This strategy will depend, in part, on the imbalance in the market, which we capture as the fraction of buy volume, \( v^B \). Three scenarios appear relevant: \( v^B < \frac{1}{2}, v^B = \frac{1}{2} \) and \( v^B > \frac{1}{2} \). We limit our attention to \( m > 0 \), with the understanding that a symmetric outcome would be attained with a sell order \( \eta < 0 \) and \( v^B = 1 - v^B \) (see Figure 5).

[FIGURE 1 HERE]

Figure 1(a) displays the optimal volume horizons \( (V^*) \) for various values of the fraction of buy volume, \( v^B \). As is apparent, the optimal trading horizon differs dramatically with imbalance in the market. The reason for this is illustrated in Figure 1(b), which demonstrates how the probabilistic loss of trading 1,000 shares is also a function of the buy imbalance \( v^B \). The
probabilistic loss is the sum of the loss from the liquidity component and the loss from the timing component. Notice that with \( m > 0 \), the liquidity component does not contribute to \( \Pi \) until shortly before the market is balanced \( (\nu^B = \wedge) \).  

6.1. SCENARIO I: \( \nu^B = 0.4 \)

In this scenario we want to buy 1,000 contracts from a market that we believe to be slightly tilted towards sales (projected buys are only 40% of the total volume). If we plot the values of \( \Pi (V, \cdot) \) for different levels of \( V \), we obtain Figure 2.

[FIGURE 2 HERE]

In this case \( \partial \Pi (V^*) = 0 \) at the optimum trading horizon of \( V^* = 6,000 \) contracts (of which 5,000 come from other market participants). The optimum occurs at the inflexion point where the liquidity component function changes from convex decreasing to concave increasing. This may seem counter-intuitive as it is natural to expect the liquidity component to be decreasing in \( V \). But this reasoning misses the important role played by market imbalance. Because we are buying in a selling market, there is a \( V^* \) at which we are narrowing \( \Sigma \) to the minimum possible and our liquidity cost is zero. Once we pass that liquidity-component optimal \( V^* \), the trading range \( \Sigma \) necessarily widens again. The only way this can happen is with an increasing concave section in the liquidity component function, which explains the appearance of the inflexion point.

If we trade the desired 1,000 contracts within less than 6,000 total contracts traded our loss \( \Pi \) will increase because \( \Sigma \) increases more rapidly than the timing component of loss declines. If, on the other hand, we trade those 1,000 while more than 6,000 contracts are traded, our loss \( \Pi \) will increase because we will be taking on both excessive liquidity and timing costs.

One question to consider is, why does OEH allow the purchase to occur in the presence of selling flow which may result in lower future prices? Or put differently, why is not optimal to have a larger, possibly infinite execution horizon for a buy order, as long as \( \nu^B < 0.5 \)? After all, prices may go lower as a result of the selling pressure, and that would give the trader a chance to buy at a better level. The reason is, this is an execution model, not an investment strategy. The portfolio manager has decided that the trade must occur as soon as liquidity conditions allow it. It is not the prerogative of OEH to speculate on the appropriateness of the trader's decision, which may be motivated for a variety of reasons (she holds private information, it is part of her asset management mandate, she must obey a stop loss or abide by a risk limit, a release is about to occur, etc.). The role of the OEH model is merely to determine the execution horizon that minimizes the informational leakage.

6.2. SCENARIO II: \( \nu^B = 0.5 \)

In this scenario, the market is expected to be balanced (buys are 50% of the total volume). If we plot the values of \( \Pi (V, \cdot) \) for different levels of \( V \), we obtain Figure 3.

---

12 The liquidity component is positive at \( \nu^B < \frac{1}{2} \) as the order \( m=1,000 \) makes the market unbalanced toward buys for \( \nu^B < \frac{1}{2} \).
The optimum now occurs at $V^* = 11,392$ contracts, with a value for $\tilde{O}(V^*) = 0.088$. The model recognizes that a larger volume horizon is needed to place a buy order in an already balanced market than in a market leaning against our order (note the change in shape of the liquidity component). In this scenario, our order does not narrow $\Sigma$ abruptly (as in Scenario I), and the only way to reduce $\Pi$ is by substantially increasing $V$ (i.e., disguising our order among greater market volume). The liquidity component function is now convex decreasing, without an inflexion point, because the market is not leaning against us. But the optimal $V^*$ is still limited because of greater timing risk with increasing $V$.

6.3. SCENARIO III: $v^B = 0.6$

The market is now expected to be tilted toward buys, which represent 60% of the total volume. If we plot the values of $\Pi[V^*]$ for different levels of $V$, we obtain Figure 4.

The optimum now occurs at $V^* = 9,817$ contracts, with a value for $\tilde{O}(V^*) = 0.2816$. Two forces contribute to this outcome: On one hand, we are leaning with the market, which means that we are competing for liquidity ($v^B > v^E$), and we need a larger volume horizon than in Scenario I. On the other hand, the gains from narrowing $\Sigma$ are offset by the additional timing risk, and $\Pi$ eventually cannot be improved further. Note that this convex, decreasing liquidity component function asymptotically converges to the same level as the concave, increasing liquidity component function in Scenario I, due to the same absolute imbalance (VPIN) of both scenarios. The equilibrium between these two forces is reached at 9,817 contracts, a volume horizon between those obtained in Scenarios I and II.

7. EXECUTION HORIZONS AND THE OSCILLATORY NATURE OF PRICES

Figure 1 illustrates how traders behave as predicted toxicity increases. For buyers in a market tilted toward buys, the execution horizon is a decreasing function of $v^B$. For sellers in a market tilted toward sells, the execution horizon is also a decreasing function. Figure 5 depicts the optimal horizon for a sell order. Note that, in this case, $v^E$ decreases as we move from left to right in the graph. For both buyers and sellers, therefore, their optimal trading horizon will be influenced by the toxicity expected in the market.

We next consider the combined impact of alternative trade sizes, sides and $v^B$. For simplicity, we assume that $\varphi(|\eta|)$ is linear in $\sqrt{m}$, with $\varphi(|\eta|) = \frac{m}{10^3}$ for $\eta \in (0, 10^3)$. The largest execution horizons occur in a perfectly balanced market, due to the inherent difficulty of concealing trading intentions. Again we observe in Figure 6 that leaning against the market (selling in a buying market or buying in a selling market) allows for shorter execution horizons, thanks to the possibility of achieving zero liquidity cost. Leaning with the market (selling in a
selling market or buying in a buying market) leads to shorter execution horizons than those in a balanced market, but longer than when the trade leans against the market.

Now consider a recent event in which toxicity played a significant role. The CFTC-SEC examination of the "flash crash" (see CFTC-SEC [2010]) indicates that, during the crash, market participants dumped orders on the market. Why would anyone reduce their execution horizon in the midst of a liquidity crisis? Perhaps the most cited example is the Waddell & Reed order to sell 75,000 E-mini S&P500 contracts. It seems at first unreasonable to execute large orders in an increasingly illiquid market. The model introduced in this paper provides a possible explanation for this behavior, and illustrates how it contributes to the oscillatory nature of prices.  

Suppose that \( v_i \) increases to a level sufficient to have a measurable impact on PIN. This prompts sellers to shorten their execution horizons in volume-time (Scenario III, left side of Figure 5) because small gains from the liquidity component come at the expense of substantial timing risks. As several sellers compete for the same gradually scarcer liquidity, an increase in the volume traded per unit of time will likely occur, which is consistent with the observation that market imbalance accelerates the rate of trading (i.e., greater volume occurs during liquidity crises, like the "flash crash"). Prices are then pushed to lower levels, at which point buyers will have even shorter execution horizons than the sellers (Scenario I, left side of Figure 2). This is caused by the convex increasing section of the optimum for buyers (Figure 1(a)), compared to the concave decreasing section of the optimum for sellers (Figure 5). The increasing activity of buyers causes prices to recover some of the lost ground, \( v_i \) returns to normal levels, and execution horizons expand (Scenario II). The outcome is an oscillatory price behavior induced solely by timing reactions of traders to an initial market imbalance.

8. VOLUME PARTICIPATION STRATEGIES
Many money managers conceal large orders among market volume by targeting a certain participation rate. Part of OEH's contribution is to show that, to determine an optimal participation rate, order side and order imbalance should also be taken into account. In this section, we will illustrate how a typical volume participation scheme performs in relative terms to an optimal execution horizon strategy.

Suppose we apply the same parameter values used in Section 6: \( 8 = 1,000, V_o = 10,000, \lfloor 5 - \frac{5}{10^4} \rfloor = 10,000, \lambda = 0.05. \) For simplicity, consider a \( \Phi(V) \) linear in \( V \), with \( \Phi(V) = \frac{m}{10^4} \), for \( m \in (0, 10^4) \). Figure 7 compares the probabilistic loss from OEH and a scheme that participates in 5% of the volume, for various buy order sizes when \( v_B = \frac{1}{2} \) (i.e. when the markets is balanced between buying and selling). What is apparent is that volume participation results in

---

13 We are not claiming that the Waddell & Reed order was executed optimally. Our argument instead is that trading faster in toxic markets is not necessarily irrational.

14 Note that in this discussion we are taking the existence of buyers and sellers as given. The effects discussed in the text arise only from optimal trading horizons. If the impacts on prices also induce changes in desired trades, the effects on market conditions may be exacerbated or moderated.
a far mostly costly trading outcome. For a desired trade size of 2000 contracts, the Volume Participation probabilistic loss is almost 50% greater than that of the OEH algorithm.

[FIGURE 7 HERE]

The divergent behavior of these algorithms is also influenced by market imbalance. Figures 8 and 9 present the equivalent results for \( \nu^B = 0.4 \) and \( \nu^B = 0.6 \) respectively. OEH's outperformance is particularly noticeable in those cases when the order leans against the market (i.e. buying in a selling market).

[FIGURE 8 HERE]

[FIGURE 9 HERE]

9. THE SQUARE ROOT RULE

Loeb [1983] was the first to present empirical evidence that market impact is a square root function of the order size. Grinold and Kahn [1999] justified this observation through an inventory risk model: Given a proposed trade size \( \lambda n \), the estimated time before a sufficient number of trades appears in the market to clear out the liquidity supplier's net inventory is a linear function of \( \frac{|m|}{\bar{V}} \), where \( \bar{V} \) is the average daily volume. Because prices are assumed to follow an arithmetic random walk, the transaction cost incorporates an inventory risk term of the form

\[
\bar{\sigma} \frac{|m|}{\sqrt{V}}.
\]

Over the last three decades, studies have variably argued in favor of linear (Breen, Hodrick and Korajczyk [2002], Kissell and Glantz [2003]), square-root (Barra [1997]) or power-law (Lillo, Farmer and Mantegna [2003]) market impact functions. A possible explanation for these discrepancies is that these studies did not control for two critical variables affecting transaction costs: order side and its relation to order imbalance. Consider once again the same parameter values used in Section 6: \( \bar{\sigma} = 1,000, V_o = 10,000, [S - S_f] = 10,000, \lambda = 0.05. \) Suppose that \( \varphi(|m|) \) is linear in \( \lambda n \), with \( \varphi(|m|) = \frac{m}{10^4} \) for \( m \in (0, 10^4) \).

Figure 10 plots the probabilistic loss \( \Pi \) that results from executing a buy order of size \( m \) at optimal horizons \( V^*(m) \) given \( \nu^B = \frac{1}{2} \). A power function fits \( \Pi \) almost perfectly, with a power coefficient very close to the \( \frac{3}{4} \) reported by Almgren, Thum, Hauptmann and Li [2005]. However, Figure 11 shows that, if \( \nu^B = 0.4 \), \( \Pi \) has a linear end and is below the levels predicted by the square root. Finally, when \( \nu^B = 0.6 \), Figure 12 displays \( \Pi \) values greater than in the other cases, as the order now competes for liquidity. These conclusions depend on our assumption of a linear \( \varphi(|m|) \), but they can be generalized to other functional forms. For example, Figure 13 shows that, for \( \varphi(|m|) \propto \sqrt{m} \) and \( \nu^B = \frac{1}{2} \), \( \Pi \) fits the square root perfectly, as originally reported by Loeb [1983].

[FIGURE 10 HERE]
10. EMPIRICAL ANALYSIS

In the previous sections we presented a theory showing that OEH optimally uses order imbalance to determine the amount of volume needed to disguise a trade. We have also shown that OEH provides an explanation for the apparently contradictory views on the functional form of the market impact function. In this section we study the empirical performance of OEH relative to a VWAP strategy.

We consider an informed trader who wishes to trade $m$ units of a particular futures contract. To keep the discussion as general as possible, we will discuss two scenarios. In the first scenario, the trader has information about the sign of the price change over the next volume bucket, but not its magnitude. In this case, we model the desired trade by $m = S \text{gn} (\Delta P_{\tau})q$, where $q$ is a standard trade size determined by the trader. In the second scenario, the trader has information about the sign as well as the magnitude of the price move, and in this case we model the desired trade by $m = f (\Delta P_{\tau})q$. As in Section 5, we can analyze the performance of an execution strategy in terms of two components: Liquidity and timing costs. The timing cost can be directly observed, as measured by the price change with respect to the fill price, excluding the impact that the order has on liquidity. For example, in the first scenario the trader may know that prices will go up, but she may not know the magnitude of the price increase, so for a sufficiently large $m$ the expected timing gain may turn into a loss due to the liquidity cost.

When the trader executes using OEH, she combines her imperfect information with market estimates of $\sigma$ and $\nu^b$ over the next bucket. In this particular exercise, the latter is based on the procedure described in Appendix 4, and the former is the standard deviation of price changes over a given sample length. Then, for some $\phi [||m||]$, $\lambda$, and a long-run volatility $\tilde{\sigma}$, assuming $[S - \overline{S}] = -2\chi\sigma$,$^{15}$ she can compute the optimal execution horizon $V^*$ over which to trade $m$ contracts. Let’s denote the average fill price over the horizon $V^*$, in absence of her trade, by $\tilde{P}_{OEH}$. The realized timing component is $(p_{\tau} - \tilde{p}_{OEH,\tau})_{TM}$. The liquidity component can be estimated as $|\overline{\Delta I}_{0,\tau} | [5 - \overline{S}] |m|$, following Eqs. (4) and (7). Thus, the total profit during volume bucket $\tau$ is

$$\Pi_{OEH,\tau} = \frac{-|\overline{\Delta I}_{OEH,\tau} | \mid S - \overline{S} \mid m + (p_{\tau} - \tilde{p}_{OEH,\tau})_{TM}}{p_{OEH,\tau}}$$

(12)

$^{15}$ Other specifications could be considered to model the maximum trading range at which market makers are willing to provide liquidity. Here, we express that number as a function of the long term volatility, multiplied by a market makers’ risk aversion factor. That factor may not coincide with $Z_{\nu}$ however we see an advantage in keeping the model as parsimonious as possible, rather than introducing a new variable.
where we have expressed the profit in terms of its liquidity \( (PL_{L_{EH,T}}) \) and timing \( (PL_{T_{EH,T}}) \) components. Similarly, if the trader executes through a VWAP with fixed horizon, the total profit during volume bucket \( \tau \) is

\[
PL_{VWAP,T} = \frac{\left[ \tilde{S} - S \right]|m|}{PL_{VWAP,T}} + \frac{(P_\tau - \tilde{P}_{VWAP,T})m}{PL_{VWAP,T}}
\]  

(13)

Based on the previous equations, we can compute the relative outperformance of OEH over VWAP in terms of its information ratio,

\[
IR = \frac{E[PL_{OEH,T} - PL_{VWAP,T}]}{\sigma[PL_{OEH,T} - PL_{VWAP,T}]} \sqrt{n}
\]  

(14)

where \( \sqrt{n} \) is the annualization factor, and \( n \) the number of independent trades per year.

Our goal is to evaluate the performance of OEH relative to a VWAP benchmark, for a trader that needs to execute a large order on a daily basis \( (n=260) \). To compute this performance, we estimate \( \nu^B \) using the methodology discussed in Easley, Lopez de Prado and O’Hara [2012b], on one volume bucket per day, where each volume bucket is composed of 25 volume bars. \( \sigma \) is estimated for each bucket looking back enough trades to have a combined volume of 5 times the ADV (an average of one week of trading activity). We then compute the optimal execution horizons for trades of size \( q = \left\{ \frac{ADV}{100}, \frac{ADV}{20}, \frac{ADV}{10} \right\} \), with a risk aversion \( \lambda = 0.05 \) and an informational leakage function \( \varphi(|m|) = \min \left( \frac{|m|}{\sqrt{ADV}}, 0.999 \right) \), where ADV is the average daily volume.\(^\text{15}\)

[**TABLE 1 HERE**]

Table 1 summarizes the data used in our calculations. We have selected these products because they encompass a wide variety of asset classes and liquidity conditions. For example, E-Mini S&P500 Futures are traded in the Chicago Mercantile Exchange, it is an equity index product, our sample contains 476,676,009 transactions recorded between January 1st 2007 and July 26th 2012, rolls occur 12 days prior to expiration date, and the ADV for that period has been 1,964,844.89 contracts.

[**TABLES 2-4 HERE**]

Tables 2, 3 and 4 report the outperformance of OEH over VWAP for trading sizes equivalent to 1%, 5% and 10% of ADV respectively, when the trader has information about the sign of the price move over the next volume bucket (not the size of the move). To interpret these results, suppose that a trader has information that the price of the E-Mini S&P500 futures will increase

\(^\text{15}\) These are rather general values, and alternative ones could be adopted, depending on the user’s specific objective. For example, \( \lambda \) could have been calibrated in order to maximize OEH’s risk-adjusted performance over VWAP.
over the next volume bucket, and for that reason she wishes to acquire a position equivalent to 1% of E-Mini S&P500 futures' ADV, i.e. buy 19,648 contracts. We compute the liquidity and timing components for that trade using VWAP, and evaluate by how much OEH beats VWAP on that same trade in dollar terms. After repeating that calculation for each of the 1,450 volume buckets, we can estimate OEH's performance over VWAP. For instance, Table 2 reports that the maximum profit attainable with this information is 12.1428 index points on average, if execution were instantaneous and costless. Out of that, OEH was able to capture 10.5104 index points on average, or 86.56% of the average maximum profit. This represents an outperformance of 4.3262 index points over VWAP's performance on the same trade, or 35.63% of the average maximum profit, with an information ratio of 10.04. As expected, OEH's IR edge over VWAP decays as we approach very large daily trades. For example, the information ratio associated with an OEH trade for 196,484 E-Mini S&P500 futures contracts (10% of its ADV) is 2.59. This occurs because for trades of that extremely large size, it becomes increasingly difficult to conceal the trader's intentions. Although the information ratio is smaller, the average dollar amount of savings is greater (0.9577 points per contract), because the trade is for a size 10 times larger.

[TABLES 5-7 HERE]

Tables 5, 6 and 7 present the results of applying our methodology when the trader has information about the side and size of the price change, for an amount equivalent to 1%, 5% and 10% of ADV respectively. OEH's edge over VWAP also decays as we approach very large daily trades, however the extent of this performance decay is much less pronounced when the trader holds size as well as sign information with regards to price changes.

11. CONCLUSIONS
The choice of execution horizon is a critical input required by many optimal execution strategies. These strategies attempt to minimize the trading costs associated with a particular order. They do not typically address the footprint that those actions leave in the liquidity provision process. In particular, most execution models do not incorporate information regarding the order's side, without which it is not possible to understand the asymmetric impact that the order will have on the liquidity provision process.

In this paper, we introduce the OEH model, which builds on asymmetric information market microstructure theory to determine the optimal execution horizon. OEH allows existing optimal execution models to minimize both an order's trading costs and its footprint in the market. In a high frequency world, this latter ability takes on increased importance. OEH is shown to perform better than schemes that target a participation rate. Our model also provides an explanation of the apparent disagreement in the literature regarding the functional form of the market impact function as a result of not controlling for the order side and its relation to the order imbalance. Overall, our analysis suggests a new way to trade in the high frequency environment that characterizes current market structure.

Our empirical study shows that EOH allows traders to achieve greater profits on their information, as compared to VWAP. If the trader's information is right, OEH will allow her to

17 The value of an index point is USD50 in the case of the E-Mini S&P500 futures.
capture greater profits on that trade. If her information is inaccurate, OEH will deliver smaller losses than VWAP. OEH is not an investment strategy on its own, but delivers substantial "execution alpha" by boosting the performance of "investment alpha".
APPENDIX

A.1. COMPUTATION OF THE OPTIMAL EXECUTION HORIZON

We need to solve the optimization problem

\[ \min_{V} \ n[V, v, b, m, [S - \bar{S}], \lambda, \sigma, \nu] \]

\[ \equiv \left| \varphi[|m|] \left(2v^b - 1\right) \left(1 - \frac{|m|}{V}\right) + \frac{m}{V} \right| \]

\[ + (1 - \varphi[|m|])(2v^b - 1) \left|\bar{S} - \bar{S}\right| - Z_\lambda \sqrt{\frac{V}{V_\sigma}} \sigma \]

subject to \( V \geq \nu m \).

Let’s denote

\[ \tilde{\lambda} \equiv \varphi[|m|] \left(2v^b - 1\right) \left(1 - \frac{|m|}{V}\right) + \frac{m}{V} + (1 - \varphi[|m|])(2v^b - 1) \]

\[ = \varphi[|m|] \left[\frac{m - (2v^b - 1)|m|}{V} + 2v^b - 1\right] + c \]

Observe that, if \( V \geq \nu m \), then \(-1 \leq \tilde{\lambda} \leq 1\). The objective function contains the absolute value of \( \tilde{\lambda} \) which depends on the choice variable \( V \) and it is not continuously differentiable around \( \tilde{\lambda} = 0 \). Accordingly, we solve the problem in cases based on the sign of \( \tilde{\lambda} \) at an optimum.

A.1.1. CASE 1: Suppose that \( \tilde{\lambda} > 0 \). Then

\[ \frac{\partial \Omega}{\partial V} = <p/m \sqrt{V^2} \frac{2v^b - l|m\nu}{V^2}, \text{ and} \]

\[ \frac{\partial \Pi(V)}{\partial V} = \varphi[|m|] \left(2v^b - 1\right) \left|\frac{|m| - m}{V^2}\right| \left|\bar{S} - \bar{S}\right| - \frac{Z_\lambda \sigma}{2\sqrt{V_\sigma}} \]

Note that the second term in (18) is positive, and so an interior solution can occur only if the first term in (18) is negative. We first suppose that the solution to the problem does not occur at the constraint \( V \geq \nu m \).

Using the first order necessary condition for an interior solution that \( \frac{\partial \Pi(V)}{\partial V} = Q_{\text{int}} \) multiplying by \( \sqrt{V} \), we obtain

\[ \varphi[|m|] \left(2v^b - 1\right) \left|\frac{|m| - m}{V}\right| \left|\bar{S} - \bar{S}\right| V^{-3/2} = \frac{Z_\lambda \sigma}{2\sqrt{V_\sigma}} \]

18
Thus,

$$V^* = \left(2\varphi \sqrt{|m|} \sqrt{2v^B - V} \sqrt{m} \sqrt{m} \sqrt{S - S} \frac{\sqrt{V_\sigma}}{Z_\lambda \sigma} \right)^{2/3}. \quad (20)$$

The $V^*$ given by Eq. (17) is a candidate for an interior solution if $\bar{\partial} I > 0$ evaluated at $V^*$ and $V^* \geq \sqrt{m}$. It is straightforward to show that $V^* \geq \sqrt{m}$ if $\nu^B < \nu^{B+}$ where

$$\nu^{B+} = \frac{Z_\lambda \sigma \sqrt{|m|}}{4\varphi \sqrt{[m]|S - S| \sqrt{V_\sigma}}} + \frac{|m|}{m} + 1. \quad (21)$$

Alternatively, if $\nu^B > \nu^{B+}$ then the candidate for a solution in this region is $\sqrt{m}$. Note that the condition $\nu^B < \nu^{B+}$ implies that $m$ is negative which in turn implies that the first term in (18) is positive and so the $V^*$ given by Eq. (17) is well defined.

In this region the second derivative of $\Pi(V)$ is

$$\frac{\partial^2 \Pi(V)}{\partial V^2} = \frac{Z_\lambda \sigma \sqrt{|m|}}{4\varphi \sqrt{[m]|S - S| \sqrt{V_\sigma}}} \frac{|m|}{m} + 1 + \frac{|m|}{m} + 1. \quad (22)$$

It can be shown that $\frac{\partial^2 \Pi(V)}{\partial V^2} > 0$ if $\nu^B < \nu^{B+}$.

A.1.2. CASE 2: Suppose that $\bar{\partial} I < 0$. Then

$$\frac{\partial \bar{\partial} I}{\partial V} = \varphi \sqrt{|m|} \frac{m - (2\nu^B - 1)|m|}{\nu^2}. \quad (23)$$

Using an argument similar to that in Case 1, we see that in this region the candidate for an interior solution is

$$V^* = \left(2\varphi \sqrt{|m|} \sqrt{m - (2\nu^B - 1)|m|} \sqrt{S - S} \frac{\sqrt{V_\sigma}}{Z_\lambda \sigma} \right)^{2/3}. \quad (24)$$

The $V^*$ given by Eq. (21) is a candidate for a solution if $\bar{\partial} I < 0$ evaluated at $V^*$ and $V^* \geq \sqrt{m}$. Note that $V^* \geq \sqrt{m}$ if $\nu^B \geq \nu^{B-}$ where

$$\nu^{B-} = \frac{|m|}{m} + 1 = \frac{Z_\lambda \sigma \sqrt{|m|}}{4\varphi \sqrt{[m]|S - S| \sqrt{V_\sigma}}}. \quad (25)$$

Note that the condition $\nu^B \geq \nu^{B-}$ implies that $m$ is negative which in turn implies that the $V^*$ given by Eq. (24) is well defined.
Alternatively, if \( v^B < v^B^- \) then the candidate for a solution in this region is \( m \).

In this region the second derivative of \( \Pi(V) \) is

\[
\frac{\partial^2 \Pi(V)}{\partial V^2} = -2\varphi|m||m - (2v^B - 1)m|V^3 [S - S] + \frac{Z_{\lambda}\sigma}{4V\sqrt{V_{\sigma}V}},
\]

(26)

It can be shown that \( \frac{\partial^2 \Pi(V)}{\partial V^2} > 0 \) if \( v^B \geq v^B^- \).

A.1.3. CASE 3: Suppose that \( \tilde{\alpha} = 0 \). In this case \( \frac{\partial \Pi(V)\,v}{\partial v} \) does not exist, and the optimum cannot be computed using calculus. However, we can still compute \( V^* \), because in this case

\[
\frac{\partial \Pi(V)}{\partial V} \left[ \frac{m - (2v^B - 1)m}{V} + (2v^B - 1) \right] + (1 - \varphi|m|)(2v^B - 1) = 0,
\]

(27)

and thus

\[
V^* = \varphi|m| \left( |m| - \frac{m}{2v^B - 1} \right).
\]

(28)

It is straightforward to show that \( V^* \geq m \) if \( v^B \geq v^B^- \) where

\[
v^B^- = \frac{1}{2} \left( \frac{m}{m \varphi|m| + 1} \right).
\]

(29)

A.2. INFORMATIONAL LEAKAGE ON THE MID-PRICE

The literature devoted to optimal execution typically differentiates between the temporary and the permanent impact that an order \( m \) has on mid-prices (e.g., Stoll [1989]). The temporary impact arises from the actual order slicing (also called trajectory) within the execution horizon, while the permanent impact is a result of the information leaked by the order flow. Because our model is not concerned with the trajectory computation, we do not take into consideration the temporary impact of the actual order slicing. With regard to the permanent impact, Almgren and Chriss [2000] and Almgren [2003] propose a linear permanent impact function that depends on \( m \).

Let us suppose that, as in the above referred models, the permanent impact function depends on \( m \), but that unlike those models it is also a function of our volume participation rate, \( \frac{|m|}{\nu} \), and whether we are leaning with or against the market. As we have noted, this is not the approach chosen by most execution strategies, but we believe it is worth considering because for large enough trades our orders will have a noticeable impact. For example, in the extreme case that
\[ \frac{|m|}{V} \approx 1, \text{ and a sufficiently large } |m| \text{, our parent trade will not go unnoticed, no matter how well we slice it (especially if we are leaning with the market). We would expect leakage to be greatest when informed traders dominate market activity over a considerable horizon. Under this conjecture, we could conceive a permanent impact function that indeed depends on factors other than } |m|. \]

In fact, the specification presented earlier in the paper already modeled an informational leakage process. In the text, we only considered the effect on the trading range, but there is no reason why we could not assume that the same way our parent trade leaves a footprint in the liquidity component, it also leaves a footprint in the mid-price. The probabilistic loss function could then take the form

\[
\Pi \equiv \varphi(|m|) \left[ (2v^B - 1) \left( 1 - \frac{|m|}{V} \right) + \frac{m}{V} \right] + (1 - \varphi(|m|))(2v^B - 1) \left( \left\lfloor \frac{S}{S} \right\rfloor \right)
\]

\[
+ |m|k - Z_\delta \frac{\sqrt{V}}{\sqrt{V}}
\]

for \( k \geq 0 \). If \( k = 0 \), the probabilistic loss reduces to our original specification. If \( k > 0 \), there is a permanent impact on prices that is linear in the informational leakage and the parent order's size. In this way we are taking into account not only the size of the order, but also its side, as we should expect a greater permanent impact when we compete with the market for scarce liquidity. The optimal execution horizon can then be computed with the following algorithm:

1. If \( (2v^B - 1)|m| < m \), try \( V_1 = \left( 2\varphi(|m|) \left[ (2v^B - 1)|m| - m \right] \left\lfloor \frac{S}{S} \right\rfloor + |m|k \right) \left( \frac{\sqrt{V}}{Z_\delta \sigma} \right)^{2/3} \)
    and compute the value of \( \overline{O}I \) associated with \( V_1, \overline{O}I[V_1] \).
    a. If \( OI[V_1] > 0 \) and \( v^B \leq v^{B+} \), then \( V^* = V_1 \) is the solution.
    b. If \( OI[V_1] > 0 \) and \( v^B > v^{B+} \), then \( V^* = |m| \) is the solution.

2. If \( (2v^B - 1)|m| > m \), try \( V_2 = \left( 2\varphi(|m|) \left[ m - (2v^B - 1)|m| \right] \left\lfloor S \right\rfloor + |m|k \right) \left( \frac{\sqrt{V}}{Z_\delta \sigma} \right)^{2/3} \)
    and compute the value of \( \overline{O}I \) associated with \( V_2, \overline{O}I[V_2] \).
    a. If \( OI[V_2] < 0 \) and \( v^B \geq v^{B-} \), then \( V^* = V_2 \) is the solution.
    b. If \( OI[V_2] < 0 \) and \( v^B < v^{B-} \), then \( V^* = |m| \) is the solution.

3. If \( (2v^B - 1)|m| = m \), then \( V_3 = |m| \) is the solution.

4. Else, try \( V_4 = \varphi(|m|) \left( |m| - \frac{m}{2v^{B-} - 1} \right) \)
    a. If \( v^B \geq v^{B+} \), then \( V^* = V_4 \) is the solution.
    b. If \( v^B < v^{B+} \), then \( V^* = |m| \) is the solution.

with

\[
\overline{O}I[V^*] = \varphi(|m|) \left[ m - (2v^B - 1)|m| \right] + 2v^B - 1 \right] + (1 - \varphi(|m|))(2v^B - 1) \]

(31)
\begin{align}
\nu^{B+} &= \frac{|m|}{2} + 1 + \frac{Z_\alpha \sqrt{|m|}}{4 \varphi[|m|][S - \bar{S} + |m||k|]\sqrt{V_\sigma}} \\
\nu^{B-} &= \frac{|m|}{2} + 1 - \frac{Z_\alpha \sqrt{|m|}}{4 \varphi[|m|][S - \bar{S} + |m||k|]\sqrt{V_\sigma}} \\
\nu^{\beta} &= \frac{1}{2} \left( \frac{|m| \varphi[^{|m|}|}{m (\varphi[^{|m|}| -1) + 1} \right)
\end{align}

A.3. ALGORITHM IMPLEMENTATION

The following is an implementation in Python of the algorithm described in this paper. Set the values given in the "Parameters" section according to your particular problem. The $k$ parameter is optional. If $k$ is not provided (or given a value $k=0$), the procedure described in Appendix 1 is followed. Otherwise, the specification discussed in Appendix 2 is applied. If you run this code with the parameters indicated below, you should get $V^*=6,000$ (the solution in Section 6.1).

```python
#!/usr/bin/env python
# By MLdP on 20120316 <ml863@cornell.edu>
# It computes the Optimal Execution Horizon

#---------------------------------------------------------------
# PARAMETERS
sigma=1000
volSigma=10000
S,S=10000
zLambda=-1.644853627 #CDF(0.05) from the Std Normal dist
vB=0.5
phi=[]
m=1000
k=0

#---------------------------------------------------------------
def signum(int):
    if(int < 0):return -1
    elif(int > 0):return 1
    else:return 0

#---------------------------------------------------------------
def getOH(v,m,phi,vB,sigma,volSigma):
    return phi*(float((2*vB-1)*abs(m)/v+2*vB-1)+\
    (1-phi)*(2*vB-1)

#---------------------------------------------------------------
def getBounds(m,phi,vB,sigma,volSigma,S,S,zLambda,k=0):
    vB_l=float(signum(m)+1)/2-zLambda*sigma*abs(m)**0.5/\ 
    float(4*phi*(S+S+abs(m)*k)*volSigma**0.5)
```
A.4. EXPECTED ORDER IMBALANCE

Given a sample of \( L \) buckets, we fit a forecasting regression model of the form

\[
\ln(v_{t+1}^B) = \beta_0 + \beta_1 \ln(v_t^B) + \epsilon_t
\]  

(35)
and because $v_t^B \in [0,1]$ we must obtain that $|\beta_1| < 1$.

The expected value of the order imbalance at $\tau$ over $\tau + 1$ is

$$E_\tau[OI_{\tau+1}] = 2e^{E_\tau[Ln(v_{\tau+1}^B)]} - 1$$  \hspace{1cm} (36)

where

$$E_\tau[Ln(v_{\tau+1}^B)] = \hat{\beta}_0 + \hat{\beta}_1 Ln(v_{\tau}^B)$$  \hspace{1cm} (37)

In general we would expect roughly balanced markets, and so $v_t^B = \frac{1}{2} \Rightarrow E_\tau[v_{\tau+1}^B] = \frac{1}{2}$. This means that the regression should cross through the equilibrium point $(Ln(v_{\tau}^B), E_\tau[Ln(v_{\tau+1}^B)]) = \left( Ln\left(\frac{1}{2}\right), Ln\left(\frac{1}{2}\right) \right)$, rather than having an equilibrium is systemically disrupted by some arbitrary intercept value $\hat{\beta}_0$. We impose that condition as

$$Ln(v_{\tau+1}^B) - Ln\left(\frac{1}{2}\right) = \hat{\beta}_1^* \left[ Ln(v_{\tau}^B) - Ln\left(\frac{1}{2}\right) \right] + \epsilon_\tau$$  \hspace{1cm} (38)

and we use,

$$Ln(v_{\tau+1}^B) = \hat{\beta}_0^* + \hat{\beta}_1^* Ln(v_{\tau}^B) + \epsilon_\tau$$  \hspace{1cm} (39)

where $\hat{\beta}_1^* = \frac{\sum_{t=1}^{T-1} \left( Ln(v_{\tau+1}^B) Ln(v_{\tau}^B) + Ln\left(\frac{1}{2}\right) \right)^2 - Ln\left(\frac{1}{2}\right) \left[ Ln(v_{\tau+1}^B) + Ln(v_{\tau}^B) \right]}{\sum_{t=1}^{T-1} \left[ Ln(v_{\tau}^B) \right]^2 + Ln\left(\frac{1}{2}\right)^2 - 2Ln\left(\frac{1}{2}\right) Ln(v_{\tau}^B)}$ and $\hat{\beta}_0^* = Ln\left(\frac{1}{2}\right) \left( 1 - \hat{\beta}_1^* \right)$.

---

18 When fitting this regression, observations pairs $(v_{\tau+1}^B, v_{\tau}^B)$ where either $v_{\tau+1}^B = 0$ or $v_{\tau}^B = 0$ are removed.
This Figure demonstrates how the optimal trading horizon for a buy order depends upon the expected fraction of buy orders in the market. When all orders are buys, \( v^B \) is 1, while if all orders are sells then \( v^B \) is 0. The optimal volume horizon is defined over shares or contracts. The figure is drawn for state variables: \( ^\wedge = 1,000, V_\sigma = 10,000, m = 1,000, [\bar{S} - \underline{S}] = 10,000, \lambda = 0.05 \) and \( \phi(|m|) = 1 \).
Figure 1(b) - Optimal probabilistic loss and its components for various $\nu^B$ on a buy order

This Figure shows the expected total trading loss for a buy order and its relation to market imbalance. The total loss is a function of the liquidity component (the cost from trading immediately) and a timing component (the cost from delaying trading). The expected market imbalance is given by the expected fraction of buy orders in the market. When all orders are buys, $\nu^B$ is 1, while if all orders are sells $\nu^B$ is 0. The optimal volume horizon is defined over shares or contracts. The figure is drawn for state variables: $S = 1,000$, $V_\sigma = 10,000$, $m = 1,000$, $|S - S'| = 10,000$, $\lambda = 0.05$ and $\varphi[|m|] = 1$. 
This Figure demonstrates the expected total trading loss for a buy order and its relation to the volume horizon when the market is expected to have more selling activity. The total loss is a function of the liquidity component (the cost from trading immediately) and a timing component (the cost from delaying trading). The optimal volume horizon is defined over shares or contracts. The figure is drawn for state variables: $\delta = 1,000$, $V_g = 10,000$, $m = 1,000$, $[S - S_0] = 10,000$, $\lambda = 0.05$ and $\varphi(|m|) = 1$. 

Figure 2 - $\Pi(V, \ldots)$ for different volume horizons ($V$), with $v^B = 0.4$
This Figure shows the expected total trading loss for a buy order and its relation to the volume horizon when the market is expected to be in a balanced state. The total loss is a function of the liquidity component (the cost from trading immediately) and a timing component (the cost from delaying trading). The optimal volume horizon is defined over shares or contracts. The figure is drawn for state variables: \( \delta = 1,000, \quad V_\sigma = 10,000, \quad m = 1,000, \quad \lceil 5 - 5 \rceil = 10,000, \quad \lambda = 0.05 \) and \( \varphi[|\eta|] = 1 \).
This Figure demonstrates the expected total trading loss for a buy order and its relation to the volume horizon when the market is expected to have a greater buy imbalance. The total loss is a function of the liquidity component (the cost from trading immediately) and a timing component (the cost from delaying trading). The optimal volume horizon is defined over shares or contracts. The figure is drawn for state variables: $\delta = 1,000$, $V_g = 10,000$, $m = 1,000$, $[S - S] = 10,000$, $\lambda = 0.05$ and $\varphi[|m|] = 1$. 

*Figure 4 - $\Pi(V,\cdot)$ for different volume horizons (V), with $V^B = 0.6$*
Figure 5 - Optimal $V$ for various $v^B$ on a sell order

This Figure shows how the optimal trading horizon for a sell order depends upon the expected order imbalance in the market. When all orders are buys (sells), $v^B$ is 1, while if all orders are sells $v^B$ is 0. The optimal volume horizon is defined over shares or contracts. The figure is drawn for state variables: $\hat{\sigma} = 1,000$, $\nu_\sigma = 10,000$, $m = 1,000$, $[S - \bar{S}] = 10,000$, $\lambda = 0.05$ and $\varphi[|\eta|] = 1$. 
Combining alternative trade sizes and sides with our three scenarios \((v^B = 0.4, \gamma^B = 1/2, \nu^B = 0.6)\) results in the optimal execution horizons displayed in the figure above. \(\delta = 1,000, V_g = 10,000, m = 1,000, [5 - S] = 10,000, \lambda = 0.05\) and \(\langle p/m \rangle\) linear.
This Figure shows the expected total trading loss for a buy order arising from either a Volume participation strategy or the OEH strategy when the market is expected to be in a balanced state. The total loss is a function of the liquidity component (the cost from trading immediately) and a timing component (the cost from delaying trading). The optimal volume horizon is defined over shares or contracts. The volume participation strategy is assumed to participate in 5% of the volume. The figure is drawn for state variables: $\sigma = 1,000$, $V = 10,000$, $[S - S] = 10,000$, $\lambda = 0.05$ and $\varphi(|\eta|)$ linear.
This Figure shows the expected total trading loss for a buy order arising from either a Volume participation strategy or the OEH strategy when the market is expected to have a greater sell imbalance. The total loss is a function of the liquidity component (the cost from trading immediately) and a timing component (the cost from delaying trading). The optimal volume horizon is defined over shares or contracts. The volume participation strategy is assumed to participate in 5% of the volume. The figure is drawn for state variables: $8 = 1,000$, $V = 10,000$, $[S - S] = 10,000$, $\lambda = 0.05$ and $\langle p/\langle m \rangle \rangle$ linear.
This Figure shows the expected total trading loss for a buy order arising from either a Volume participation strategy or the OEH strategy when the market is expected to have a greater buy imbalance. The total loss is a function of the liquidity component (the cost from trading immediately) and a timing component (the cost from delaying trading). The optimal volume horizon is defined over shares or contracts. The volume participation strategy is assumed to participate in 5% of the volume. The figure is drawn for state variables: $\delta = 1,000$, $V = 10,000$, $[S - S] = 10,000$, $\lambda = 0.05$ and $p/m$ linear.
When order flow is balanced, the probabilistic loss follows a functional form close to the square root.

Figure 10 - Probabilistic loss under $v^B = \frac{1}{2}$ and $\phi \{\pi|v|\}$ linear

$y = 14.329x^{0.6476}$
$R^2 = 0.9989$
Figure 11 - Probabilistic loss under $\nu^B = 0.4$ and $\phi \{ \pi_1 \}$ linear

When order flow is leaning against the market, the probabilistic loss has a piecewise linear functional form.
When the order is large and competing for liquidity, the probabilistic loss is greater than predicted by the square root.
Figure 1S - Probabilistic loss under $v_B = \frac{1}{2}$ and $\phi(\|\pi\|) \text{oc} \sqrt{m}$

The probabilistic loss exactly fits the square root when $v_B = \frac{1}{2}$ and $\phi(\|\eta\|) \text{oc} V_m$. 

\[ y = 56.735x^{0.5} \]

\[ R^2 = 1 \]
<table>
<thead>
<tr>
<th>Futures Contract</th>
<th>Exchange</th>
<th>Group</th>
<th>Start</th>
<th>End</th>
<th>Roll</th>
<th>Records</th>
<th>ADV</th>
</tr>
</thead>
<tbody>
<tr>
<td>E-Mini S&amp;P500</td>
<td>CME</td>
<td>Equity</td>
<td>1/1/2007</td>
<td>7/26/2012</td>
<td>12</td>
<td>476,676,009</td>
<td>1,964,844.89</td>
</tr>
<tr>
<td>T-Note</td>
<td>CBOT</td>
<td>Rates</td>
<td>1/1/2007</td>
<td>7/26/2012</td>
<td>28</td>
<td>95,091,010</td>
<td>921,056.33</td>
</tr>
<tr>
<td>EUR/USD</td>
<td>CME</td>
<td>FX</td>
<td>1/1/2007</td>
<td>7/26/2012</td>
<td>10</td>
<td>188,197,121</td>
<td>233,201.17</td>
</tr>
<tr>
<td>WTI Crude Oil</td>
<td>NYMEX</td>
<td>Energy</td>
<td>1/1/2007</td>
<td>7/26/2012</td>
<td>19</td>
<td>164,619,912</td>
<td>194,902.36</td>
</tr>
<tr>
<td>Gold</td>
<td>COMEX</td>
<td>Metals</td>
<td>1/1/2007</td>
<td>7/26/2012</td>
<td>27</td>
<td>62,672,073</td>
<td>81,854.96</td>
</tr>
<tr>
<td>Corn</td>
<td>CBOT</td>
<td>Softs</td>
<td>1/1/2007</td>
<td>7/26/2012</td>
<td>20</td>
<td>41,833,299</td>
<td>73,860.53</td>
</tr>
<tr>
<td>Natural Gas</td>
<td>NYMEX</td>
<td>Energy</td>
<td>1/1/2007</td>
<td>7/26/2012</td>
<td>20</td>
<td>4,494,294</td>
<td>6,171.32</td>
</tr>
</tbody>
</table>

Table 1 - Description of the data series used in the numerical examples

<table>
<thead>
<tr>
<th>Futures Contract</th>
<th>Information</th>
<th>Trade Size</th>
<th>Max Profit</th>
<th>OEH Profit (Pts)</th>
<th>Outperf.(Pts)</th>
<th>Outperf. (%)</th>
<th>IR</th>
</tr>
</thead>
<tbody>
<tr>
<td>E-Mini S&amp;P500</td>
<td>Sign</td>
<td>0.01*ADV</td>
<td>12.1, 28</td>
<td>10.5104</td>
<td>4.3262</td>
<td>35.63%</td>
<td>10.04</td>
</tr>
<tr>
<td>T-Note</td>
<td>Sign</td>
<td>0.01*ADV</td>
<td>0.3966</td>
<td>0.3441</td>
<td>0.1322</td>
<td>33.33%</td>
<td>9.18</td>
</tr>
<tr>
<td>EUR/USD</td>
<td>Sign</td>
<td>0.01*ADV</td>
<td>0.0074</td>
<td>0.0064</td>
<td>0.0028</td>
<td>37.28%</td>
<td>10.62</td>
</tr>
<tr>
<td>WTI Crude Oil</td>
<td>Sign</td>
<td>0.01*ADV</td>
<td>1.3913</td>
<td>1.1949</td>
<td>0.4582</td>
<td>32.93%</td>
<td>10.02</td>
</tr>
<tr>
<td>Gold</td>
<td>Sign</td>
<td>0.01*ADV</td>
<td>9.4, 932</td>
<td>8.1780</td>
<td>3.2875</td>
<td>34.63%</td>
<td>9.68</td>
</tr>
<tr>
<td>Corn</td>
<td>Sign</td>
<td>0.01*ADV</td>
<td>8.4, 173</td>
<td>7.2806</td>
<td>3.1640</td>
<td>37.59%</td>
<td>9.67</td>
</tr>
<tr>
<td>Natural Gas</td>
<td>Sign</td>
<td>0.01*ADV</td>
<td>0.1098</td>
<td>0.0945</td>
<td>0.0409</td>
<td>37.26%</td>
<td>9.50</td>
</tr>
<tr>
<td>Lean Hogs</td>
<td>Sign</td>
<td>0.01*ADV</td>
<td>0.74, 51</td>
<td>0.6334</td>
<td>0.2813</td>
<td>35.07%</td>
<td>10.51</td>
</tr>
<tr>
<td>Cotton#2</td>
<td>Sign</td>
<td>0.01*ADV</td>
<td>1.3211</td>
<td>1.1358</td>
<td>0.4675</td>
<td>35.38%</td>
<td>7.66</td>
</tr>
</tbody>
</table>

Table 2 - EOH’s outperformance over VWAP for trades equivalent to 1% of ADV and information regarding the side of the price move over the next bucket

<table>
<thead>
<tr>
<th>Futures Contract</th>
<th>Information</th>
<th>Trade Size</th>
<th>Max Profit</th>
<th>OEH Profit (Pts)</th>
<th>Outperf.(Pts)</th>
<th>Outperf. (%)</th>
<th>IR</th>
</tr>
</thead>
<tbody>
<tr>
<td>E-Mini S&amp;P500</td>
<td>Sign</td>
<td>0.05*ADV</td>
<td>12.1428</td>
<td>8.2047</td>
<td>2.2770</td>
<td>18.75%</td>
<td>5.63</td>
</tr>
<tr>
<td>T-Note</td>
<td>Sign</td>
<td>0.05*ADV</td>
<td>0.3966</td>
<td>0.2682</td>
<td>0.0649</td>
<td>16.37%</td>
<td>4.91</td>
</tr>
<tr>
<td>EUR/USD</td>
<td>Sign</td>
<td>0.05*ADV</td>
<td>0.0074</td>
<td>0.0051</td>
<td>0.0015</td>
<td>20.78%</td>
<td>6.51</td>
</tr>
<tr>
<td>WTI Crude Oil</td>
<td>Sign</td>
<td>0.05*ADV</td>
<td>1.3913</td>
<td>0.9275</td>
<td>0.2217</td>
<td>15.94%</td>
<td>5.33</td>
</tr>
<tr>
<td>Gold</td>
<td>Sign</td>
<td>0.05*ADV</td>
<td>9.4, 932</td>
<td>6.3202</td>
<td>1.6253</td>
<td>17.12%</td>
<td>5.15</td>
</tr>
<tr>
<td>Corn</td>
<td>Sign</td>
<td>0.05*ADV</td>
<td>8.4, 173</td>
<td>5.6471</td>
<td>1.7222</td>
<td>20.46%</td>
<td>5.73</td>
</tr>
<tr>
<td>Natural Gas</td>
<td>Sign</td>
<td>0.05*ADV</td>
<td>0.1098</td>
<td>0.0731</td>
<td>0.0221</td>
<td>20.14%</td>
<td>5.68</td>
</tr>
<tr>
<td>Lean Hogs</td>
<td>Sign</td>
<td>0.05*ADV</td>
<td>0.74, 51</td>
<td>0.4820</td>
<td>0.1230</td>
<td>16.51%</td>
<td>5.32</td>
</tr>
<tr>
<td>Cotton#2</td>
<td>Sign</td>
<td>0.05*ADV</td>
<td>1.3211</td>
<td>0.8776</td>
<td>0.2372</td>
<td>17.96%</td>
<td>4.24</td>
</tr>
</tbody>
</table>

Table 3 - EOH’s outperformance over VWAP for trades equivalent to 5% of ADV and information regarding the side of the price move over the next bucket
<table>
<thead>
<tr>
<th>Futures Contract</th>
<th>Information</th>
<th>Trade Size</th>
<th>Max Profit</th>
<th>OEH Profit (Pts)</th>
<th>Outperf. (Pts)</th>
<th>Outperf. (%)</th>
<th>IR</th>
</tr>
</thead>
<tbody>
<tr>
<td>E-Mini S&amp;P500</td>
<td>Sign</td>
<td>0.1*ADV</td>
<td>12.1,28</td>
<td>6.3551</td>
<td>0.9577</td>
<td>7.89%</td>
<td>2.59</td>
</tr>
<tr>
<td>T-Note</td>
<td>Sign</td>
<td>0.1*ADV</td>
<td>0.3966</td>
<td>0.2065</td>
<td>0.0212</td>
<td>5.36%</td>
<td>1.82</td>
</tr>
<tr>
<td>EUR/USD</td>
<td>Sign</td>
<td>0.1*ADV</td>
<td>0.0074</td>
<td>0.0039</td>
<td>0.0007</td>
<td>9.18%</td>
<td>3.17</td>
</tr>
<tr>
<td>WTI Crude Oil</td>
<td>Sign</td>
<td>0.1*ADV</td>
<td>1.3913</td>
<td>0.7037</td>
<td>0.0600</td>
<td>4.31%</td>
<td>1.58</td>
</tr>
<tr>
<td>Gold</td>
<td>Sign</td>
<td>0.1*ADV</td>
<td>9.4932</td>
<td>4.77,6</td>
<td>0.5125</td>
<td>5.40%</td>
<td>1.82</td>
</tr>
<tr>
<td>Corn</td>
<td>Sign</td>
<td>0.1*ADV</td>
<td>8.173</td>
<td>4.2081</td>
<td>0.6715</td>
<td>7.98%</td>
<td>2.63</td>
</tr>
<tr>
<td>Natural Gas</td>
<td>Sign</td>
<td>0.1*ADV</td>
<td>0.1098</td>
<td>0.0599</td>
<td>0.0091</td>
<td>8.30%</td>
<td>2.57</td>
</tr>
<tr>
<td>Lean Hogs</td>
<td>Sign</td>
<td>0.1*ADV</td>
<td>0.7451</td>
<td>0.3580</td>
<td>0.0314</td>
<td>4.22%</td>
<td>1.46</td>
</tr>
<tr>
<td>Cotton#2</td>
<td>Sign</td>
<td>0.1*ADV</td>
<td>1.3211</td>
<td>0.6582</td>
<td>0.0869</td>
<td>6.58%</td>
<td>1.72</td>
</tr>
</tbody>
</table>

*Table 4 - EOH’s outperformance over VWAP for trades equivalent to 10% of ADV and information regarding the side and size of the price move over the next bucket*

<table>
<thead>
<tr>
<th>Futures Contract</th>
<th>Information</th>
<th>Trade Size</th>
<th>Max Profit</th>
<th>OEH Profit (Pts)</th>
<th>Outperf. (Pts)</th>
<th>Outperf. (%)</th>
<th>IR</th>
</tr>
</thead>
<tbody>
<tr>
<td>E-Mini S&amp;P500</td>
<td>Sign</td>
<td>0.01*ADV</td>
<td>15.8723</td>
<td>14.1671</td>
<td>6.4076</td>
<td>40.37%</td>
<td>8.52</td>
</tr>
<tr>
<td>T-Note</td>
<td>Sign</td>
<td>0.01*ADV</td>
<td>0.5291</td>
<td>0.4721</td>
<td>0.1959</td>
<td>37.03%</td>
<td>6.74</td>
</tr>
<tr>
<td>EUR/USD</td>
<td>Sign</td>
<td>0.01*ADV</td>
<td>0.0098</td>
<td>0.0087</td>
<td>0.0039</td>
<td>39.98%</td>
<td>8.74</td>
</tr>
<tr>
<td>WTI Crude Oil</td>
<td>Sign</td>
<td>0.01*ADV</td>
<td>1.8682</td>
<td>1.6672</td>
<td>0.6830</td>
<td>36.56%</td>
<td>8.39</td>
</tr>
<tr>
<td>Gold</td>
<td>Sign</td>
<td>0.01*ADV</td>
<td>12.5753</td>
<td>11.4060</td>
<td>4.7222</td>
<td>37.55%</td>
<td>6.96</td>
</tr>
<tr>
<td>Corn</td>
<td>Sign</td>
<td>0.01*ADV</td>
<td>12.3966</td>
<td>11.0999</td>
<td>5.1200</td>
<td>41.30%</td>
<td>5.77</td>
</tr>
<tr>
<td>Natural Gas</td>
<td>Sign</td>
<td>0.01*ADV</td>
<td>0.1380</td>
<td>0.1230</td>
<td>0.0566</td>
<td>40.98%</td>
<td>7.97</td>
</tr>
<tr>
<td>Lean Hogs</td>
<td>Sign</td>
<td>0.01*ADV</td>
<td>0.8442</td>
<td>0.7552</td>
<td>0.3348</td>
<td>39.66%</td>
<td>7.92</td>
</tr>
<tr>
<td>Cotton#2</td>
<td>Sign</td>
<td>0.01*ADV</td>
<td>1.7879</td>
<td>1.6020</td>
<td>0.7070</td>
<td>39.54%</td>
<td>6.52</td>
</tr>
</tbody>
</table>

*Table 5 - EOH’s outperformance over VWAP for trades equivalent to 1% of ADV and information regarding the side and size of the price move over the next bucket*

<table>
<thead>
<tr>
<th>Futures Contract</th>
<th>Information</th>
<th>Trade Size</th>
<th>Max Profit</th>
<th>OEH Profit (Pts)</th>
<th>Outperf. (Pts)</th>
<th>Outperf. (%)</th>
<th>IR</th>
</tr>
</thead>
<tbody>
<tr>
<td>E-Mini S&amp;P500</td>
<td>Sign</td>
<td>0.05*ADV</td>
<td>15.8723</td>
<td>11.4107</td>
<td>4.0384</td>
<td>25.44%</td>
<td>7.34</td>
</tr>
<tr>
<td>T-Note</td>
<td>Sign</td>
<td>0.05*ADV</td>
<td>0.5291</td>
<td>0.3744</td>
<td>0.1120</td>
<td>21.17%</td>
<td>6.30</td>
</tr>
<tr>
<td>EUR/USD</td>
<td>Sign</td>
<td>0.05*ADV</td>
<td>0.0098</td>
<td>0.0071</td>
<td>0.0025</td>
<td>25.40%</td>
<td>8.26</td>
</tr>
<tr>
<td>WTI Crude Oil</td>
<td>Sign</td>
<td>0.05*ADV</td>
<td>1.8682</td>
<td>1.3453</td>
<td>0.4091</td>
<td>21.90%</td>
<td>7.25</td>
</tr>
<tr>
<td>Gold</td>
<td>Sign</td>
<td>0.05*ADV</td>
<td>12.5753</td>
<td>9.3511</td>
<td>2.9878</td>
<td>23.84%</td>
<td>5.94</td>
</tr>
<tr>
<td>Corn</td>
<td>Sign</td>
<td>0.05*ADV</td>
<td>12.3966</td>
<td>8.6294</td>
<td>3.0115</td>
<td>24.29%</td>
<td>6.28</td>
</tr>
<tr>
<td>Natural Gas</td>
<td>Sign</td>
<td>0.05*ADV</td>
<td>0.1380</td>
<td>0.0988</td>
<td>0.0358</td>
<td>25.98%</td>
<td>6.91</td>
</tr>
<tr>
<td>Lean Hogs</td>
<td>Sign</td>
<td>0.05*ADV</td>
<td>0.8442</td>
<td>0.6141</td>
<td>0.2119</td>
<td>25.11%</td>
<td>7.50</td>
</tr>
<tr>
<td>Cotton#2</td>
<td>Sign</td>
<td>0.05*ADV</td>
<td>1.7879</td>
<td>1.2726</td>
<td>0.4329</td>
<td>24.21%</td>
<td>5.46</td>
</tr>
</tbody>
</table>

*Table 6 - EOH’s outperformance over VWAP for trades equivalent to 5% of ADV and information regarding the side and size of the price move over the next bucket*
Table 7 - EOH’s outperformance over VWAP for trades equivalent to 10% of ADV and information regarding the side and size of the price move over the next bucket.
REFERENCES

• CFTC-SEC (2010). "Findings Regarding the Market Events of May 6, 2010".
Optimal Risk Budgeting under a Finite Investment Horizon

Marcos Lopez de Prado* Ralph Vince^ Qiji Zhu†

December 24, 2013

First version: October, 2013

*Head of Quantitative Trading & Research at Hess Energy Trading Company, New York, NY 10036, and a Research Affiliate at Lawrence Berkeley National Laboratory, Berkeley, CA 94720, USA. Email: lopezdeprado@lbl.gov

†The president of LSP Partners, LLC, 35185 Holbrook Rd, Bentleyville, OH 44022, Email: ralph@ralphvince.com

‡Professor, Department of Mathematics, Western Michigan University, Kalamazoo, MI 49008, Fax: (269)387-4530, Email: zhu@wmich.edu
Optimal Risk Budgeting under a Finite Investment Horizon

Abstract

Growth Optimal Portfolio (GOP) theory determines the path of bet sizes that maximize long-term wealth. This multi-horizon goal makes it more appealing among practitioners than myopic approaches, like Markowitz's mean-variance or risk parity. The GOP literature typically considers risk-neutral investors with an infinite investment horizon. In this paper, we compute the optimal bet sizes in the more realistic setting of risk-averse investors with finite investment horizons. We find that, under this more realistic setting, the optimal bet sizes are considerably smaller than previously suggested by the GOP literature. We also develop quantitative methods for determining the risk-adjusted growth allocations (or risk budgeting) for a given finite investment horizon.

Keywords. Growth-optimal portfolio, risk management, Kelly Criterion, finite investment horizon, drawdown.

AMS Classifications. 91G10, 91G60, 91G70, 62C, 60E.
1 Introduction

Growth Optimal Portfolio (GOP) theory is influential in portfolio management [3, 4, 8, 9] along with Modern Portfolio theory [7]. The GOP advocates to allocate investment capital at the (asymptotic) expected growth-optimal allocation so as to maximize the long-term expected compounded return of the portfolio. The attractive property of such portfolios includes optimal asymptotic growth, hence the name.

Despite its theoretical advantage, the growth-optimal allocation is also known to be generally too risky in practice [5, 6, 9, 10, 15, 16] and various methods have been suggested to reduce the risk exposure of expected growth-optimal allocations. One unsatisfactory aspect of these methods is that they are heuristic and there is no theoretical justification on why and how to do it.

Recently, in [14], Vince and Zhu observed that the GOP neglected two important practical considerations. First, the GOP optimizes the expected optimal growth assuming an infinite investment horizon. In reality, investors only invest in a finite time horizon. Second, in the GOP theory, the focus is on the expected growth only. Risk is ignored. In practice, of course, risk is a critical factor for any investment decision. Incorporating these two practical considerations in analyzing the bet size of the game of blackjack, Vince and Zhu show in [14], analytically and experimentally that the optimal bet size suggested by Kelly's formula [3], a predecessor of the GOP theory, needs to be adjusted downward considerably.

We consider our findings in terms of points within a return manifold which we term leverage space, the surface of expected geometric return in N + 1 dimensions for N simultaneous events, at Q periods. Each of the N events is characterized by an axis in the N + 1 dimensional manifold and bound between 0, where nothing is risked, and 1, where the risk is as small as possible so as to assure total loss with the manifestation of worst-case. Further, whenever there exists a cumulative outcome or feedback mechanism which can range positive or negative from one discrete point to another., we reside at some loci on the return surface within the leverage space manifold whether we acknowledge this or not, and pay the consequences or reap the benefits of such loci therein.

GOP, in effect, states that one should select those coordinates in the leverage space at the (asymptotic, Q → ∞) peak. We note Kelly's formula [3] refers to this asymptotic peak. Using the model in [13] to represent expected geometric return as a function of Q, we find that for a single position with a positive expectation, expected growth is always maximized at a fraction risked of 1/0 for Q = 1. As Q increases, this expected growth-optimal fraction rapidly decreases toward its asymptotic value as specified in the Kelly Criterion (the shape of the return surface within the leverage space manifold is a function of Q), and thus we generally use this asymptotic value for all situations as a proxy for the actual expected growth-optimal fraction in all cases except for small values for Q. Given the unwieldy model for actual expected-growth in [13], we use as a proxy herein Kelly's formula for the average expected growth per play assuming infinite play, express this as expected growth
after $Q$ plays, and very accurately derive values for the curve over the domain $t \in [0, 1]$ at any given but not too-small values for $Q$ in our analysis. The point becomes moot in the context discussed here as the important loci discussed in this paper do not appear until $Q$ reaches satisfactory values to satisfy this requirement. Thus, the Kelly Criterion solution is always less aggressive than what actually is the expected growth-optimal fraction for any finite horizon.

Further, as demonstrated in [14] we scale the Kelly Criterion answer using worst-case, so as to comport to being an actual fraction to risk so that multiple, simultaneous propositions can be considered properly (relative to each other), short sales considered, etc., such that we are always using a fraction and thus bounding the axes of the leverage space manifold at 0 and 1 for all possible simultaneous propositions. This is discussed further in the beginning of Section 3.

The results in [14] are based on two simple observations: the graph of the total return function for playing blackjack hands a finite number of times is a bell shaped curve and the risk, as measured by the maximum drawdown, is approximately proportional to the bet size. As a result, the slope of the line connecting $(0, 0)$ and any point on the return curve is proportional to the return / risk ratio where the risk is measured by the maximum drawdown. Thus, the optimal bet size maximizing the return / risk ratio is the line emanating from $(0, 0)$ and tangent to the return curve depicted in the Fig. 1.

We can see that the corresponding bet size for this point is somewhat more conservative than the Kelly-optimal bet size corresponding to the peak of the return curve. Moreover, also noteworthy in Fig. 1, is the inflection point corresponding to the middle line.

The inflection point is significant in that it is the boundary of increasing or decreasing marginal return when the bet size increases. If one were to pursue increasing the return / risk ratio, then it does not make sense to stop before reaching the inflection point. However, increasing the bet size beyond the inflection point arguably is a matter of choice. Because between the inflection point and the return / risk maximum point, while increasing the bet size still improves the return / risk ratio, the marginal benefit of doing so diminishes.
Figure 2: Inflection point

Thus, the interval between the inflection point and the return/ risk maximizing point is a reasonable region for players with different risk aversion to choose their appropriate strategy. Indeed these observations are corroborated by Monte Carlo simulations in [14].

Both of these aforementioned points are a function of horizon, $Q$, and migrate towards the asymptotic peak as $Q \to \infty$.1

Although [14] focuses on bet size for playing blackjack, the idea and methods also apply to general capital allocation for investment problems involving multiple risky assets or investment strategies. In fact, inflection points have been discussed in [13] in a more general context. The main goal of this paper is to provide a practical guide on how to implement the idea in [14, 13] for investment problems involving multiple investment instruments. When there are multiple assets involved, the path from the optimal GOP allocation to the completely riskless position of all cash (bonds) is not unique. In fact, there are infinitely many possible paths to choose from. Although given a risk measure, one can argue it is reasonable to follow a path that on every level of the return manifold, the path should travel through the point that minimizes the risk measure. In reality finding such a path may turn out to be very costly. Therefore, other alternative choices of paths should also be allowed. The risk measure when involving multiple assets and investment strategies is also more complicated. For each investment strategy the same argument in [14] still applies and the size of the funds allocated to it is approximately proportional to the drawdown. However, the proportional constants for different investment strategies may well be different. Moreover, the total drawdown will also depend on how the drawdowns for different investment strategies are correlated adding more technical issues into the equation.

Once a return / risk path is determined, the return function along this path becomes a one variable function of the parameter that defines the path. One may attempt to use the method in [14] to determine the trade-off between return and risk. However, the idea in [14] as alluded to in the previous paragraph only works when the parameter is proportional

---

1Thus we see the answer provided by Kelly's formula [3], though never the actual expected growth-optimal point, but rather the asymptotic limit of such, is also the asymptotic limit to these other two critical points of risk-adjusted expected growth-optimality.
to the risk. This only happens for a few return / risk paths. In practical capital allocation problems the comparison of different return / risk paths is often necessary. We show that the inflection points on different return / risk paths are all on one manifold determined by Sylvester’s criterion of negative definiteness of a matrix involving the derivatives of the return functions and provides a reasonable approximation. Similarly, we also develop equations characterizing the manifolds of all return / risk optimal points.

The rest of the paper is arranged as follows: we first illustrate our results using a simple concrete example in the next section. Then we discuss the general model and growth-optimal allocation in Section 3. Section 4 discusses the return / risk paths and special cases in which the methods in [14] directly apply. In Sections 5 and 6 we develop methods of characterizing the manifolds of inflection points and return/ risk optimal points, respectively. We also discuss approximations where appropriate. We then discuss application examples in Section 7. In Section 8 we discuss the significance of the points where cumulative growth has a deleterious effect as well as ways to change the central location of the growth curve when we do not control the amount at risk, so as to indirectly control it. We conclude in Section 9 and discuss avenues for further research regarding this material.

2 An Example

Let us consider playing a game where two coins are tossed simultaneously. Coin 1 is a .50/.50 coin that pays 2:1, and Coin 2 is a .60/.40 coin that pays 1:1 (to be non-symmetrical, as the simplest case). We play $Q = 50$ times. The joint probability distribution is summarized in Table 1.

<table>
<thead>
<tr>
<th>Coin1</th>
<th>Coin2</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>1</td>
<td>0.3</td>
</tr>
<tr>
<td>2</td>
<td>-1</td>
<td>0.2</td>
</tr>
<tr>
<td>-1</td>
<td>1</td>
<td>0.3</td>
</tr>
<tr>
<td>-1</td>
<td>-1</td>
<td>0.2</td>
</tr>
</tbody>
</table>

Table 1. Two Coins

For $f_1, f_2$ being the amounts risked on Coin 1 and Coin 2, respectively, $f_i \in [0, 1], i = 1, 2,$ then the expected return function is

$$r(f_1, f_2) = \exp(50K(f_1, f_2)) - 1,$$

where

$$l(f_1, f_2) = [0.3 \ln(1 + 2f_1 + f_2) + 0.2 \ln(1 + 2f_1 - f_2) + 0.3 \ln(1 - f_1 + f_2) + 0.2 \ln(1 - f_1 - f_2)]$$

is the expected log return function. Fig. 3 plots $r$ as a function of $f_1$ and $f_2$. Note that each of the four summands in the definition of $l(f_1, f_2)$ is a composition of the strictly concave
Figure 3: Leverage Space - Two Coin Example - Altitude is Growth at Q=50

log function and an affine function and, therefore, strictly concave. Thus, \( l(\beta, f_2) \) itself is also a strictly concave function defined on the leverage space \( (\bar{f}, f_2) \in [0,1] \times [0,1] \). We know that strictly concave function have at most one maximum point. On the other hand, \( l(\bar{f}, f_2) \) approaches \(-\infty\) as \( (f_1, f_2) \) approaches the upper and right boundaries of the leverage space \( [0, 1] \times [0, 1] \). Since both games have a positive expected return, neither the left boundary of the leverage space where \( f_1 = 0 \) nor the lower boundary where \( f_2 = 0 \) will contain the maximum for the log return function \( l(f_1, f_2) \). Thus, \( Z(\bar{f}, f_2) \) must attain a unique maximum in the interior of the leverage space. As the composition of the monotonic increasing function \( \exp() \) and the log return functions \( Z(\bar{f}, f_2) \), \( r(f_1, f_2) \) and \( Z(\bar{f}, f_2) \) share the same maxima, it follows that \( r(\bar{f}, f_2) \) also has a unique maximum in the interior of the leverage space which is the Kelly-optimal bet sizes for these two games played simultaneously. We can determine this Kelly-optimal point by applying the first order necessary condition:

\[
\nabla r(f_1, f_2) = 50 \exp(50 l(f_1, f_2)) \nabla l(f_1, f_2) = 0
\]

and represents it by the vector \( \kappa = (0.243, 0.180) \). As discussed above \( \kappa \) is the unique solution to (2.1).

Now we consider the risk in terms of the drawdowns. In [14] we have already observed that for each individual coin tossing game the drawdown is approximately proportional to the bet size. In fact, assuming that the sequence of consecutive returns of \( l_1, l_2, \ldots, Z_m \) is what causes the maximum drawdown in a particular sequence of play in a one player game,
then the drawdown of betting a fixed percentage \( l \) in this sequence of play will be

\[
|\sum_{i=1}^{m} (i + f_{t_i}) - i| = |\sum_{i=1}^{m} l_i + f^2 \sum_{1 \leq i < j \leq m} l_i l_j + f^3 \sum_{1 \leq i < j < k \leq m} l_i l_j l_k + \ldots|.
\]

We take the absolute value here because maximum drawdown is usually defined as a positive percentage while \((1 + f_{t_1}) (1 + f_{t_2}) \cdots (1 + f_{t_m}) - 1\) calculates the percentage lost, signified as a negative number. Usually, \( l \) is far smaller than 1 and, therefore, the first term on the right hand side of (2.2) dominates. Clearly this term is proportional to \( l \) by a linear factor of \( c = \sum_{i} T_{i}$^T \cdot l_i \). For a concrete game, \( c \) can be estimated by simulation using \( c \approx \text{drawdown} / l \) for small \( l \).

**Example 2.1.** Tossing a biased coin as described previously with a payoff of 1:1 and a probability of winning of .60, for 50 tosses, and assuming the following particular realization of sequential outcomes

\[
\{1, -1, 1, -1, -1, -1, 1, -1\}
\]

\[
\{1, 1, 1, 1, -1, 1, -1, 1, 1, 1, -1, 1, 1, 1, -1\}
\]

\[
\{1, 1, 1, 1, 1, -1, 1, -1, 1, -1\}
\]

We can see that the maximum percentage drawdown is caused by the sequence inside the brackets \( \{ \ldots \} \). In this one player game, with a bet size \( l \), this maximum drawdown as a function of \( l \) can be calculated as

\[
dd(f) = |(1 - l)^4 (1 + f)(1 - f)^2 - 1| = l \cdot 5l + 9l^2 - 5l^3 - 5l^4 + 9l^5 - 5l^6 + f^7
\]

Since the practical bet size \( l \) is usually small (\( l < 0.243 \) in this example), we can use the first order term \( 5l = | -5l| \) as an approximation of the maximum drawdown \( dd(f) \), so that in this example \( c = 5 \). To find \( c \) approximately in this case we can use the above approximation with \( l = 0.01 \). The result is

\[
c \approx dd(0.01)/0.01 = 4.91,
\]

which is quite accurate.

For a single game as discussed in [14], the magnitude of this proportional constant is not important. As long as the drawdown is approximately proportional to the bet size, we can use the bet size as a proxy for the drawdown as the risk. Now that we are considering two different games together, the proportional constants for the two games...
are in general different. When trying to approximate the aggregated drawdown, it is now important to estimate the two different proportional constants. Simulating these two games for 2000 rounds of playing 50 simultaneous tosses ($Q = 50$) each, then calculating the mean, demonstrates that for the two individual games the drawdowns are proportional to $\epsilon_1$ and $c_2 f_2$ where $\epsilon_1 = 5.73$ and $c_2 = 6.52$, respectively.

Another complication in discussing the approximation of the aggregated drawdowns in this situation is that we need to consider the correlation of the two drawdowns. When the two drawdowns occur independently of each other (referring to overlap of their duration in time), the aggregated drawdown is roughly the largest between the two linear approximations, 

\[ \text{max}(ci_1, c2/2) \] 

The other extreme is the two drawdowns are completely dependent so that the aggregated drawdown is proportional to $c_1 f_1 + c_2 f_2$. In general, there are some correlations between the drawdowns of the two games and we get something in between.

For each level $t$ of the return $r$, a risk-averse investor may select, on the level curve of $v(h, f_2) = t$, the point $(h(t), f_2(t))$ that minimizes the risk as measured by the drawdown. As $t$ progresses from 0 to the maximum of the return, $(h(t), f_2(t))$ will trace out a curve in the leverage space from $(0, 0)$ to $\kappa$. Each different choice of the approximation of the drawdown corresponds to such a curve. If we draw all such curves, they then aggregate to provide us a region in leverage space which represent allocations achieving minimum risk under given return. Since it is impossible to trace infinitely many such curves, we focus on the two extreme cases alluded to above to derive the boundaries of this region.

Consider the completely dependent case first. Our problem becomes, for each $t \in [0, rK]$, 

\[
\begin{align*}
\text{minimize} & \quad c_1 f_1 + c_2 f_2 \\
\text{subject to} & \quad r(f_1, f_2) = t.
\end{align*}
\] 

Observing that changing $r(f_1, f_2) = t$ to $r(f_1, f_2) \geq t$ does not change the solution, we see that problem (2.3) is a convex optimization problem and, therefore, has a unique solution determined by the Lagrange multiplier rule:

\[ Vr = \lambda(\epsilon_1, c_2). \]

Taking the ratio of the two components and using the computation of $Vr$ in (2.1) we can see that the curve $(h(t), f_2(t))$ is determined by the equation

\[ c_2 \frac{dr}{df_1} = c_1 \frac{dr}{df_2}, \]

starting from $\kappa$ until it intercepts one of the coordinate axes. Afterwards, it follows the intercepted coordinate axis to $(0, 0)$. We will name this curve Path 1 for reference in the future.
Similarly, when the two drawdowns are completely independent, we need to solve the convex constrained minimization problem

\[
\begin{align*}
\text{minimize} & \quad \max(c_1 f_1, c_2 f_2) \\
\text{subject to} & \quad r(i, f_2) \geq t.
\end{align*}
\] (2-5)

Here the optimality condition is (see e.g. [2])

\[
V r = \lambda (0 \delta \theta, (1 - 0)\delta \theta), \lambda > 0, \theta \in [0, 1].
\]

In this optimality condition \(0 = 0\) corresponding to \(c_2 f_2 > \delta \mu \delta \lambda\) and leads to \(f_2 = 0.18\). Similarly, \(0 = 1\) corresponds to \(c_2 f_2 < \delta \mu \delta \lambda\) and leads to \(f_1 = 0.243\). It follows that before either \(i(t)\) reaches 0.243 or \(f_2(t)\) reaches 0.18 we should always have \(0 \in (0, 1)\) which corresponds to \(c_2 f_2 = c_1 f_1\). Thus, the path \((f_1(t), f_2(t))\) follows the line

\[
c_2 f_2 = c_1 f_1
\] (2.6)

until either \(i(t)\) reaches 0.243 or \(f_2(t)\) reaches 0.18. After that, it follows the straight line to \(k = (0.243, 0.18)\). We will name this curve Path 2. For \(\epsilon_1 = 5.73\) and \(c_2 = 6.12\), the two curves discussed above are illustrated in Fig. 4, where the curves corresponding to (2.4) and (2.6) are colored in blue and black, respectively.
Next, we consider the return / risk optimizing point $\zeta$ and the inflection point $\nu$ on these paths. We already know that
\[
\min(-\frac{\partial^2 r}{\partial f_1^2}, \text{det}(\text{Hessian}(r))) = 0
\]  
(2.7)
determines the curve at which inflection of the graph of $r$ occurs. To determine $\zeta$ for the completely dependent case, we need to maximize $r(i_1, f_2)((c_1 f_1 + c_2 f_2)$. The optimality condition is
\[
0 = \nabla r(c_1 f_1 + c_2 f_2) - r(c_1, c_2)
\]
\[
(c_1 f_1 + c_2 f_2)^2
\]
which is equivalent to
\[
\text{Vr}(ci_1 + c2) = r(c_1, c_2).
\]
Using $(f_1, f_2)$ to dot product both sides and canceling $(c_1 i_1 + c_2 f_2)$ obtaining
\[
\text{Vr} - (l_1, l_2) = r.
\]  
(2.8)
It turns out that this is true in general (the proof uses a similar calculation involving subdifferentials for convex functions and will be given in Section 6). Thus, the $\zeta$ point corresponding to any curve will have to be on the curve defined by equation (2.8). In particular, we can find the $\zeta$ points corresponding to the independent and completely dependent cases by looking at the interception of the corresponding curves as shown in Fig. 5, in which the curves corresponding to Equations (2.8) and (2.7) are represented in red and green, respectively, and the blue and black curves again represent (2.4) and (2.6), respectively. The corresponding significant points are summarized below.

<table>
<thead>
<tr>
<th></th>
<th>$\nu$</th>
<th>$\zeta$</th>
<th>$\kappa$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(0.1885,0.077)</td>
<td>(0.2175,0.133)</td>
<td>(0.243,0.18)</td>
</tr>
<tr>
<td>2</td>
<td>(0.148,0.1385)</td>
<td>(0.1935,0.18)</td>
<td>(0.243,0.18)</td>
</tr>
</tbody>
</table>

Table 2. Two coins example with $Q = 50$

3 The general model

Consider investing in $M$ investment systems represented by a random vector $X = (X_1, \ldots, X_M)$ with $N$ different outcomes \{6\,^1\!, \ldots, 6\,^N\} where $b^i = 6^\text{M}$. We consider investing in those investment systems for $Q$ holding periods and suppose that $\text{Prob}(X = b^i) = p_i$. We wish to determine how best to allocate funds into those investment systems. We re-scale so that the allocation can be represented in leverage space [12]. Let $\underline{w}_m = \min(b_m^1, \ldots, b_m^N)$. Define the scaled random vector $Y = (X_1/\underline{w}_1, \ldots, X_M/\underline{w}_M)$ and the scaled outcome $\underline{a}^\nu = (b_1^0/\underline{w}_1, \ldots, b_M^0/\underline{w}_M)$. Then $\text{Prob}(Y = \underline{a}^\nu) = p_i$. Now, each allocation can be
Figure 5: $\zeta$ and $v$

represented by a vector $l = (l_1, \ldots, l_M) \in [0, 1]^M$ where $l_m$ represents the fraction of the total number of shares one can invest in the $m$th investment vehicle such that the worst loss $w_m$ will result in the loss of all the investment capital.

The expected gain over $Q$ holding periods the following representation:

**Theorem 3.1.** (Representation of expected gain over $Q$ holding periods) If one invests in the investment system $X$ described above for $Q$ periods, then the expected gain as a function of $f$ is

$$G_X^Q(f) = \left[ \sum_{n=1}^{N} p_n \left(1 + f \cdot a_n\right)^{1/Q} \right]^Q. \quad (3.1)$$

To prove we need only use the multi-variable polynomial expansion formula of

$$(A_1 + A_2 + \ldots + A_N)^Q = \sum_{1 \leq k_1, k_2, \ldots, k_Q \leq N} A_{k_1} A_{k_2} \cdots A_{k_Q} \text{ where } A_n = p_n \left(1 + f \cdot a_n\right)^{1/Q}.$$  

We defer the detail to the Appendix and turn to the limit of the expected gain in (3.1) which is useful as an approximation.

**Theorem 3.2.** (Limiting holding period gains)

$$\lim_{Q \to +\infty} G_X^Q(f) = \prod_{n=1}^{N} (1 + f \cdot a_n)^{p_n}. \quad (3.2)$$
Proof. We calculate the limit of \( G_X^Q(f) \) first:
\[
\lim_{Q \to \infty} \ln G_X^Q(f) = \lim_{Q \to \infty} Q \ln \left[ \sum_{n=1}^{N} p_n \left( 1 + f \cdot a_n \right)^{1/Q} \right]
\]
\[
= \lim_{r \to 0} \sum_{n=1}^{N} \ln \left( \frac{r}{1 + f \cdot a_n} \right) \frac{p_n}{r}, \quad (r = l/Q)
\]
\[
= \sum_{n=1}^{N} \frac{p_n \ln \left( 1 + f \cdot a_n \right)}{1 + f \cdot a_n}, \quad \text{L'Hospital's rule}
\]
It follows that
\[
\lim_{Q \to +\infty} G_X^Q(f) = e^{\lim_{Q \to +\infty} \ln G_X^Q(f)} = \prod_{n=1}^{N} \left( 1 + f \cdot a_n \right)^{p_n}.
\]
Q.E.D.

We denote the limiting expected gain per period by
\[
G_X(f) := \Pi_{n=1}^{N} \left( 1 + f \cdot a_n \right)^{p_n}.
\]
Then, the total return \( \phi_X^Q(f)^2 - 1 \) can be approximated by \( \phi_X^Q(f)^2 - 1 \). We have seen in [14] that, for reasonably large \( Q \), this approximation is quite accurate and we will use this approximation in the analysis below. We note that using the log return function
\[
l_X(f) := \sum_{n=1}^{N} p_n \ln (1 + f \cdot a_n)
\]
we have the representation
\[
|G_X(f)|^Q = \exp(Q l_X(f)).
\]
Since the exponential function is monotone increase and the log return function is concave, \( rQ(f) \) attains a unique maximum which can be determined by solving the equation
\[
0 = Vr_{Q}(f) = Q \exp(Q l_X(f)) Vl_X(f),
\]
or equivalently
\[
Vl_X(f) = 0.
\]
We denote the solution \( \kappa \) and note that this is the asymptotic growth-optimal allocation and therefore is independent of \( Q \).
4 Return / risk paths

As illustrated in the example discussed in Section 2, we know that the allocation $\kappa$ is usually too risky. In practice, one needs to reduce the risk and take a middle ground between $f = 0$ and $f = \kappa$. It is clear from the example in Section 2 there are many possible paths connecting these two possible allocations providing various trade-offs between risk and return.

**Definition 4.1.** A mapping $f : [a, b] \to \mathbb{R}^{M}$ is called a return / risk path if it has the following properties

1. $f$ has piecewise continuous second order derivatives.
2. $f(a) = 0$ and $f(b) = \kappa$.
3. $\lambda \mapsto \mathbb{Q}(f(\lambda))$ is an increasing function on $[a, b]$.
4. There is a risk measure $\mathbb{Q}$ on leverage space such that $\lambda \mapsto \mathbb{Q}(f'(\lambda))$ is an increasing function on $[a, b]$.

It is easy to see that both Path 1 and Path 2 in the example in Section 2 are return / risk paths. Along such paths we can show that an inflection point always exits as long as $\mathbb{Q}$ is large enough.

**Theorem 4.2.** Let $f : [a, b] \to \mathbb{R}^{M}$ be a return / risk path. Then, when $\mathbb{Q}$ is sufficiently large, the function $\lambda \mapsto \mathbb{Q}(f'(\lambda))$ has an inflection point on $(a, b)$ which is determined by the equation

$$Q[\nabla x (f(t)) \cdot f'(t)]^2 + \langle f'(t), \text{Hessian}_{x} (f(t)) f''(t) \rangle + V I_x (f(t)) f''(t) = 0.$$

**Proof.** By Condition 1 in Definition 4.1, we can choose $c$ close enough to $b$ such that $f''$ is continuous on $[c, b]$. Define

$$\phi_{Q}(c) := Q[\nabla x (f'(c)) \cdot f'(c)]^2 + \langle f'(c), \text{Hessian}_{x} (f'(c)) f''(c) \rangle + V I_x (f(t)) f''(c).$$

Direct computation shows that

$$\frac{d^2 \phi_{Q}(c)}{dt^2} = Q \exp(\mathbb{Q}(\nabla x (f'(c)))) 4 \phi_{Q}(c).$$

Thus, we need only show that $\phi_{Q}(c) = 0$ has a solution on $(c, b)$ for $\mathbb{Q}$ sufficiently large. Observing that $V I_x (f(b)) = V I_x (\kappa) = 0$ and Hessianix $(\kappa)$ is negative definite, we have $\phi_{Q}(b) = \langle f'(b), \text{Hessian}_{x} (f'(b)) f''(b) \rangle < 0$. On the other hand, for $\mathbb{Q}$ sufficiently large

$$\phi_{Q}(c) := Q[\nabla x (f'(c)) \cdot f'(c)]^2 + \langle f'(c), \text{Hessian}_{x} (f'(c)) f''(c) \rangle + \nabla I_x (f(c)) f''(c) > 0. \tag{4.1}$$
Thus, $\phi_Q(t) = 0$ has a solution on $(c, b)$. Q.E.D.

Since $c$ can be chosen arbitrarily close to $b$, with $Q \to \infty$ we will be able to find an inflection point approaching $\kappa$.

On the other hand, the relationship between the risk measure and the parameters in these two paths are different. For Path 2, the risk measure is piecewise linearly related to the parameter $t$. Thus, we can use $t$ as a proxy for the risk and using the method in [14] for one risky asset to determine $t$ values corresponding to the inflection point and the return / risk maximum point. However, this is not the case for Path 1 in which the risk $f_i(t)$, $f_2(t)$ as a nonlinear function of $t$ is not proportional to $t$. Thus, it is not always possible to use the return / risk path to convert a problem involving multiple strategies into one with only one parameter and use the method discussed in [14]. In the following two sections we seek more general ways of finding inflection points and return / risk maximizing points.

5 Determining the manifold of inflection points using Sylvester's criterion

Next we show that, similar to the example discussed Section 2, in general, the function $\rho_4(t)$ is concave near $\kappa$.

**Theorem 5.1.** (Concavity) The function $rQ(f)$ is concave near $\kappa$.

**Proof.** Recall that the concavity of a multi-variable function at a given point is characterized by its Hessian to be negative definite. Calculating the mixed second order partial derivative of $\rho_4(t)$, we have

$$\frac{\partial^2}{\partial f_i \partial f_j} rQ(f) = Q \exp(Q l_X(f)) \left( \frac{\partial^2}{\partial f_i \partial f_j} l_X(f) + Q \frac{\partial l_X(f)}{\partial f_i} \frac{\partial l_X(f)}{\partial f_j} \right).$$

Since, for $i = 1, \ldots, N$,

$$\frac{\partial l_X(\kappa)}{\partial f_i} = 0$$

We have

$$\frac{\partial^2}{\partial f_i \partial f_j} rQ(\kappa) = Q \exp(Q l_X(\kappa)) \frac{\partial^2 l_X(\kappa)}{\partial f_i \partial f_j}.$$ 

It follows that

$$\text{Hessian (} rQ(\kappa) \text{)} = 6 \exp(\delta l_X(\kappa)) \text{Hessian (} \rho_4(\kappa) \text{)}.$$ 

The concavity of $l_X(f)$ implies that Hessian($rQ(\kappa)$) is negative definite. Since the second order partial derivatives of $rQ(f)$ are continuous around $\kappa$, the Hessian of $rQ(\kappa)$ is negative definite in the neighborhood of $\kappa$. Thus, $rQ(f)$ is concave near $\kappa$. Q.E.D.

It is clear that inflection point occurs where the concavity of $rQ(f)$ changes. A characterization of this set is the following:
Theorem 5.2. (Set of inflection points) The inflection point along any risk return path is contained in the following set of $f$ in leverage space characterized by

$$\min \left( -a_{11}(f), \det \begin{bmatrix} a_{11}(f) & a_{12}(f) \\ a_{21}(f) & a_{22}(f) \end{bmatrix}, \ldots, (-1)^N \det \begin{bmatrix} a_{11}(f) & a_{12}(f) & \cdots & a_{1N}(f) \\ a_{21}(f) & a_{22}(f) & \cdots & a_{2N}(f) \\ \vdots & \vdots & \ddots & \vdots \\ a_{N1}(f) & a_{N2}(f) & \cdots & a_{NN}(f) \end{bmatrix} \right) = 0.$$ 

where $a_{ij}(f) := \frac{\partial^2}{\partial f_i \partial f_j} f_N(f) + Q \frac{\partial f_N(f) \partial f_i}{\partial f_j}$.

Proof. Sylvester's criterion for negative definiteness of a matrix tells us that $f_N(i)$ is negative definite if and only if

$$-a_{11}(f) > 0, \det \begin{bmatrix} a_{11}(f) & a_{12}(f) \\ a_{21}(f) & a_{22}(f) \end{bmatrix} > 0, \ldots, (-1)^N \det \begin{bmatrix} a_{11}(f) & a_{12}(f) & \cdots & a_{1N}(f) \\ a_{21}(f) & a_{22}(f) & \cdots & a_{2N}(f) \\ \vdots & \vdots & \ddots & \vdots \\ a_{N1}(f) & a_{N2}(f) & \cdots & a_{NN}(f) \end{bmatrix} > 0.$$ 

Thus, the set in the theorem characterizes the boundary where this condition is violated for the first time—the potential inflection points. Q.E.D.

We note that the computation in determining the set given in Theorem 5.2 is quite involved. A practical (conservative) approximation is

$$\{ f \in R^N : \min (-a_{11}(f), -a_{22}(f), \ldots, -a_{NN}(f)) = 0 \}.$$ 

(5.1) 

6 Determining the manifold of return / risk maximum points

We now turn to the manifolds of return / risk maximum points. We begin by a general definition of a kind of risk measure that covers the several patterns of risks discussed in Section 2 using the idea in [11].

Definition 6.1. We say $m(f)$ is a risk measure coherent to the investment allocation $f$ if $m(0) = 0$ and $m(tf) = tm(f)$ for any $t > 0$.

Coherent risk measures have the following serendipitous property:

Lemma 6.2. Let $m(f)$ be a risk measure coherent to the investment allocation $f$. Then for any $\xi \in dm(f)$, the convex subdifferential of $m$ at $f$, we have

$$m(f) = \langle \xi, f \rangle.$$
Proof. The definition of the convex subdifferential $\xi \in dm(f)$ implies that
\[ m(z) - m(f) \geq (\xi, \zeta - f), Vz. \]
For $t > 0$, setting $z = tf$ in the above inequality and using the coherent property of $m(f)$ yields
\[ (t - 1)m(f) \geq (t - 1)(\xi, f), Vt > 0. \]
Hence
\[ m(f) = (\xi, f). \]
Q.E.D.

Now we can characterize the manifold in leverage space that maximizes the return / risk ratio.

**Theorem 6.3.** Let $m(f)$ be a risk measure coherent to the investment allocation $f$. Then the set of allocations $f$ that maximizes the return / risk ratio $rQ(f)/m(f)$ is represented by
\[ \{ l : (Vr L(f), l) = rQ(f) \}. \]

Proof. We observe that maximize $rQ(f)/m(f)$ is equivalent to the problem
\[ \text{maximize } \frac{rQ(f)}{y} \quad \text{subject to } m(f) - y \leq 0, \]
and the maximum is always attained when $y = m(f)$. The solution to the above problem is characterized by
\[ \left( \frac{f}{y}, \frac{rQ(f)}{y^2} \right) \in \lambda(\partial m(f), -1), \]
or
\[ Vr Q(f) \in X \text{ym}(f) \]
and
\[ rQ(f) = \lambda y^2. \]
Using Lemma 6.2 and inclusion (6.2), we have
\[ (Vr Q(f), l) = \lambda y m(f). \]
Combining this with (6.3) and $y = m(f)$ we arrive at
\[ (Vr Q(f), l) = rQ(l). \]
Q.E.D.

Note that the linear approximation of drawdown is coherent to the investment allocation $l$. Many other types of risk measures also have this coherent property.
7 Applications

We now turn to concrete examples to illustrate how to apply the theory discussed in the previous sections.

Example 7.1. First we recall the example of betting two simultaneous hands of blackjack in [14]. In this example, although there are two players involved, their strategy is the same. Thus, by symmetry we can reason that the two players should use the same bet size. As a result, the problem is reduced such that there is only one bet size to calculate and it can be solved using the methods in [14].

However, in general such a reduction is most of the time impossible.

Example 7.2. A small, private equity firm is considering further funding for two very separate companies over the course of 72 months (6 years) into the future. They wish to maximize their marginal increase in return with respect to marginal increase in risk over this period, investing simultaneously in both companies.

Their study of these companies reveal that for each million provided in funding, Company A will show a monthly profit of $15,300 with a probability of .4 or a loss of $5,000 with a probability of .6.

In considering Company B, there is a .8 probability of a $5,000 monthly profit and a .2 probability of a $9,200 monthly loss.

This would typically lead to the following joint probabilities:

<table>
<thead>
<tr>
<th>Company A</th>
<th>Company B</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>-5000</td>
<td>-9200</td>
<td>.12</td>
</tr>
<tr>
<td>-5000</td>
<td>5000</td>
<td>.48</td>
</tr>
<tr>
<td>15300</td>
<td>-9200</td>
<td>.068</td>
</tr>
<tr>
<td>15300</td>
<td>5000</td>
<td>.32</td>
</tr>
</tbody>
</table>

Table 2. Two Companies

However, further analysis of the past, given that the month-by-month performance of the two companies is in part contingent on business conditions, amends the probabilities to show greater interdependency and our private equity firm deems the probabilities associated with these outcomes, when taken together, to be more accurately:

<table>
<thead>
<tr>
<th>Company A</th>
<th>Company B</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>-5000</td>
<td>-9200</td>
<td>.2</td>
</tr>
<tr>
<td>-5000</td>
<td>5000</td>
<td>.36</td>
</tr>
<tr>
<td>15300</td>
<td>-9200</td>
<td>.06</td>
</tr>
<tr>
<td>15300</td>
<td>5000</td>
<td>.38</td>
</tr>
</tbody>
</table>

Table 3. Two Companies adapted
Thus, the log return function and the cumulative return function in this case is

\[
l(\frac{1}{2}, \frac{1}{2}) = 0.21 \ln(1 - \frac{1}{2}) + 0.36 \ln \left( \frac{1 + \frac{5000}{9200}}{1 - \frac{1}{2}} \right) + 0.06 \ln \left( \frac{1 + \frac{5000}{9200}}{1 - \frac{1}{2}} \right) + 0.38 \ln \left( \frac{1 + \frac{5000}{9200}}{1 - \frac{1}{2}} \right)
\]

and

\[
r(\frac{1}{2}, \frac{1}{2}) = \exp(72K/\frac{1}{2}) - 1,
\]

respectively.

Numerically solving equation

\[
Vr(\frac{1}{2}, \frac{1}{2}) = 0,
\]

we find the peak of the (asymptotic) expected growth-optimal allocations to be \( \kappa = (0.245, 0.121) \) for Company A and Company B, respectively, with an average growth per month of 1.0981493. Next, we consider the risk adjusted return. In this case since we only invest in a relatively short 72 months time-frame, it is reasonable to assume the risk related to the two companies are proportional to the largest monthly loss of -5000 and -9200, respectively. In the leverage space this is to say the risk of company A and B are proportional to \( f_1 \) and \( f_2 \) respectively. Moreover, the probability of such losses occurring simultaneously is relatively high at 20%. Therefore, we assume that the aggregated risk is proportional to their sum \( f_1 + f_2 \). We can see that the path that corresponding to the lease risk for each level set is given by the equation

\[
\frac{dr}{\partial f_1} = \frac{dr}{\partial f_2}.
\]
This path is depicted in Fig. 6 which follows the horizontal axis and then the blue curve. Also depicted in Fig. 6 are the curves corresponding to the inflection points in green and the return / risk maximizing points in red, calculated using the methods in Sections 5 and 6. This gives us an estimate of $\nu_2 = (0.218, 0)$ and $\zeta_2 = (0.23, 0.05)$.

8 Further applications of growth regulator points and the point $\psi$

The properties of this cumulative growth rate curve, these specific points of concern in growth rate regulation are not only applicable to the market trader, the portfolio manager or the gambler, but apply potentially in a plethora of possible activities, ranging from finance, population growths, the spread of pathogens, pharmaceutical dosages, military strategy, game theory, operations research, athletic performance, costs curves in various endeavors, or many arenas where there is a cumulative outcome or feedback mechanism which can range positive or negative from one discrete point to another.

Thus far, we have focused on the notion that an increasing growth is desirable, but this is not always the case with respect to many of these aforementioned growth curves. True, the portfolio manager finds increasing growth to accrue to his benefit, but let us examine, for instance, the growth of a cumulative national debt. Clearly, this is a cumulative growth rate which can range positive to negative (as a function of period-on-period differences between revenues and expenditures) and as such exhibits the same character in terms of growth regulation points ($\nu, \xi, \kappa$) as for the instance of favoring growth. In this case, however, what we considered the benefits of these points are now detriments of them. This example is a one-axis example within the leverage space manifold. Some concerns where we wish to minimize growth may have multiple axes, just as with a portfolio comprised of more than one component having multiple axes in the leverage space manifold in the case of favoring growth.

We now introduce a fourth important point of growth regulation here, $\psi$ as described in [11], representing that point $> \kappa$ (traveling from $l_0 = 0$, along any axis, past $\kappa$) where the cumulative growth rate crosses the altitude of 1.0 to the downside. Such a point of growth regulation represents an allocation along that axis great enough to "bend the curve," and great enough to turn a function which was otherwise increasing into one that is now diminishing.

Just as we have, left of the asymptotic peak, $\kappa$, the points $\nu$ and $\xi$, in that order, "from the left," we have, starting at $\kappa$ and moving "to the right along any axis," the points $\nu$—(the other inflection point) and the $\psi$ point where the growth function crosses 1.0 to the downside. This last point is also a critical point of regulation as any quantity, multiplied by a number less than 1, is accordingly reduced, and if done repeatedly, approaches zero.

Interestingly, this need only occur on one axis and the growth function along all axes is annihilated, and this affords us new insight into the effects of our activity both beneficial
in some cases and detrimental in others.

Let us return to our multiple coin toss example where we are tossing one coin with a 2:1 payoff and \( p = .5 \), simultaneously with a flawed coin that, though pays 1:1 has \( p = .6 \). We can see that at a horizon of 50 simultaneous plays, \( Q = 50 \), \( \kappa \) exists at .243, .180, respectively. We see also that there are many points beyond the peak along any axis where the growth function drops below 1.0, and thus, any combination of allocation values, at a certain level of aggressiveness, will create a \( \psi \) point. But this can also occur along any given axis merely by virtue of being too aggressive along only that axis. If, for example, we were more aggressive with the second coin, the flawed coin, to a level beyond about .50, even though the first coin is still at a .24 allocation, the cumulative growth curve is experiencing the results of a \( \psi \) point.

In this coin toss example, if the first coin had, instead of a .24 allocation, a .23 allocation, then the \( \psi \) point would be found with an allocation to the second coin of .506. If the first coin’s allocation was .25, the \( \psi \) point then occurs at a corresponding .496 allocation to the second coin. Since the caveat to the existence of the leverage space manifold holds that the outcomes from one discrete point to another range from positive to negative, there must be at least one negative point. For example, a growth rate characterized as geometric growth has a constant rate of change - the increases per time period are constant (e.g. \( r(f) = \alpha \)), and therefore does not satisfy this caveat and such a growth rate can not be translated into a leverage space manifold. Because of this necessary condition, in the leverage space manifold there will be a \( \psi \) point along every axis with \( \kappa < \psi \leq 1 \).

For the portfolio manager, the effects can thus be devastating based on only one component within his portfolio. Over-allocating to just one component, even in the absence of borrowing, can undermine the growth of the entire portfolio. This is often contrary to what is generally believed pertaining to the virtue of diversification.

This notion that a \( \psi \) point exists beyond \( \kappa \) for any axis means that growth functions we wish to diminish which consist of multiple components need only have the allocation along one of the components increased to its \( \psi \) point and the entire growth rate becomes a diminishing one.

Achieving such points of growth regulation is not always as simple as it is for the gambler or portfolio manager where the amount wagered or invested can be prescribed so as to commit to a certain point on the curve, but rather is often achieved through practices and/or policy. Nevertheless, these points are at work in all such arenas where there is a cumulative outcome or feedback mechanism which can range positive or negative from one discrete point to another, and no benefit is derived by ignoring these points of growth regulation and steps to achieve them.

Let us consider now how these practices and policies can be used to achieve this \( \psi \) point. From [11] let us consider a time sequence of growth multiples, \( G_1, \ldots, G_n \). Thus each has equal probability, \( 1/n \), and determine its total growth \( (r_n) \) after \( n \) periods as the product of all \( n \) points:

\[
r_n = G_1 \cdot G_2 \cdot \ldots \cdot G_n.
\]
It is important to note that the set of Gs here are a function of /0. .1) selected. However, we will consider the case where the value for / is unknown to us.

We can thus determine the (geometric) mean growth per period

\[ G(f) = r_n(f)^{1/n} \]

Finally, we can determine our growth multiple to a horizon, Q

\[ G_Q(f) = G(f)^Q. \]

However, we can also very closely estimate G(f) by taking the arithmetic mean, A of all G(l . . . n), and the variance, V of G(l . . . n), and we have

\[ G_{est}(f)^2 = A(f)^2 - V(f). \]

Substituting standard deviation for variance, we see this estimate comports to the familiar Pythagorean Theorem

\[ A(f) = \sqrt{G_{est}(f)^2 + SD(f)^2}. \]

Thus, a point is a ψ point if, along a given axis, for given values of / for all other axes, the smallest value for / along the independent axis such that

\[ A(f)^2 - V(f) <\leq 1 \]

where A(f) is the arithmetic average growth multiple (per simultaneous play) and V is the variance in that set of points. Again, the value for / itself is often unknown, as in the case of the per-period growth rates of cumulative federal debt wherein we do not know what the value for / is, even though there is always a value for /, de facto.

We are discussing the case where the implementor cannot control / - a case where we are subject to the rules of the points of regulation (0 < ν < ζ < κ < ν < ψ < 1) upon the curve of expected return, wishing to achieve certain points, but absent being able to inject our selected value for /. However, this estimated value for G demonstrates that we can move the location of the curve along the / value between 0...1 as a function of period-on-period variance, rather than inject a known / value. We see that increases in variance are akin to increasing the value for /.

Acknowledging of these points of important growth regulation and accepting the fact that we can select various points along the curve either by directly injecting values for / along the various axes or by changing the variance with respect to the arithmetic mean of the points, we move the curve between the locations 0,0...0 and 1,1...1, and, in effect, shift the effects of our imposed location on the surface in the leverage space manifold. Thus, through changes in practices and/or policy, we amend our location (between 0...1 for the various axes) on the return curve by changing the variance in the growth rate from
period-to-period without increasing the arithmetic average growth rate. Thus, to diminish a growth rate that satisfies the necessary caveat of the period-on-period outcomes ranging from positive to negative, variance along at least one axis must be increased to a $\psi$ point.

To the gambler or portfolio manager, the values between $\nu$ and $\zeta$ provide for that range of values where he will experience his greatest risk-adjusted return, and at $\kappa$, his greatest return without respect to any other concern. If one is concerned with a case where growth is considered disadvantageous, the implications of these points are reversed. Continuing beyond $\kappa$ and moving rightward along an axis, if one considers growth disadvantageous, then the region between $\kappa$ and $\nu$ — is where growth is being diminished faster than the resource or effort to do so is used or incurred. At $\nu$, this flips, and now marginal increase in resources or effort to bend the curve yield lesser results until the point $\psi$ is achieved, at which point the rate of growth is entirely broken and is now a rate of attenuation rather than growth.

9 Conclusion and further research

Theoretical analysis, Monte Carlo simulation and analysis of real world examples show that the practical consideration of investment in a finite horizon and adjusting for risk, the leverage on different risky investment strategies suggested by the classical growth-optimal portfolio theory needs to be adjusted down considerably. We show that the inflection point and return / risk maximizing point discussed in [14] are reasonable loci in leverage space in cases of capital allocation to risky assets or investment strategies. However, when multiple strategies are involved reducing risk exposure from the growth-optimal allocation, $\kappa$, there exists infinitely many choices of paths. Analyzing the problem path by path is difficult. We established equations for determining the manifolds of inflection points and the return / risk maximizing points. These equations can be solved numerically to determine a region for reasonable choices of leverage. Examples are presented to show how to apply these methods in practice.

As usual, our research here leads to additional questions of practical importance. For example, portfolio insurance allocates a certain percentage to the underlying asset or portfolio based on the delta of a hypothetical option on the underlying asset or portfolio. Thus, portfolio insurance is a practice where one traverses along the return curve between its bounds at 0 and 1 as a function of the underlying price relative to its hypothetical option. The curve is identical in shape to the return curves discussed here, as reinvestment is constantly occurring (and thus, delta and / are interchangeable in this context), yet, the practitioners, in order to mimic the hypothetical option are not capitalizing on any of the important growth-regulation points mentioned in this paper.

Similarly, inverse ETFs and leveraged ETFs (cases of constant reinvestment) have a delta embedded within them as:
This delta too, a number between 0 and some number whose absolute value can be greater than 1, is followed so as to track the promoted criterion of the ETF, e.g. "Triple Long ETF." Yet, as with portfolio insurance, the implementors, in keeping with the promoted criterion of the ETF, are not able to capitalize on the important growth regulating points (perhaps to be used as bounds on the delta). Further research into this area is clearly warranted.

References


Pseudo-Mathematics and Financial Charlatanism: The Effects of Backtest Overfitting on Out-of-Sample Performance

David H. Bailey, Jonathan M. Borwein, Marcos Lopez de Prado, and Qiji Jim Zhu

Another thing I must point out is that you cannot prove a vague theory wrong. [...] Also, if the process of computing the consequences is indefinite, then with a little skill any experimental result can be made to look like the expected consequences.

—Richard Feynman [1964]

Introduction

A backtest is a historical simulation of an algorithmic investment strategy. Among other things, it computes the series of profits and losses that such strategy would have generated had that algorithm been run over that time period. Popular performance statistics, such as the Sharpe ratio or the Information ratio, are used to quantify the backtested strategy’s return on risk. Investors typically study those backtest statistics and then allocate capital to the best performing scheme.

Regarding the measured performance of a backtested strategy, we have to distinguish between two very different readings: in-sample (IS) and out-of-sample (OOS). The IS performance is the one simulated over the sample used in the design of the strategy (also known as "learning period" or "training set" in the machine-learning literature). The OOS performance is simulated over a sample not used in the design of the strategy (a.k.a. "testing set"). A backtest is realistic when the IS performance is consistent with the OOS performance.

When an investor receives a promising backtest from a researcher or portfolio manager, one of her key problems is to assess how realistic that simulation is. This is because, given any financial series, it is relatively simple to overfit an investment strategy so that it performs well IS.

Overfitting is a concept borrowed from machine learning and denotes the situation when a model targets particular observations rather than a general structure. For example, a researcher could design a trading system based on some parameters that target the removal of specific recommendations that she knows led to losses IS (a practice known as "data snooping"). After a few iterations, the researcher will come up with "optimal parameters", which profit from features that are present in that particular sample but may well be rare in the population.

Recent computational advances allow investment managers to methodically search through thousands or even millions of potential options for a profitable investment strategy. In many instances, that search involves a pseudo-mathematical argument which is spuriously validated through a backtest. For example, consider a time series of daily prices for a stock $X$. For every day in the sample, we can compute one average price of that stock using the previous $m$ observations $\bar{x}_m$ and another average price using the previous $n$ observations $\bar{x}_n$ where $m < n$. A popular investment strategy called "crossing moving averages" consists of owning $X$ whenever $\bar{x}_m > \bar{x}_n$. Indeed, since the sample size determines a limited number of parameter combinations that $m$ and $n$ can adopt,
it is relatively easy to determine the pair \((m, n)\) that maximizes the backtest's performance. There are hundreds of such popular strategies, marketed to unsuspecting lay investors as mathematically sound and empirically tested.

In the context of econometric models several procedures have been proposed to determine overfit in White [27], Romano et al. [23], and Harvey et al. [9]. These methods propose to adjust the p-values of estimated regression coefficients to account for the multiplicity of trials. These approaches are valuable for dealing with trading rules based on an econometric specification.

The machine-learning literature has devoted significant effort to studying the problem of overfitting. The proposed methods typically are not applicable to investment problems for multiple reasons. First, these methods often require explicit point forecasts and confidence bands over a defined event horizon in order to evaluate the explanatory power or quality of the prediction (e.g., "E-mini S&P500 is forecasted to be around 1,600 with a one-standard deviation of 5 index points at Friday's close"). Very few investment strategies yield such explicit forecasts; instead, they provide qualitative recommendations (e.g., "buy" or "strong buy") over an undefined period until another such forecast is generated, with random frequency. For instance, trading systems, like the crossing of moving averages explained earlier, generate buy and sell recommendations with little or no indication as to forecasted values, confidence in a particular recommendation, or expected holding period.

Second, even if a particular investment strategy relies on such a forecasting equation, other components of the investment strategy may have been overfitted, including entry thresholds, risk sizing, profit taking, stop-loss, cost of capital, and so on. In other words, there are many ways to overfit an investment strategy other than simply tuning the forecasting equation. Third, regression overfitting methods are parametric and involve a number of assumptions regarding the underlying data which may not be easily ascertainable. Fourth, some methods do not control for the number of trials attempted.

To illustrate this point, suppose that a researcher is given a finite sample and told that she needs to come up with a strategy with an SR (Sharpe Ratio, a popular measure of performance in the presence of risk) above 2.0, based on a forecasting equation for which the AIC statistic (Akaike Information Criterion, a standard of the regularization method) rejects the null hypothesis of overfitting with a 95 percent confidence level (i.e., a false positive rate of 5 percent). After only twenty trials, the researcher is expected to find one specification that passes the AIC criterion. The researcher will quickly be able to present a specification that not only (false) passes the AIC test but also gives an SR above 2.0. The problem is, AIC's assessment did not take into account the hundreds of other trials that the researcher neglected to mention. For these reasons, commonly used regression overfitting methods are poorly equipped to deal with backtest overfitting.

Although there are many academic studies that claim to have identified profitable investment strategies, their reported results are almost always based on IS statistics. Only exceptionally do we find an academic study that applies the "hold-out" method or some other procedure to evaluate performance OOS. Harvey, Liu, and Zhu [10] argue that there are hundreds of papers supposedly identifying hundreds of factors with explanatory power over future stock returns. They echo Ioannidis [13] in concluding that "most claimed research findings are likely false." Factor models are only the tip of the iceberg. The reader is probably familiar with many publications solely discussing IS performance.

This situation is, quite frankly, depressing, particularly because academic researchers are expected to recognize the dangers and practice of overfitting. One common criticism, of course, is the credibility problem of "holding-out" when the researcher had access to the full sample anyway. Leinweber and Sisk [15] present a meritorious exception. They proposed an investment strategy in a conference and announced that six months later they would publish the results with the pure (yet to be observed) OOS data. They called this approach "model sequestration", which is an extreme variation of "hold-out".

**Our Intentions**

In this paper we shall show that it takes a relatively small number of trials to identify an investment strategy with a spuriously high backtested performance. We also compute the minimum backtest length (MinBTL) that an investor should require given the number of trials attempted. Although in our examples we always choose the Sharpe ratio to evaluate performance, our methodology can be applied to any other performance measure.

We believe our framework to be helpful to the academic and investment communities by providing a benchmark methodology to assess the reliability of a backtested performance. We would

---

1We invite the reader to read specific instances of pseudo-mathematical financial advice at this website: http://www.m-a-f-f-i -a.org. Also, Edesses (2007) provides numerous examples.
Historically, scientists have led the way in exposing those who utilize pseudoscience to extract a commercial benefit. As early as the eighteenth century, physicists exposed the nonsense of astrologers. Yet mathematicians in the twenty-first century have remained disappointingly silent with regard to those in the investment community who, knowingly or not, misuse mathematical techniques such as probability theory, statistics, and stochastic calculus. Our silence is consent, making us accomplices in these abuses.

The rest of our study is organized as follows: The section "Backtest Overfitting" introduces the problem in a more formal way. The section "Minimum Backtest Length (MinBTL)" defines the concept of Minimum Backtest Length (MinBTL). The section "Model Complexity" argues how model complexity leads to backtest overfitting. The section "Overfitting in Absence of Compensation Effects" analyzes overfitting in the absence of compensation effects. The section "Overfitting in Presence of Compensation Effects" studies overfitting in the presence of compensation effects. The section "Is Backtest Overfitting a Fraud?" exposes how backtest overfitting can be used to commit fraud. The section "A Practical Application" presents a typical example of backtest overfitting. The section "Conclusions" lists our conclusions. The mathematical appendices supply proofs of the propositions presented throughout the paper.

**Backtest Overfitting**

The design of an investment strategy usually begins with a *prior or belief* that a certain pattern may help forecast the future value of a financial variable. For example, if a researcher recognizes a lead-lag effect between various tenor bonds in a yield curve, she could design a strategy that bets on a reversion towards equilibrium values. This model might take the form of a cointegration equation, a vector-error correction model, or a system of stochastic differential equations, just to name a few. The number of possible model configurations (or trials) is enormous, and naturally the researcher would like to select the one that maximizes the performance of the strategy. Practitioners often rely on *historical simulations* (also called backtests) to discover the optimal specification of an investment strategy. The researcher will evaluate, among other variables, what are the optimal sample sizes, signal update frequency, entry and profit-taking thresholds, risk sizing, stop losses, maximum holding periods, etc.

The *Sharpe ratio* is a statistic that evaluates an investment manager's or strategy's performance on the basis of a sample of past returns. It is defined as the ratio between average excess returns (in excess of the rate of return paid by a risk-free asset, such as

![Figure 1. Overfitting a backtest's results as the number of trials grows.](image)

Figure 1 provides a graphical representation of Proposition 1. The blue (dotted) line shows the maximum of a particular set of $N$ independent random numbers, each following a Standard Normal distribution. The black (continuous) line is the expected value of the maximum of that set of $N$ random numbers. The red (dashed) line is an upper bound estimate of that maximum. The implication is that it is relatively easy to wrongly select a strategy on the basis of a maximum Sharpe ratio when displayed IS.

feel sufficiently rewarded in our efforts if at least this paper succeeded in drawing the attention of the mathematical community regarding the widespread proliferation of journal publications, many of them claiming profitable investment strategies on the sole basis of its performance. This is perhaps understandable in business circles, but a higher standard is and should be expected from an academic forum.

We would also like to raise the question of whether mathematical scientists should continue to tolerate the proliferation of investment products that are misleadingly marketed as mathematically sound. In the recent words of Sir Andrew Wiles,

One has to be aware now that mathematics can be misused and that we have to protect its good name. [29]

We encourage the reader to search the Internet for terms such as "stochastic oscillators", "Fibonacci ratios", "cycles", "Elliot wave", "Golden ratio", "parabolic SAR", "pivot point", "momentum", and others in the context of finance. Although such terms clearly evoke precise mathematical concepts, in fact in almost all cases their usage is scientifically unsound.
a government note) and the standard deviation of the same returns. Intuitively, this can be interpreted as a "return on risk" (or as William Sharpe put it, "return on variability"). But the standard deviation of excess returns may be a misleading measure of variability when returns follow asymmetric or fat-tailed distributions or when returns are not independent or identically distributed. Suppose that a strategy's excess returns (or risk premiums), \( r_t \), are independent and identically distributed (IID) following a Normal law:

\[
(1) \quad r_t \sim \mathcal{N}(\mu, \sigma^2),
\]

where \( \mathcal{N} \) represents a Normal distribution with mean \( \mu \) and variance \( \sigma^2 \). The annualized Sharpe ratio (SR) can be computed as

\[
(2) \quad SR = \frac{\mu}{\sigma} \sqrt{T},
\]

where \( q \) is the number of returns per year (see Lo [17] for a detailed derivation of this expression).

Sharpe ratios are typically expressed in annual terms in order to allow for the comparison of strategies that trade with different frequency. The great majority of financial models are built upon the IID Normal assumption, which may explain why the Sharpe ratio has become the most popular statistic for evaluating an investment's performance.

Since \( \mu, \sigma \) are usually unknown, the true value SR cannot be known for certain. Instead, we can estimate the Sharpe ratio as \( \hat{SR} = \frac{\hat{\mu}}{\hat{\sigma}} \sqrt{T} \), where \( \hat{\mu} \) and \( \hat{\sigma} \) are the sample mean and sample standard deviation. The inevitable consequence is that SR calculations are likely to be subject of substantial estimation errors (see Bailey and Lopez de Prado [2] for a confidence band and an extension of the concept of Sharpe Ratio beyond the IID Normal assumption).

From Lo [17] we know that the distribution of the estimated annualized Sharpe ratio \( \hat{SR} \) converges asymptotically (as \( y \to \infty \)) to

\[
(3) \quad \frac{\hat{SR}}{\sqrt{\frac{1 + \frac{\hat{SR}^2}{2\sigma^2}}{y}}} \sim \mathcal{N}(\sqrt{\frac{1}{y}}, \frac{1 + \frac{\hat{SR}^2}{2\sigma^2}}{y}),
\]

where \( y \) is the number of years used to estimate \( \hat{SR} \). As \( y \) increases without bound, the probability distribution of \( \hat{SR} \) approaches a Normal distribution with mean \( \sqrt{\frac{1}{y}} \) and variance \( \frac{1 + \frac{\hat{SR}^2}{2\sigma^2}}{y} \). For a sufficiently large \( y \), (3) provides an approximation of the distribution of \( \hat{SR} \).

Even for a small number \( N \) of trials, it is relatively easy to find a strategy with a high Sharpe ratio IS but which also delivers a null Sharpe ratio OOS. To illustrate this point, consider \( N \) strategies with \( T = y \) returns distributed according to a Normal law with mean excess returns \( \mu \) and with standard deviation \( \sigma \). Suppose that we would like to select the strategy with optimal \( \hat{SR} \) IS, based on one year of observations. A risk we face is choosing a strategy with a high Sharpe ratio IS but zero Sharpe ratio OOS. So we ask the question, how high is the expected maximum Sharpe ratio IS among a set of strategy configurations where the true Sharpe ratio is zero?

Figure 2 shows the tradeoff between the number of trials (\( N \)) and the minimum backtest length (MinBTL) needed to prevent skill-less strategies to be generated with a Sharpe ratio IS of 1. For instance, if only five years of data are available, no more than forty-five independent model configurations should be tried. For that number of trials, the expected maximum SR IS is 1, whereas the expected SR OOS is 0. After trying only seven independent strategy configurations, the expected maximum SR IS is 1 for a two-year long backtest, while the expected SR OOS is 0. The implication is that a backtest which does not report the number of trials \( N \) used to identify the selected configuration makes it impossible to assess the risk of overfitting.
Figure 3. Performance IS vs. OOS before introducing strategy selection.

Figure 3 shows the relation between SR IS (x-axis) and SR OOS (y-axis) for \( \mu = 0, \sigma = 1, N = 1000, T = 1000 \). Because the process follows a random walk, the scatter plot has a circular shape centered at the point (0, 0). This illustrates the fact that, in absence of compensation effects, overfitting of the IS performance (x-axis) has no bearing on the OOS performance (y-axis), which remains around zero.

(let's say zero). MinTRL was developed to evaluate a strategy's track record (a single realized path, \( N = 1 \)). The question we are asking now is different, because we are interested in the backtest length needed to avoid selecting a skill-less strategy among \( N \) alternative specifications. In other words, in this article we are concerned with overfitting prevention when comparing multiple strategies, not in evaluating the statistical significance of a single Sharpe ratio estimate. Next, we will derive the analogue to MinTRL in the context of overfitting, which we will call Minimum Backtest Length (MinBTL), since it specifically addresses the problem of backtest overfitting.

From (3), if \( \mu = 0 \) and \( y = 1 \), then \( \sqrt{SR^*} - \mathcal{N}(0, 1) \). Note that because \( SR = 0 \), increasing \( q \) does not reduce the variance of the distribution. The proof of the following proposition is left for the appendix.

**Proposition 1.** Given a sample of IID random variables, \( x_i \sim Z, n = 1, \ldots, N \), where \( Z \) is the CDF of the Standard Normal distribution, the expected maximum of that sample, \( E[\max(x)] = E[\max\{\frac{1}{N}\}] \), can be approximated for a large \( N \) as

\[
E[\max] \approx (1 - y)Z^{-1} \left[ 1 - \frac{1}{N} \right] + yZ^{-1} \left[ 1 - \frac{1}{N} e^{-1} \right]
\]

where \( y \approx 0.5772156649 \ldots \) is the Euler-Mascheroni constant and \( N \gg 1 \).

An upper bound to (4) is \( \sqrt{2 \ln[N]} \). Figure 1 plots, for various values of \( N \) (x-axis), the expected Sharpe ratio of the optimal strategy. For example, if the researcher tries only \( N = 10 \) alternative configurations of an investment strategy, she is expected to find a strategy with a Sharpe ratio IS of 1.57 despite the fact that all strategies are expected to deliver a Sharpe ratio of zero OOS (including the "optimal" one selected IS).

Proposition 1 has important implications. As the researcher tries a growing number of strategy configurations, there will be a nonnull probability of selecting IS a strategy with null expected performance OOS. Because the hold-out method does not take into account the number of trials attempted before selecting a model, it cannot assess the representativeness of a backtest.

**Minimum Backtest Length (MinBTL)**

Let us consider now the case that \( \mu = 0 \) but \( y \neq 1 \). Then, we can still apply Proposition 1 by rescaling the expected maximum by the standard deviation of the annualized Sharpe ratio, \( y^{-1/2} \). Thus, the researcher is expected to find an "optimal" strategy with an IS annualized Sharpe ratio of

\[
E[\max] \approx y^{-1/2} \left( (1 - y)Z^{-1} \left[ 1 - \frac{1}{N} \right] + yZ^{-1} \left[ 1 - \frac{1}{N} e^{-1} \right] \right)
\]

Equation (5) says that the more independent configurations a researcher tries \( N \), the more likely she is to overfit, and therefore the higher the acceptance threshold should be for the backtested result to be trusted. This situation can be partially mitigated by increasing the sample size \( y \). By solving (5) for \( y \), we obtain the following statement.

**Theorem 2.** The Minimum Backtest Length (MinBTL, in years) needed to avoid selecting a strategy with an IS Sharpe ratio of \( E[\max(x)] \) among \( N \) independent strategies with an expected OOS Sharpe ratio of zero is

\[
\text{MinBTL} = \left( \frac{(1 - y)Z^{-1} \left[ 1 - \frac{1}{N} \right] + yZ^{-1} \left[ 1 - \frac{1}{N} e^{-1} \right]}{E[\max(x)]} \right)^2 < 2 \ln[N] \left( \frac{E[\max(x)]}{\ln[1]} \right)^2
\]

Equation (6) tells us that MinBTL must grow as the researcher tries more independent model configurations \( N \) in order to keep constant the expected maximum Sharpe ratio at a given level.

\(^3\text{See Example 3.5 of Embrechts et al. [5] for a detailed treatment of the derivation of upper bounds on the maximum of a Normal distribution.} \)
Figure 2 shows how many years of backtest length (MinBTL) are needed so that \( E[\max X] \) is fixed at 1. For instance, if only five years of data are available, no more than forty-five independent model configurations should be tried or we are almost guaranteed to produce strategies with an annualized Sharpe ratio IS of 1 but an expected Sharpe ratio OOS of zero. Note that Proposition 1 assumed the \( N \) trials to be independent, which leads to a quite conservative estimate. If the trials performed were not independent, the number of independent trials \( N \) involved could be derived using a dimension-reduction procedure, such as Principal Component Analysis.

We will examine this tradeoff between \( N \) and \( T \) in greater depth later in the paper without requiring such a strong assumption, but MinBTL gives us a first glance at how easy it is to overfit by merely trying alternative model configurations. As an approximation, the reader may find it helpful to remember the upper bound to the minimum backtest length (in years), \( \text{MinBTL} < \frac{2 \ln(N)}{T} \).

Of course, a backtest may be overfit even if it is computed on a number greater than MinBTL. From that perspective, MinBTL should be considered a necessary, non-sufficient condition to avoid overfitting. We leave to Bailey et al. \([1]\) the derivation of a more precise measure of backtest overfitting.

**Model Complexity**

How does the previous result relate to model complexity? Consider a one-parameter model that may adopt two possible values (like a switch that generates a random sequence of trades) on a sample of \( T \) observations. Overfitting will be difficult, because \( N = 2 \). Let's say that we make the model more complex by adding four more parameters so that the total number of parameters becomes 5, i.e., \( N = 2^5 = 32 \). Having thirty-two independent sequences of random trades greatly increases the possibility of overfitting.

While a greater \( N \) makes overfitting easier, it makes perfectly fitting harder. Modern supercomputers can only perform around \( 2^{50} \) raw computations per second, or less than \( 2^{58} \) raw computations per year. Even if a trial could be reduced to a raw computation, searching \( N = 2^{100} \) will take us \( 2^{42} \) supercomputer-years of computation (assuming a 1 Petaflop/s system, capable of \( 10^{15} \) floating-point operations per second). Hence, a skill-less brute force search is certainly impossible. While it is hard to perfectly fit a complex skill-less strategy, Proposition 1 shows that there is no need for that. Without perfectly fitting a strategy or making it overcomplex, a researcher can achieve high Sharpe ratios. A relatively simple strategy with just seven binomial independent parameters offers \( N = 2^7 = 128 \) trials, with an expected maximum Sharpe ratio above 2.6 (see Figure 1).

We suspect, however, that backtested strategies that significantly beat the market typically rely on some combination of valid insight, boosted by some degree of overfitting. Since believing in such an artificially enhanced high-performance strategy will often also lead to overleveraging, such overfitting is still very damaging. Most Technical Analysis strategies rely on filters, which are sets of conditions that trigger trading actions, like the random switches exemplified earlier. Accordingly, extra caution is warranted to guard against overfitting in using Technical Analysis strategies, as well as in complex nonparametric modeling tools, such as Neural Networks and Kernel Estimators.

Here is a key concept that investors generally miss:

A researcher that does not report the number of trials \( N \) used to identify the selected backtest configuration makes it impossible to assess the risk of overfitting.

Because \( N \) is almost never reported, the magnitude of overfitting in published backtests is unknown. It is not hard to overfit a backtest (indeed, the previous theorem shows that it is hard not to), so we suspect that a large proportion of backtests published in academic journals may be misleading. The
Overfitting in Absence of Compensation Effects

Regardless of how realistic the prior being tested is, there is always a combination of parameters that is optimal. In fact, even if the prior is false, the researcher is very likely to identify a combination of parameters that happens to deliver an outstanding performance IS. But because the prior is false, OOS performance will almost certainly underperform the backtest’s results. As we have described, this phenomenon, by which IS results tend to outperform the OOS results, is called overfitting. It occurs because a sufficiently large number of parameters are able to target specific data points—say by chance buying just before a rally and shorting a position just before a sell-off—rather than triggering trades according to the prior.

To illustrate this point, suppose we generate $N$ Gaussian random walks by drawing from a Standard Normal distribution, each walk having a size $T$. Each performance path $m_T$ can be obtained as a cumulative sum of Gaussian draws

$$A m_T = \mu + \sigma \varepsilon_T,$$

where the random shocks $\varepsilon_T$ are IID distributed $\varepsilon_T \sim Z, T = 1, \ldots, T$. Suppose that each path has been generated by a particular combination of parameters, backtested by a researcher. Without loss of generality, assume that $\mu = 0, \sigma = 1$, and $T = 1000$, covering a period of one year (with about four observations per trading day). We divide these paths into two disjoint samples of equal size, 500, and call the first one IS and the second one OOS.

At the moment of choosing a particular parameter combination as optimal, the researcher had access to the IS series, not the OOS. For each model configuration, we may compute the Sharpe ratio of the series IS and compare it with the Sharpe ratio of the series OOS. Figure 3 shows the resulting scatter plot. The $p$-values associated with the intercept and the IS performance (SR a priori) are respectively 0.6261 and 0.7469.

The problem of overfitting arises when the researcher uses the IS performance (backtest) to choose a particular model configuration, with the expectation that configurations that performed well in the past will continue to do so in the future. This would be a correct assumption if the parameter configurations were associated with a truthful prior, but this is clearly not the case of the simulation above, which is the result of Gaussian random walks without trend ($\mu = 0$).

Figure 4 shows what happens when we select the model configuration associated with the random walk with highest Sharpe ratio IS. The performance of the first half was optimized IS, and the performance of the second half is what the investor receives OOS. The good news is that under these situations is not likely to be better among practitioners. In our experience, overfitting is pathological within the financial industry, where proprietary and commercial software is developed to estimate the combination of parameters that best fits (or, more precisely, overfits) the data. These tools allow the user to add filters without ever reporting how such additions increase the probability of backtest overfitting. Institutional players are not immune to this pitfall. Large mutual fund groups typically discontinue and replace poorly performing funds, introducing survival bias and selection bias. While the motivation of this practice may be entirely innocent, the effect is the same as that of hiding experiments and inflating expectations.

We are not implying that those technical analysts, quantitative researchers, or fund managers are “snake oil salesmen”. Most likely they most genuinely believe that the backtested results are legitimate or that adjusted fund offerings better represent future performance. Hedge fund managers are often unaware that most backtests presented to them by researchers and analysts may be useless, and so they unknowingly package faulty investment propositions into products. One goal of this paper is to make investors, practitioners, and academics aware of the futility of considering backtest without controlling for the probability of overfitting.

Figure 5 illustrates what happens once we add a “model selection” procedure. Now the SR IS ranges from 1.2 to 2.6, and it is centered around 1.7. Although the backtest for the selected model generates the expectation of a 1.7 SR, the expected SR OOS is unchanged and lies around 0.

---

Figure 5. Performance degradation after introducing strategy selection in absence of compensation effects.
conditions, there is no reason to expect overfitting to induce negative performance. This is illustrated in Figure 5, which shows how the optimization causes the expected performance IS to range between 1.2 and 2.6, while the OOS performance will range between -1.5 and 1.5 (i.e., around $\mu$, which in this case is zero). The p-values associated with the intercept and the IS performance (SR a priori) are respectively 0.2146 and 0.2131. Selecting an optimal model IS had no bearing on the performance OOS, which simply equals the zero mean of the process. A positive mean ($\mu > 0$) would lead to positive expected performance OOS, but such performance would nevertheless be inferior to the one observed IS.

**Overfitting in Presence of Compensation Effects**

Multiple causes create compensation effects in practice, such as overcrowded investment opportunities, major corrections, economic cycles, reversal of financial flows, structural breaks, bubble bursts, etc. Optimizing a strategy’s parameters (i.e., choosing the model configuration that maximizes the strategy’s performance IS) does not necessarily lead to improved performance (compared to not optimizing) OOS, yet again leading to overfitting.

In some instances, when the strategy’s performance series lacks memory, overfitting leads to no improvement in performance OOS. However, the presence of memory in a strategy's performance series induces a compensation effect, which increases the chances for that strategy to be selected IS, only to underperform the rest OOS. Under these circumstances, IS backtest optimization is in fact detrimental to OOS performance. 4

**Global Constraint**

Unfortunately, overfitting rarely has the neutral implications discussed in the previous section. Our previous example was purposely chosen to exhibit a globally unconditional behavior. As a result, the OOS data had no memory of what occurred IS. Centering each path to match a mean $\mu$ removes one degree of freedom:

$$\bar{A} m_T = A m_T + \mu - \frac{1}{T} \sum_{t=1}^{T} A m_T.$$  

Figure 6 shows that adding a single global constraint causes the OOS performance to be negative even though the underlying process was trendless. Also, a strongly negative linear relation between performance IS and OOS arises, indicating that the more we optimize IS, the worse the OOS performance of the strategy.

We may rerun the same Monte Carlo experiment as before, this time on the centered variables $\bar{A} m_T$. Somewhat scarily, adding this single global constraint causes the OOS performance to be negative even though the underlying process was trendless. Moreover, a strongly negative linear relation between performance IS and OOS arises, indicating that the more we optimize IS, the worse the OOS performance. Figure 6 displays this disturbing pattern. The p-values associated with the intercept and the IS performance (SR a priori) are respectively 0.5005 and 0, indicating that the negative linear relation between IS and OOS Sharpe ratios is statistically significant.

The following proposition is proven in the appendix.

**Proposition 3.** Given two alternative configurations (A and B) of the same model, where $\sigma_{IS}^A = \sigma_{OOS}^A = \sigma_{OOS}^B = \sigma_{IS}^B$ imposing a global constraint $\mu^A = \mu^B$, in place of $T \times t$

$$SR_A > SR_B \iff S_r^A_{OOS} < S_r^B_{OOS}.$$
Figure 7 illustrates that a serially correlated performance introduces another form of compensation effects, just as we saw in the case of a global constraint. For example, if $\phi = 0.995$, it takes about 138 observations to recover half of the deviation from the equilibrium. We have rerun the previous Monte Carlo experiment, this time on an autoregressive process with $\mu = 0$, $\sigma = 1$, $\phi = 0.995$, and have plotted the pairs of performance IS vs. OOS.

Recentering a series is one way to introduce memory into a process, because some data points will now compensate for the extreme outcomes from other data points. By optimizing a backtest, the researcher selects a model configuration that spuriously works well IS and consequently is likely to generate losses OOS.

Serial Dependence

Imposing a global constraint is not the only situation in which overfitting actually is detrimental. To cite another (less restrictive) example, the same effect happens if the performance series is serially conditioned, such as a first-order autoregressive process.

$$A m_T = (1 - \phi) \mu + (\phi - 1) \sigma m_{T-1} + \sigma \varepsilon_t$$

or, analogously,

$$m_T = (1 - \phi) \mu + \phi m_{T-1} + \sigma \varepsilon_t,$$

where the random shocks are again IID distributed as $\varepsilon_t \sim \mathcal{N}(0, 1)$. The following proposition is proven in the appendix. The number of observations that it takes for a process to reduce its divergence from the long-run equilibrium by half is known as the half-life period, or simply half-life (a familiar physical concept introduced by Ernest Rutherford in 1907).

**Proposition 4.** The half-life period of a first-order autoregressive process with autoregressive coefficient $\phi \in (0, 1)$ occurs at

$$\tau = \frac{\ln[n]}{\ln(\phi)}.$$  

For example, if $\phi = 0.995$, it takes about 138 observations to retrace half of the deviation from the equilibrium. This introduces another form of compensation effect, just as we saw in the case of a global constraint. If we rerun the previous Monte Carlo experiment, this time for the autoregressive process with $\mu = 0, \sigma = 1, \phi = 0.995$, and plot the pairs of performance IS vs. OOS, we obtain Figure 7.

The $p$-values associated with the intercept and the IS performance (SR a priori) are respectively 0.4513 and 0, confirming that the negative linear relation between IS and OOS Sharpe ratios is again statistically significant. Such serial correlation is a well-known statistical feature, present in the performance of most hedge fund strategies. Proposition 5 is proved in the appendix.

**Proposition 5.** Given two alternative configurations (A and B) of the same model, where $\sigma^A_{IS} = \sigma^A_{OOS} = \sigma^B_{IS} = \sigma^B_{OOS}$ and the performance series follows the same first-order autoregressive stationary process,

$$SR^A_{IS} > SR^B_{IS} \Leftrightarrow SR^B_{OOS} < SR^A_{OOS},$$

Proposition 5 reaches the same conclusion as Proposition 3 (a compensation effect) without requiring a global constraint.

**Is Backtest Overfitting a Fraud?**

Consider an investment manager who emails his stock market forecast for the next month to $2^n x$ prospective investors, where $x$ and $n$ are positive integers. To half of them he predicts that markets will go up, and to the other half that markets will go down. After the month passes, he drops from his list the names to which he sent the incorrect forecast, and sends a new forecast to the remaining $2^{n-1}x$ names. He repeats the same procedure $n$ times, after which only $x$ names remain. These $x$ investors have witnessed $n$ consecutive infallible forecasts and maybe extremely tempted to give this investment manager all of their savings. Of course, this is a fraudulent scheme based on random screening: The investment manager is hiding the fact that for every one of the $x$ successful witnesses, he has tried $2^n$ unsuccessful ones (see Harris [8, p. 473] for a similar example).

To avoid falling for this psychologically compelling fraud, a potential investor needs to consider the economic cost associated with manufacturing the successful experiments, and require the investment manager to produce a number $n$ for which...
the scheme is uneconomic. One caveat is, even if \( n \) is too large for a skill-less investment manager, it may be too low for a mediocre investment manager who uses this scheme to inflate his skills.

Not reporting the number of trials (\( N \)) involved in identifying a successful backtest is a similar kind of fraud. The investment manager only publicizes the model that works but says nothing about all the failed attempts, which as we have seen can greatly increase the probability of backtest overfitting.

An analogous situation occurs in medical research, where drugs are being tested by hundreds or thousands of patients; however, only the best outcomes are publicized. The reality is that the selected outcomes may have been heated in spite of (rather than thanks to) the treatment or due to a placebo effect (recall Proposition 1). Such behavior is unscientific—not to mention dangerous and expensive—and has led to the launch of the alltrials.net project, which demands that all results (positive and negative) for every experiment are made publicly available. A step forward in this direction is the recent announcement by Johnson & Johnson that it plans to open all of its clinical test results to the public [14]. For a related discussion of reproducibility in the context of mathematical computing, see Stodden et al. [25].

Hiding trials appears to be standard procedure in financial research and financial journals. As an aggravating factor, we know from the section "Overfitting in Presence of Compensation Effects" that backtest overfitting typically has a detrimental effect on future performance due to the compensation effects present in financial series. Indeed, the customary disclaimer "past performance is not an indicator of future results" is too optimistic in the context for backtest overfitting. When investment advisers do not control for backtest overfitting, good backtest performance is an indicator of negative future results.

**A Practical Application**

Institutional asset managers follow certain investment procedures on a regular basis, such as rebalancing the duration of a fixed income portfolio (PIMCO); rolling holdings on commodities (Goldman Sachs, AIG, JP Morgan, Morgan Stanley); investing or divesting as new funds flow at the end of the month (Fidelity, BlackRock); participating in the regular U.S. Treasury Auctions (all major investment banks); delivering in anticipation of payroll, FOMC or GDP releases; tax-driven effects around the end of the year and mid-April; positioning for electoral cycles, etc. There are a large number of instances where asset managers will engage in somewhat predictable actions on a regular basis. It should come as no surprise that a very popular investment strategy among hedge funds is to profit from such seasonal effects.

For example, a type of question often asked by hedge fund managers follows the form: "Is there a time interval every [ ] when I would have made money on a regular basis?" You may replace the blank space with a word like day, week, month, quarter, auction, nonfarm payroll (NFP) release, European Central Bank (ECB) announcement, presidential election year, . . . The variations are as abundant as they are inventive. Doyle and Chen [4] study the "weekday effect" and conclude that it appears to "wander".

The problem with this line of questioning is that there is always a time interval that is arbitrarily "optimal" regardless of the cause. The answer to one such question is the title of a very popular investment classic, _Do Not Sell Stocks on Monday_, by Hirsch [12]. The same author wrote an almanac for stock traders that reached its forty-fifth edition in 2012 and he is also a proponent of the "Santa Claus Rally", the quadrennial political/stock market cycle, and investing during the "Best Six Consecutive Months" of the year, November through April. While these findings may indeed be caused by some underlying seasonal effect, it is easy to demonstrate that any random data contains similar patterns. The discovery of a pattern is typically has no bearing OOS, yet again is a result of overfitting. Running such experiments.

---

**Figure 8. Backtested performance of a seasonal strategy (Example 6).**

We have generated a time series of 1000 daily prices (about four years) following a random walk. The PSR-Stat of the optimal model configuration is 2.83, which implies a less-than 1 percent probability that the true Sharpe ratio is below 0. Consequently, we have been able to identify a plausible seasonal strategy with an \( SR \) of 1.27 despite the fact that no true seasonal effect exists.
without controlling for the probability of backtest overfitting will lead the researcher to spurious claims. OOS performance will disappoint, and the reason will not be that "the market has found out the seasonal effect and arbitrage away the strategy's profits." Rather, the effect was never there; instead, it was just a random pattern that gave rise to an overfitted trading rule. We will illustrate this point with an example.

**Example 6.** Suppose that we would like to identify the optimal monthly trading rule given four customary parameters: Entry day, Holding period, Stop loss, and Side. *Side* defines whether we will hold long or short positions on a monthly basis. *Entry day* determines the business day of the month when we enter a position. *Holding period* gives the number of days that the position is held. *Stop loss* determines the size of the loss (as a multiple of the series’ volatility) that triggers an exit for that month’s position. For example, we could explore all nodes that span the set {1, ..., 22} for *Entry day*, the set {1, ..., 20} for *Holding period*, the set {0, ..., 10} for *Stop loss*, and {-1, 1} for *Side*. The parameter combinations involved form a four-dimensional mesh of 8,800 elements. The optimal parameter combination can be discovered by computing the performance derived by each node.

First, we generated a time series of 1,000 daily prices (about four years), following a random walk. Figure 8 plots the random series, as well as the performance associated with the optimal parameter combination: *Entry day* = 1, *Holding period* = 4, *Stop loss* = -1 and *Side* = 1. The annualized Sharpe ratio is 1.27.

Given the elevated Sharpe ratio, we could conclude that this strategy’s performance is significantly greater than zero for any confidence level. Indeed, the *PSR-Stat* is 2.83, which implies a less than 1 percent probability that the true Sharpe ratio is below 0.\(^5\) Several studies in the practitioners and academic literature report similar results, which are conveniently justified with some ex-post explanation ("the posterior gives rise to a prior"). What this analysis misses is an evaluation of the probability that this backtest has been overfit to the data, which is the subject of Bailey et al. [1].

In this practical application we have illustrated how simple it is to produce overfit backtests when answering common investment questions, such as the presence of seasonal effects. We refer the reader to the appendix section "Reproducing the Results in Example 6" for the implementation of this experiment in the Python language. Similar experiments can be designed to demonstrate overfitting in the context of other effects, such as trend following, momentum, mean-reversion, event-driven effects, etc. Given the facility with which elevated Sharpe ratios can be manufactured is, the reader would be well advised to remain highly suspicious of backtests and of researchers who fail to report the number of trials attempted.

**Conclusions**

While the literature on regression overfitting is extensive, we believe that this is the first study to discuss the issue of overfitting on the subject of investment simulations (backtests) and its negative effect on OOS performance. On the subject of regression overfitting, the great Enrico Fermi once remarked (Mayer et al. [20]):

I remember my friend Johnny von Neumann used to say, with four parameters I can fit an elephant, and with five I can make him wiggle his trunk.

The same principle applies to backtesting, with some interesting peculiarities. We have shown that backtest overfitting is difficult indeed to avoid. Any perseverant researcher will always be able to find a backtest with a desired Sharpe ratio regardless of the sample length requested. Model complexity is only one way that backtest overfitting is facilitated. Given that most published backtests do not report the number of trials attempted, many of them may be overfitted. In that case, if an investor allocates capital to them, performance will vary: *It will be around zero if the process has no memory, but it may be significantly negative if the process has memory*. The standard warning that "past performance is not an indicator of future results” understates the risks associated with investing on overfit backtests. When financial advisors do not control for overfitting, positive backtested performance will often be followed by negative investment results.

We have derived the expected maximum Sharpe ratio as a function of the number of trials (N) and sample length. This has allowed us to determine the *Minimum Backtest length* (MinBTL) needed to avoid selecting a strategy with a given IS Sharpe ratio among N trials with an expected OOS Sharpe ratio of zero. Our conclusion is that the more trials a financial analyst executes, the greater should be the IS Sharpe ratio demanded by the potential investor.

We strongly suspect that such backtest overfitting is a large part of the reason why so many algorithmic or systematic hedge funds do not live up to the elevated expectations generated by their managers.

---

\(^5\)The Probabilistic Sharpe Ratio (orPSR) is an extension to the SR. Nonnormality increases the error of the variance estimator, and PSR takes that into consideration when determining whether an SR estimate is statistically significant. See Bailey and Lopez de Prado [2] for details.
We would feel sufficiently rewarded in our efforts if this paper succeeds in drawing the attention of the mathematical community to the widespread proliferation of journal publications, many of them claiming profitable investment strategies on the sole basis of in-sample performance. This is understandable in business circles, but a higher standard is and should be expected from an academic forum.

A depressing parallel can be drawn between today’s financial academic research and the situation denounced by economist and Nobel Laureate Wassily Leontief writing in Science (see Leontief [16]):

A dismal performance

‘... “What economists revealed most clearly was the extent to which their profession lags intellectually.” This editorial comment by the leading economic weekly (on the 1981 annual proceedings of the American Economic Association) says, essentially, that the “king is naked.” But no one taking part in the elaborate and solemn procession of contemporary U.S. academic economics seems to know it, and those who do don’t dare speak up.’

[...]

[E]conomisticians fit algebraic functions of all possible shapes to essentially the same sets of data without being able to advance, in any perceptible way, a systematic understanding of the structure and the operations of a real economic system.

[...]

That state is likely to be maintained as long as tenured members of leading economics departments continue to exercise tight control over the training, promotion, and research activities of their younger faculty members and, by means of peer review, of the senior members as well.

We hope that our distinguished colleagues will follow this humble attempt with ever-deeper and more convincing analysis. We did not write this paper to settle a discussion. On the contrary, our wish is to ignite a dialogue among mathematicians and a reflection among investors and regulators. We should do well also to heed Newton’s comment after he lost heavily in the South Seas bubble; see [21]:

For those who had realized big losses or gains, the mania redistributed wealth. The largest honest fortune was made by Thomas Guy, a stationer turned philanthropist, who owned £54,000 of South Sea stock in April 1720 and sold it over the following six weeks for £234,000. Sir Isaac Newton, scientist, master of the mint, and a certifiably rational man, fared less well. He sold his £7,000 of stock in April for a profit of 100 percent. But something induced him to reenter the market at the top, and he lost £20,000. “I can calculate the motions of the heavenly bodies,” he said, “but not the madness of people.”

Appendices

Proof of Proposition 1

Embrechts et al. [5, pp. 138-147] show that the maximum value (or last order statistic) in a sample of independent random variables following an exponential distribution converges asymptotically to a Gumbel distribution. As a particular case, the Gumbel distribution covers the Maximum Domain of Attraction of the Gaussian distribution, and therefore it can be used to estimate the expected value of the maximum of several independent random Gaussian variables.

To see how, suppose there is a sample of IID random variables, $z_n \sim Z, n = 1, \ldots, N$, where $Z$ is the CDF of the Standard Normal distribution. To derive an approximation for the sample maximum, $\max_n = \max (z_n)$, we apply the Fisher-Tippett-Gnedenko theorem to the Gaussian distribution and obtain that

$$
\lim_{N \to \infty} \text{Prob} \left[ \frac{\max_n - \alpha}{\beta} \leq x \right] = G[x],
$$

where

- $G[x] = e^{-e^{-x}}$ is the CDF for the Standard Gumbel distribution.
- $\alpha = Z^{-1} \left[ 1 - \frac{1}{N^\gamma} \right], \beta = Z^{-1} \left[ 1 - \frac{1}{N^\gamma} \right] - \alpha$, and $Z^{-1}$ corresponds to the inverse of the Standard Normal's CDF.

The normalized constants $(\alpha, \beta)$ are derived in Resnick [22] and Embrechts et al. [5]. The limit of the expectation of the normalized maxima from a distribution in the Gumbel Maximum Domain of Attraction (see Proposition 2.1(iii) in Resnick [22]) is

$$
\lim_{N \to \infty} E \left[ \frac{\max_n - \alpha}{\beta} \right] = y,
$$

where $y$ is the Euler-Mascheroni constant, $y \approx 0.57721 56649 \ldots$. Hence, for $N$ sufficiently large, the mean of the sample maximum of standard normally distributed random variables can be approximated by

$$
E[\max_n] \approx \alpha + \gamma \beta = (1 - y)Z^{-1} \left[ 1 - \frac{1}{N^\gamma} \right] + yZ^{-1} \left[ 1 - \frac{1}{N^\gamma} \right],
$$

where $N > 1$. 

MAY 2014
NOTICES OF THE AMS
469
Proof of Proposition 3

Suppose there are two random samples (A and B) of the same process \( \{A_\tau, \tau\} \), where A and B are of equal size and have means and standard deviations \( \mu^A, \mu^B, \sigma^A, \sigma^B \). A fraction \( \epsilon \) of each sample is called IS, and the remainder is called OOS, where for simplicity we have assumed that \( \sigma^A_{\epsilon} = \sigma^B_{\epsilon} = \sigma^A_{\epsilon} = \sigma^B_{\epsilon} \). We would like to understand the implications of a global constraint \( \mu^A = \mu^B \).

First, we note that \( \mu^A = \delta \mu^A_{\epsilon} + (1 - \delta) \mu^A_{\epsilon} \) and \( \mu^B = \epsilon \mu^A_{\epsilon} + (1 - \epsilon) \mu^B_{\epsilon} \). Then, \( \mu^A_{\epsilon} > \mu^B_{\epsilon} \Longleftrightarrow \mu^A > \mu^B \Longleftrightarrow \mu^A_{\epsilon} > \mu^B_{\epsilon} \Longleftrightarrow (1 - \epsilon) \mu^A + \epsilon \mu^B \Longleftrightarrow (1 - \epsilon) \mu^A + \epsilon \mu^B \Longleftrightarrow \mu^A_{\epsilon} > \mu^B_{\epsilon} \).

Second, because of the global constraint \( \mu^A = \mu^B \), \( \mu^A_{\epsilon} = \mu^B_{\epsilon} \) and \( \mu^A_{\epsilon} > \mu^B_{\epsilon} \Longleftrightarrow \mu^A_{\epsilon} > \mu^B_{\epsilon} \Longleftrightarrow \mu^A_{\epsilon} > \mu^B_{\epsilon} \Longleftrightarrow \mu^A_{\epsilon} > \mu^B_{\epsilon} \Longleftrightarrow \mu^A_{\epsilon} > \mu^B_{\epsilon} \Longleftrightarrow \mu^A_{\epsilon} > \mu^B_{\epsilon} \).

We can divide this expression by \( \sigma^A_{\epsilon} > 0 \), with the implication that

\[
\text{SR}^A > \text{SR}^B_{\epsilon} \Longleftrightarrow \text{SR}^A_{\epsilon} > \text{SR}^B_{\epsilon}
\]

where we have denoted \( \text{SR}^A_{\epsilon} = \frac{\mu^A_{\epsilon}}{\sigma^A_{\epsilon}} \) etc. Note that we did not have to assume that \( A_{\epsilon} \) is IID, thanks to our assumption of equal standard deviations. The same conclusion can be reached without assuming equality of standard deviations; however, the proof would be longer but no more revealing (the point of this proposition is the implication of global constraints).

Proof of Proposition 4

This proposition compiles the half-life of a first-order autoregressive process. Suppose there is a random variable \( m_{\tau} \), that takes values of a sequence of observations \( \tau \in \{1, \ldots, T\} \), where

\[
m_{\tau} = (1 - \phi) \mu + \phi m_{\tau-1} + \sigma \varepsilon_t
\]

such that the random shocks are IID distributed as \( \varepsilon_t \sim N(0, 1) \). Then

\[
\lim_{\tau \to \infty} E_0[m_{\tau}] = \mu
\]

if and only if \( \phi \not\in (-1, 1) \). In particular, from Bailey and Lopez de Prado [3] we know that the expected value of this process at a particular observation \( \tau \) is

\[
E_0[m_{\tau}] = \mu (1 - \phi^\tau) + \phi^\tau \mu_0.
\]

Suppose that the process is initialized or reset at some value \( m_0 \neq \mu \). We ask the question, how many observations must pass before

\[
E_0[m_{\tau}] = \frac{\mu + m_0}{2}?
\]

Inserting (20) into (19) and solving for \( \tau \), we obtain

\[
\tau = \frac{\ln[2]}{\ln[\phi]},
\]

which implies the additional constraint that \( \phi \in (0, 1) \).

Proof of Proposition 5

Suppose that we draw two samples (A and B) of a first-order autoregressive process and generate two subsamples of each. The first subsample is called IS and is comprised of \( \tau = 1, \ldots, T \), and the second subsample is called OOS and is comprised of \( \tau = T + 1, \ldots, T \), with \( \epsilon \in (0, 1) \) and T an integer multiple of \( \epsilon \). For simplicity, let us assume that \( \sigma^A_{\epsilon} = \sigma^B_{\epsilon} = \sigma^A_{\epsilon} = \sigma^B_{\epsilon} \). From Proposition 4, (18), we obtain

\[
E_0[m_{T}] = m_{T} \Rightarrow (1 - \phi)(1 - \phi^T) = (1 - \phi)(1 - \phi^T),
\]

Because \( 1 - \phi \epsilon > 0 \), \( \sigma^A_{\epsilon} = \sigma^B_{\epsilon}, \text{SR}^B_{\epsilon} > \text{SR}^A_{\epsilon} = m_{T}^\prime \Rightarrow m_{T} > m_{T}^\prime \). This means that the OOS of A begins with a seed that is greater than the seed that initializes the OOS of B. Therefore, \( m_{T} > m_{T}^\prime \Rightarrow E_0[m_{T}] > m_{T}^\prime \Rightarrow E_0[m_{T}] > m_{T}^\prime \). Because \( \sigma^A_{\epsilon} = \sigma^B_{\epsilon} \), we conclude that

\[
\text{SR}^B > \text{SR}^A \Rightarrow \text{SR}^B_{\epsilon} < \text{SR}^A_{\epsilon} < \text{SR}^B_{\epsilon} < \text{SR}^A_{\epsilon}
\]

Reproducing the Results in Example 6

Python code implementing the experiment described in “A Practical Application” can be found at http://www.quantresearch.info/Software.htm and at http://www.financial-math.org/software/.

Acknowledgments

We are indebted to the editor and two anonymous referees who peer-reviewed this article for the Notices of the American Mathematical Society. We are also grateful to Tony Anagnostakis (Moore Capital), Marco Avellaneda (Courant Institute, NYU), Peter Carr (Morgan Stanley, NYU), Paul Embrechts (ETH Zurich), Matthew D. Foreman (University of California, Irvine), Jeffrey Lange (Guggenheim Partners), Attilio Meucci (KKR, NYU), Natalia Nolde (University of British Columbia ad ETH Zurich), and Riccardo Rebonato (PIMCO, University of Oxford). The opinions expressed in this article are the authors’ and they do not necessarily reflect the views of the Lawrence Berkeley National Laboratory, Guggenheim Partners, or any other organization. No particular investment or course of action is recommended.

References


STOP-OUTS UNDER SERIAL CORRELATION AND "THE TRIPLE PENANCE RULE"

David H. Bailey
Marcos Lopez de Prado

First version: December 2012
This version: October 2014

David H. Bailey is recently retired from the Lawrence Berkeley National Laboratory and is a Research Fellow at University of California, Davis, Department of Computer Science, Davis, CA 95616, USA. E-mail: david@davidbailey.com. URL: www.davidhbailey.com.

Marcos Lopez de Prado is Senior Managing Director at Guggenheim Partners, New York, NY 10017, USA, and Research Affiliate at Lawrence Berkeley National Laboratory, Berkeley, CA 94720, USA. E-mail: lopzeprado@lbj.gov. URL: www.OuaiffResearch.info.

We are grateful to the Editor-in-Chief and two anonymous referees for their suggestions. We would also like to thank Tony Anagnostakis (Moore Capital), Jose Blanco (UBS), Peter Carr (Morgan Stanley, NYU), Matthew D. Foreman (University of California, Irvine), Marco Dion (J.P. Morgan), Jeffrey S. Lange (Guggenheim Partners), David Leinweber (CIFT, Lawrence Berkeley National Laboratory), Attilio Meucci (KKR, NYU), Riccardo Rebonato (PIMCO, University of Oxford) and Luis Viceira (HBS).

Supported in part by the Director, Office of Computational and Technology Research, Division of Mathematical, Information, and Computational Sciences of the U.S. Department of Energy, under contract number DE-AC02-05CH11231.

The opinions expressed in this study are the authors' and they do not necessarily reflect the views of the institutions they are affiliated with.
ABSTRACT

At what loss should a portfolio manager be stopped-out? What is an acceptable time under water? We demonstrate that, under standard portfolio theory assumptions, the answer to the latter question is strikingly unequivocal: On average, the recovery spans three times the period involved in accumulating the maximum quantile loss for a given confidence level. We denote this principle the "triple penance rule".

We provide a theoretical justification to why investment firms typically set less strict stop-out rules to portfolio managers with higher Sharpe ratios, despite the fact that they should be expected to deliver superior performance. We generalize this framework to the case of first-order auto-correlated investment outcomes, and conclude that ignoring the effect of serial correlation leads to a gross underestimation of the downside potential of hedge fund strategies, by as much as 70%. We also estimate that some hedge funds may be firing more than three times the number of skillful portfolio managers, compared to the number that they were willing to accept, as a result of evaluating their performance through traditional metrics, such as the Sharpe ratio.

We believe that our closed-form compact expression for the estimation of downside potential, without having to assume IID cashflows, will open new practical applications in risk management, portfolio optimization and capital allocation. The Python code included confirms the accuracy of our analytical solution.

Keywords: Downside, time under water, stop-out, triple penance, serial correlation, Sharpe ratio.

AMS Classification: 91G10, 91G60, 91G70, 62C, 60E.
JEL Classification: GO, GI, G2, G15, G24, E44.
1.- INTRODUCTION

Multi-manager investment firms are routinely faced with the decision to stop-out a portfolio manager (PM). This is a decision of the utmost importance, intended to protect the well-being of the overall funds. It typically has dramatic consequences, including the removal of the PM involved. Despite of the relevance and recurrence of stop-outs, we are not aware of the existence of a theoretical framework addressing this particular question. Such framework would be very useful in practice, as it would allow the firm to approach the stop-out problem in an objective and transparent manner, thus avoiding the personal conflict that is so often associated with employee dismissals.

The question of when do we possess enough evidence to discontinue (or stop-out) an investment strategy can be approached as a decision problem under uncertainty. This uncertainty arises from the fact that we cannot be sure whether negative outcomes, accumulated over time, are the result of bad performance, or mere bad luck. We can use decision theory to construct at test that, given a significance level \( \alpha \), rejects or not the null hypothesis that a PM's performance is consistent with skill. In this paper we introduce a framework to test for that hypothesis in three alternative formulations: The maximum quantile-loss \( (\text{MaxQL}) \), the quantile-time under water \( (\text{TuW}) \), and a procedure to translate realized losses into implied time under water \( (\Pi \text{TuW}) \).

Consider a multi-manager firm, whose executives must decide where and when a particular PM must be stopped-out.\(^1\) Multi-manager firms allocate capital to a PM based on the statistics that characterize his track record. These statistics are then used to generate MaxQL or TuW limits, which reflect the firm's appetite for false positives. Very tight limits are adopted by firms who are willing to fire a truly skillful PM, however unlucky he may have been. In other words, realized MaxQL and TuW exceeding certain limits are taken as sufficient evidence (for a certain confidence level) that the PM is not living up to the expectations which granted him a capital allocation.

We begin our discussion with the standard mean-variance framework introduced by the seminal work of Markowitz [1952, 1956, 1959]. Under the assumption of IID Normal outcomes, we determine the MaxQL and the TuW associated with a particular confidence level. With these results as a backdrop, we generalize our framework to incorporate the possibility of first-order autoregressive (or AR(1)) investment outcomes. Allowing this type of serial conditionality offers a richer analysis than standard mean-variance approaches. Higher-order serial conditionality would lead to different numerical results, however not conceptually different conclusions than those derived from an AR(1) specification. This is because the key feature that leads to substantial MaxQL and TuW is serial-dependence, which is already incorporated by an AR(1) process. To our knowledge, this is the first time that a compact expression is published which provides analytical estimates of quantile-loss potentials under first-order autoregressive cashflows.

A first goal of this paper is to provide an analytical framework to the stop-out problem, allowing for serial-conditionality of outcomes. We show through Monte Carlo

\(^1\) Our analysis can be applied to managers as well as strategies, and so we will refer to one or the other interchangeably.
experiments that the closed-form compact solution\(^2\) presented here is accurate. In providing an analytical estimate of downside potential, even in the presence of serial-conditionality, we open the possibility to integrate our results in optimization problems, rather than having to resort to computationally-expensive numerical methods. A second goal is to provide a theoretical justification as to why PMs with high Sharpe ratios may be given more permissive stop-out limits. A third goal is to formalize the relationship between the two main financial variables involved in a stop-out, MaxQL and TuW, by deriving the ITuW. We believe that ITuW is a more effective way to communicate stop-out limits.

The rest of the study is organized as follows: Section 2 reviews the academic literature on the topic. Section 3 introduces the framework. Section 4 determines the maximum quantile-loss over any horizon. Section 5 determines the time under water for a given confidence level, and the implied time under water for a given loss. Section 6 combines both concepts (maximum quantile-loss and quantile-time under water) into the "triple penance rule." Section 7 presents a numerical example. Section 8 explains why PMs with higher Sharpe ratios tend to receive less strict stop-out limits. Section 9 generalizes our framework to the case of first-order auto-correlated cashflows. Section 10 applies our framework to a long series of Hedge Fund Research (HFR) indices, and evaluates the impact that auto-correlation has on hedge funds' downside potential and firing practices. Section 11 summarizes our conclusions. The mathematical appendices prove the propositions presented throughout the paper. The Python code that numerically validates the accuracy of our analytical results can be found at www.QuamResearch.info/downloads/DF_AppelKuces.pdf.

2.- LITERATURE REVIEW
Lopez de Prado and Peijan [2004] showed that serial correlation is the main feature responsible for large MaxQL and TuW outcomes, even more so than Non-Normality. Non-Normality is a lesser concern because, as long as investment outcomes are independent and identically distributed (IID), the Central Limit Theorem ensures that the cumulative distribution of those investment outcomes converges to a Normal distribution as time passes. That paper allowed for serial-conditionality when modeling MaxQL and TuW, however those values had to be computed through a Monte Carlo.

This is not the first paper to discuss the impact that serial dependence has on performance metrics or quantile-losses. Lo [2002] derived the analytical solution to the asymptotic distribution of the Sharpe ratio under serial correlation, and found that the annual Sharpe ratio for a hedge fund can be overstated by as much as 65% because of the presence of serial correlation in monthly returns. In a very interesting study, Hayes [2006] fits a Markov chain (switching) model to hedge fund returns to estimate their downside potential under that type of serial dependence. Our work differs from Hayes’ in a number of ways. We use an autoregressive specification, which allows for Gaussian (continuous)

\(^2\) We use the term closed-form solution in the sense of being exact and analytically derived. With the term compact, we mean that this solution does not involve mathematical operators such as discrete summation, and therefore it is amenable to Analysis.
shocks, while the Markov chain only considers a fixed step (up or down). Empirical evidence supports the autoregressive nature of hedge fund returns’ serial dependence. For example, Getmansky et al. [2004] study various sources for hedge fund returns’ serial correlation and conclude that exposure to illiquid investments is the most likely. It seems implausible that the returns of illiquid investments will follow the conditioned fixed step implied by a Markov chain. In his study, Hayes argues that a Markov chain asymptotically approximates to an AR(1) model, however he also acknowledges that the speed of convergence is slow, and both models will yield different answers. This is a problem, because we could not rely on a downside risk model which is only accurate after an extended time under water. The conclusion is that it would be desirable to develop a downside risk model with serial correlation following an autoregressive specification. The present paper addresses this gap in the literature.

The academic literature measures downside potential according to at least three distinct approaches: (i) As an extreme value (or drawdown), like in Grossman and Zhou [1993], Magdon-Ismail and Atiya [2004], Carr et al. [2011], Yang and Zhong [2012] or Zhang et al. [2013]; (ii) As a quantile loss (an analogue to VaR), like in Lopez de Prado and Peijan [2004], Mendes and Leaf [2005] or Hayes [2006]; and (iii) As the average of a specified percentage of the largest losses over an investment horizon (an analogue to CVaR), like in Chekhlov, Uryasev and Zabarankin [2003, 2005], or Pavlikov, Uryasev and Zabarankin [2012]. We are interested in computing stop-out limits consistent with the testing of a hypothesis, thus the first approach (extreme loss or drawdown) is not useful to our problem. Of the other two approaches, we have opted for the second one because of the prevalence of VaR formulations among practitioners and regulators. Since January 1997, the U.S. Securities and Exchange Commission has required all publicly traded companies to disclose their financial risks. Value-at-Risk (VaR) is one of the three S.E.C. approved formats (Lin et al. [2010]). Practitioners and regulators’ preference for quantile-loss risk management methods is also evidenced by the Basel II Accords, published in June 2004 (Chen [2014]). While downside risk can be measured in many different ways, it is important to understand the practical implications of decision-making under the most widely used risk framework.

3.- THE FRAMEWORK
Suppose an investment strategy which yields a sequence of cash inflows $\Delta \pi_\tau$ as a result of a sequence of bets $\tau \in \{1, \ldots, \infty\},$ where

$$\Delta \pi_\tau = \mu + \sigma \varepsilon_\tau$$

such that the random shocks are IID distributed $\varepsilon_\tau \sim N(0,1).$ Because these random shocks $\varepsilon_\tau$ are independent and Normally distributed, so is the random variable $\Delta \pi_\tau,$ with $\Delta \pi_\tau \sim N(\mu, \sigma^2).$ This provides for negative as well as positive outcomes, although in the context of stop-outs we are naturally interested in the former rather than the latter. Throughout the remainder of the paper, we will assume that $\mu > 0.$ While Bailey and Lopez de Prado [2013] demonstrate that it may be optimal to allocate capital to strategies
with \( \mu \leq 0 \) as long as their correlation to the overall portfolio is sufficiently low, that is not a scenario most practitioners would consider.

Downside periods arise from the accumulation of cash inflows \( \Delta \pi_j \) over \( t \) sequential bets (or equivalently, a period of length \( t \)). If these bets are taken with a certain regular frequency, \( t \) can also be interpreted in terms of time elapsed. For instance, twelve monthly bets would span a period of one calendar year. Let us define a function \( \pi_t \) that accumulates the cashflows \( \Delta \pi_j \) over \( t \) bets.

\[
\pi_t = \sum_{\tau=1}^{t} \Delta \pi_{\tau}
\]

where \( t \in \{0, 1, \ldots, \infty\} \) and \( \pi_0 = 0 \). At the origin of the investment cycle, the cumulative performance is set to zero, because we would like to evaluate the downside potential of \( \pi_t \) following that reset point \( t=0 \). Because \( \pi_t \) is the aggregation of \( t \) IID random variables \( \Delta \pi_t \sim N(\mu, \sigma^2) \), we know that \( \pi_t \sim N(\mu t, \sigma^2 t) \).

For a significance level \( \alpha < \frac{1}{2} \), we define the quantile function for \( \pi_t \):

\[
Q_{\alpha,t} = \mu t + Z_\alpha \sigma \sqrt{t}
\]

where \( Z_\alpha \) is the critical value of the Standard Normal distribution associated with a probability \( \alpha \) of performing worse than \( Q_{\alpha,t} \), i.e. \( \alpha = Prob[n_t \leq Q_{\alpha,t}] \). Then, the maximum quantile-loss is defined as:

\[
QL_{\alpha,t} = \max\{0, -Q_{\alpha,t}\}
\]

Note that \( QL_{\alpha,1} \) coincides with the standard Value-at-Risk (VaR) for that investment at a \((1 - \alpha)\) confidence level (see Jorion [2006] for a discussion of VaR), which is a deterministic function of \( \{\mu, \sigma, \alpha, t\} \).

**4. MAXIMUM QUANTILE LOSS**

VaR is a risk metric limited to a particular horizon, typically one step ahead. We will move beyond VaR by determining the maximum quantile-loss regardless of the number of bets (or time horizon) involved. In words, we would like to answer the question: Up to how much could a particular strategy lose with a given confidence level? Proposition 1 computes analytically the maximum quantile-loss for a given significance level (proved in the Appendix).

**DEFINITION 1:** Given a significance level \( \alpha \), we define maximum quantile-loss of an investment as \( \text{MaxQL}_\alpha = \max_t [QL_{\alpha,t}] = \max\{0, -\min_t [Q_{\alpha,t}]\} \).
**PROPOSITION 1**: Assuming IID cashflows $\Lambda \pi_t \sim N(\mu, \sigma^2)$, and $\mu > 0$, the maximum quantile-loss associated with a significance level $\alpha < \frac{1}{2}$ is

$$\text{MaxQL}_\alpha = \frac{(Z_{\alpha}\sigma)^2}{4\mu}$$

(5)

which occurs at the time (or bet)

$$t^*_\alpha = \left(\frac{Z_{\alpha}\sigma}{2\mu}\right)^2$$

(6)

5.- **QUANTILE TIME UNDER WATER**

PMs are also routinely stopped-out if they do not recover from a loss after a period of time. In this section we determine that period for a given significance level.

**DEFINITION 2**: Given a significance level $\alpha$, we define quantile-time under water of an investment as the minimum time $\text{TuW}_\alpha$ with $\text{TuW}_\alpha > 0$, such that $Q_{\alpha, \text{TuW}_\alpha} = 0$.

Proposition 2 computes analytically the quantile-time under water for a given significance level (proved in the Appendix).

**PROPOSITION 2**: Assuming IID cashflows $\Lambda \pi_t \sim N(\mu, \sigma^2)$, and $\mu > 0$, the quantile-time under water associated with a significance level $\alpha < \frac{1}{2}$ is

$$\text{TuW}_\alpha = \left(\frac{Z_{\alpha}\sigma}{\mu}\right)^2$$

(7)

Suppose that a PM experiences a performance $\tilde{\pi}_t < 0$ after $t$ observations. For how long may we not be able to receive performance fee due to $\tilde{\pi}_t$? $\tilde{\pi}_t$ is a realized performance, which is consistent with a quantile loss for some confidence level $\tilde{\alpha}$, $-Q_{\tilde{\alpha}}$. It would be useful to translate that loss $\tilde{\pi}_t$ in terms of time under water, because that would allow us to express that monetary loss as a cost of opportunity (lost performance fee). Proposition 3 computes what is the time under water implied by $\tilde{\pi}_t$ (proved in the Appendix).

**PROPOSITION 3**: Given a realized performance $\tilde{\pi}_t < 0$ and assuming $\mu > 0$, the implied time under water is

$$\text{ITuW}_\tilde{\pi}_t = \frac{\tilde{\pi}_t^2}{\mu^2} - \frac{2\tilde{\pi}_t}{\mu} + t$$

(8)

Proposition 3 is useful because, given that $\tilde{\pi}_t$ has occurred, $\text{MaxTuW}^\wedge$ has become a realistic scenario of time under water. If, for example, $\tilde{\pi}_t$ is so negative that $\text{ITuW}_\tilde{\pi}_t >$
TuW_a, the firm has a strong argument to stop-out the strategy, even if \( \bar{M} \).\n\( \bar{M} \) Eq. (8) makes another key point: It not only matters how much money a PM has lost, but critically, for how long.

As we will see in Section 7, this Implied Time under Water (ITuW) is a better way of communicating stop-outs than giving a MaxQL_a limit, because it allows us to enforce limits at all times, even before hitting the maximum admissible loss or exhausting the time under water limit. Moreover, note that the calculation of TuW_a only requires three input variables \((\bar{M}, T, \mu)\), where the first two are directly observable. Unlike in the case of TuW_a, there is no need to input \( \sigma \).

6.- THE TRIPLE PENANCE RULE
The concepts of maximum quantile-loss \( (\text{MaxQL}_a) \) and quantile-time under water \( (\text{TuW}_a) \) are closely related. This is formalized in the following theorem (proved in the Appendix).

**THEOREM I for "triple penance rule"): Under standard portfolio theory assumptions, a strategy's maximum quantile-loss \( \text{MaxQL}_a \) for a significance level \( a \) occurs after \( t^*_a \) observations. Then, the strategy is expected to remain under water for an additional \( 3t^*_a \) after the maximum quantile-loss, with a confidence \( 1 - \alpha \).

If we define \( \text{Penance} = \frac{t^*_{\text{TuW}}}{\bar{M}_a} - 1 \), then the "triple penance rule" tells us that, assuming independent \( \bar{M}_a \) identically distributed as Normal (which is the standard portfolio theory assumption), \( \text{Penance} = 3 \), regardless of the Sharpe ratio of the strategy. In other words, it takes three times longer to recover from the maximum quantile-loss than the time it took to produce it, for a given significance level \( a < \frac{1}{2} \). This rule has important practical implications with regards to how long it will take for a PM to recover from a fresh new bottom. Figure 1 provides a graphical representation.

[FIGURE 1 HERE]

Should \( \bar{M}_a \) exhibit positive serial correlation, \( \text{MaxQL}_a, t^*_a \) and \( \text{TuW}_a \) will tend to be substantially greater than in the case of \( \bar{M}_a \) IID Normal, however \( \text{Penance} \) will tend to be smaller than 3. We will discuss this case in Sections 9 and 10.

7.- NUMERICAL EXAMPLE
Consider two portfolio managers: PM1 and PM2. PM1 is expected to make US$10m over a year, with an annual standard deviation of also US$10m and a monthly trading frequency. For simplicity, we will assume a risk-free rate of zero.\(^3\) This implies an annualized Sharpe ratio of 1 (see Sharpe [1975, 1994] for a formal definition). On the

\(^3\) We do this to simplify calculations, without loss of generality. Alternatively, the reader could think of these performance numbers as net of the risk-free rate.
other hand, PM2 will run the same risk budget as PM1, however he is expected to deliver a Sharpe ratio of 1.5. Table 1 summarizes this problem.

(TABLE 1 HERE)

For monthly bets, and a 95% confidence level, we would stop-out PM1 if he hits a cumulative loss of US$6,763,858.64, or stays under water for more than 2.706 years (about 33 months). Because PM2 is supposed to deliver a greater risk-adjusted performance (due to his Sharpe ratio of 1.5), he must trade under tighter constraints: We would stop-out PM2 if his losses exceed US$4,509,239.09, or if he remains under water longer than 1.2 years (about 15 months). Figure 2 plots the quantile-loss for a 95% confidence level, as a function of years passed, assuming monthly bets. In both cases, the time under water that follows the maximum quantile-loss is precisely 3 times the number of observations that occur up to the bottom performance. This is consistent with the "triple penance" rule.

(FIGURE 2 HERE)

Beyond stopping out in terms of maximum quantile-loss and time under water, this framework provides a basis for reassessing investments that perform worse than the quantile lines plotted in Figure 2 at any particular point in time. For example, suppose that PM1 has a cumulative loss of US$5,000,000 after being 2 years under water. Even though the loss is below the maximum quantile-loss, this scenario augurs a time under water of 3.125 years for the same confidence level implied by this observation (applying Proposition 3). That exceeds the pre-established limit of 2.706 years under water, and the firm may decide to stop-out PM1. Therefore, an effective way to communicate a downside limit is to translate a realized loss in terms of the implied time under water: Should $\text{ITuW}^n > \text{TuW}_q$, the strategy or PM will be stopped-out, regardless of the actual $\tilde{\pi}_t$, because the cost of opportunity (lost performance fee) is just too high for the firm.

8.- WHY DO BETTER MANAGERS GET LESS STRICT STOP-OUT LIMITS?
The previous numerical example gave a tighter stop-out limit to the PM with greater Sharpe ratio. And yet, the experienced reader is likely aware that hedge funds typically give greater stop-out limits to PMs with higher Sharpe ratios. To understand the reason for this apparent paradox, we need to incorporate into our model a bit of hedge fund business reality which is currently absent from our formal framework.

Hedge funds fund their operations through the collected management fee, and pay bonuses from the performance fee. Good PMs are likely to leave if they do not perceive a bonus within a certain timeframe. Hedge funds want to minimize the probability of defections among good PMs, who may abandon the firm leaving a loss behind. Thus, firms are willing to give more permissive stop-out limits to higher Sharpe ratio PMs.

Let us see how this argument fits in our framework. We can rewrite Eq. (6) as
\[ t^*_a = T \left( \frac{Z_a}{2SR} \right)^2 \]  

where \( T \) is the total number of independent bets implemented in a year, and \( SR \) is the annualized Sharpe ratio. Combining Eq. (9) in Eq. (7), we obtain that

\[ \frac{T \mu W_a}{T} = \left( \frac{Z_a}{SR} \right)^2 \]  

\( \frac{T \mu W_a}{T} \) is the quantile-time under water, for a confidence level \((1 - \alpha)\), expressed in years. However, it is known that many firms fix stop-out limits to a constant value for all PMs, \( \frac{T \mu W_a}{T} = K \). They do so because they fear that stopping out a star portfolio manager on the basis of time reduces her chances to recover, and may trigger her resignation. This sets a floor to the value that \( K \) may adopt. Values of \( K \) are also capped, because firms are aware that a good PM will leave if she does not receive frequent bonuses. What is effect of setting a constant time stop-out for all PMs regardless of their Sharpe ratio? Under these circumstances, Eq. (10) leads to

\[ K = \left( \frac{Z_a}{SR} \right)^2 \]  

Eq. (11) evidences the existence of a trade-off between greater Sharpe ratio and greater tolerance to downside potential. For double SR, a double \( Z_a \) is admissible, which allows for a substantially lower value of \( \alpha \) (recall that \( \alpha < \frac{1}{2} \), and thus \( Z_a < 0 \)). More precisely, the significance level admissible, subject to an exogenously set target \( K \), is

\[ \bar{\alpha} = Z \left[ -SR\sqrt{K} \right] \]  

Once again, the negative sign appears due to the fact that \( \alpha < \frac{1}{2} \), Eq. (12) gives us a nice expression, which incorporates the business reality we discussed at the beginning of this section. It explicitly tells us that a hedge fund is more permissive with PMs with high Sharpe ratios, despite the fact that we should expect those same PMs to perform better and hence operate with lower downside potential.

Following our example from the previous section, \( \bar{\alpha} \) is the answer to the question: What significance level is consistent with setting a constant time stop-out \( K \)? Following the example presented in Section 7, for \( K = 1 \) (one year), PM1 would be stopped-out at a significance level \( \bar{\alpha}_1 = 0.1587 \), and PM2 would be stopped-out at a significance level \( \bar{\alpha}_2 = 0.0668 \). As Figure 3 shows, now we are imposing stricter stop-out limits on PM1 than on PM2, the opposite of what we saw in Section 7. This is how in fact hedge funds typically operate, constrained as they are by the reality of having to protect themselves against defections.
Similarly, for an exogenously set $K = 2$ (two years), PM1 would be stopped-out at a significance level $\bar{a}_1 = 0.0787$, and PM2 would be stopped-out at a significance level $\bar{a}_2 = 0.0169$. Figure 4 shows that in this case the hedge fund sets even more permissive stop-out limits on PM2 relative to PM1 than we saw in Figure 3.

This greater permissiveness towards PMs with higher Sharpe ratios is contrary to what standard portfolio theory would have predicted, and yet our framework shows that hedge funds operating in this way act rationally, in an attempt to minimize the risk of defection among their most talented portfolio managers.

9. STOP-OUT LIMITS UNDER FIRST-ORDER AUTO-CORRELATED CASHFLOWS

Suppose an investment strategy which yields a sequence of cash inflows $\Delta \pi_\tau$ as a result of a sequence of bets $\tau \in \{1, \ldots, \infty\}$, where

$$\Delta \pi_\tau = (1 - \phi) \mu + \varphi \Delta \pi_{\tau-1} + \sigma \varepsilon_\tau$$  \hspace{1cm} (13)

such that the random shocks are IID distributed $\varepsilon_\tau \sim N(0,1)$. Eq. (13) is initialized by a seed value $\Delta \pi_0$, which is not necessarily null. These random shocks $\varepsilon_\tau$ follow an independent and identically distributed Gaussian process, however $\Delta \pi_\tau$ is neither an independent nor an identically distributed process. This is due to the parameter $\phi$, which incorporates a first-order serial-correlation effect of auto-regressive form. Appendix 5 shows that a necessary and sufficient condition for $\Delta \pi_\tau$ to be stationary is that $\phi \in (-1,1)$. In that case, the above process has an asymptotic expectation $\lim_{\tau \to \infty} E_0[\Delta \pi_\tau] = \mu$, and an asymptotic variance $\lim_{\tau \to \infty} V_0[\Delta \pi_\tau] = \frac{\sigma^2}{1 - \phi^2}$, where the zero subscript at $E_0$ and $V_0$ denote expectations formed at the origin ($\tau = 0$). If these bets are taken with a certain regular frequency, $t$ can also be interpreted in terms of time elapsed. For instance, twelve monthly bets would span a period of one year.

Like in Section 3, let us define a function $\pi_t$ that accumulates the cash inflows $\Delta \pi_\tau$ over time.

$$\pi_t = \sum_{\tau=1}^{t} \Delta \pi_\tau$$  \hspace{1cm} (14)

where $t \in \{0, 1, \ldots, \infty\}$ and $\pi_0 = 0$ at the onset of the investment cycle, when the stop-out limits are set up. The following Proposition is proved in the Appendix.
PROPOSITION 4: Under the stationarity condition $\phi \in (-1,1)$, the distribution of a cumulative function $n$ of a first-order auto-correlated random variable $\Delta \pi_t$ follows a Normal distribution with parameters:

$$
\pi_t \sim N\left( \frac{\varphi^{t+1} - \varphi}{\varphi - 1} (\Delta \pi_0 - \mu) + \mu t, \frac{\sigma^2}{(\varphi - 1)^2} \left( \frac{\varphi^{2(t+1)} - 1}{\varphi^2 - 1} - 2 \frac{\varphi^{t+1} - 1}{\varphi - 1} + t + 1 \right) \right)
$$

(15)

For a significance level $\alpha < \frac{1}{2}$, we can estimate the quantile-loss function as the lower band for $\pi_t$ after $t$ bets:

$$
Q_{\alpha, t} = \frac{\varphi^{t+1} - \varphi}{\varphi - 1} (\Delta \pi_0 - \mu) + \mu t + Z_{\alpha} \frac{\sigma}{|\varphi - 1|} \left( \frac{\varphi^{2(t+1)} - 1}{\varphi^2 - 1} - 2 \frac{\varphi^{t+1} - 1}{\varphi - 1} + t + 1 \right)^{1/2}
$$

(16)

where $Z_{\alpha}$ is the critical value of the Standard Normal distribution associated with a probability $\alpha$ of performing worse than $Q_{\alpha, t}$. Like before, the quantile-loss function is finally obtained as $QL_{\alpha, t} = \max \{ 0, -Q_{\alpha, t} \}$.

We are not aware of previously published analytical estimates of quantile-losses under first-order auto-correlated cashflows. Proposition 4 is particularly useful in practice, because it gives us that closed-form solution presented in Eq. (16), and allows us to enunciate Proposition 5 (see the Appendix for a proof).

PROPOSITION 5: For $\mu > 0$, $Q_{\alpha, t}$ is unimodal, a global minimum exists ($\text{Min} Q_{\alpha}$) and $\text{Max} QL_{\alpha} = \max \{ 0, -\text{Min} Q_{\alpha} \}$ can be computed.

Appendices 9 and 10 present algorithms to determine the maximum quantile-loss and time under water in this more general framework. These procedures can be easily integrated in optimization problems, such as portfolio optimization subject to quantile-loss or time under water constraints under serial conditionality. This is relevant, because many times researchers are compelled to adopt the ubiquitous IID assumption solely for computational reasons, contrary to empirical evidence that would have advised them to apply an expression like Eq. (16).

10. DOWNSIDE POTENTIAL IN THE HEDGE FUND INDUSTRY

We are ready to put into practice the theory introduced in the earlier sections. We have downloaded from Bloomberg a long series of monthly Net Asset Values (NAVs) for Hedge Fund Research Indices (HFR), and selected those series that go from January 1st
1990 to January 1st 2013. This gives us 265 data points for each of the indices listed in Table 2.

**[TABLE 2 HERE]**

NAV$s do not follow a stationary process (see Meucci [2005] for a comprehensive discussion of this subject). In order to apply our framework, we first need to perform a logarithmic transformation on the NAV$s. The maximum likelihood estimator of $\phi$ is $\hat{\phi} = \frac{1}{o} \sum_{i=0}^{o} [\Delta \pi_{\tau}, \Delta \pi_{\tau-1}] / \sum_{i=0}^{o} [\Delta \pi_{\tau-1}, \Delta \pi_{\tau-1}]$, where $\Delta \pi_{\tau}$ is the series of first order differences on the log-NAVs at observation $\tau$, and $\text{Cov}_{\tau}$ is the covariance operator. The zero subscript denotes expectations formed at the origin ($\tau = 0$). Following Appendix 5, we estimate $\mu$ as $\hat{\mu} = \mu_\infty$, and $\sigma$ as $\hat{\sigma} = \sigma_\infty \sqrt{1 - \phi^2}$, where $\mu_\infty = \lim_{\tau \to \infty} E_0[\Delta \pi_{\tau}] = \mu$ is the asymptotic expected value and $\sigma_\infty^2 = \lim_{\tau \to \infty} V_0[\Delta \pi_{\tau}] = \sigma_o^2 \frac{1}{1 - \phi^2}$ is the asymptotic variance. $\mu_\infty$ and $\sigma_\infty^2$ can be approximated by the large-sample estimates of mean and variance of cashflows. Once we have estimated the triplet $(\hat{\mu}, \hat{\sigma}, \hat{\phi})$, we can compute the $\text{MaxQL}_a$ and $\text{TuW}_a$ using the code in Appendix 11.4 We are assuming $a = 0.05$ and $\Delta \pi_0 = 0$, but different scenarios can be simulated by changing the appropriate parameters in the code. Table 3 reports the results.

**[TABLE 3 HERE]**

As discussed in Appendix 8, the convergence of the procedure requires $\hat{\mu} > 0$ and $\hat{\phi} \in [0,1)$, which is consistent with 24 of the 26 cases listed above. As we can deduce from the t-Stat values for $\hat{\phi}$ reported in Table 3, $\phi$ is statistically significantly in 21 out of 26 cases at a 95% confidence level. Despite of this overwhelming empirical evidence, let us suppose that $\phi = 0$ in all of the above cases. This is of course a misleading assumption, however it is very common for practitioners and academics to assume that returns are independent to avoid dealing with serial correlation. Table 4 reports the corresponding $\text{MaxQL}_a$ and $\text{TuW}_a$ under that scenario.

**[TABLE 4 HERE]**

Results in Table 4 can be computed in two different ways: i) Running the code in Appendix 11 for $\phi = 0$ or ii) applying Eqs. (5) and (7). The conclusion is that unrealistically assuming $\phi = 0$ leads to a gross underestimation of the downside potential of hedge fund investments. For example, in the case of "HFRI RV: Fixed Income-Convertible Arbitrage Index" (Code "HFRICA1 Index"), the maximum quantile-loss under the assumption of independence is only 3.799, while considering first-order auto-correlation would yield 11.609. This means that wrongly assuming independence leads to a 67% underestimation compared to taking into account first-order auto-correlation. Quantile-time under water if we assume independence is only 21.28 (monthly) observations, while considering first-order auto-correlation would yield 74.42. This means that wrongly assuming independence leads to a 71% underestimation compared to

---

taking into account first-order auto-correlation. *Penance* measures how long it takes to recover from the maximum quantile-loss, as a multiple of the time it took to reach the bottom. More precisely, \( \text{Penance} = \frac{\max \{w_t \}}{t_a^*} - 1 \). Table 4 reports that *Penance* for hedge fund indices ranges between 1.6 and 3. Although positive serial correlation leads to greater downside potential, longer periods to reach the bottom \( t_a^* \) and longer periods under water, the *Penance* may be substantially smaller. In particular, *Penance* is smaller the higher \( \phi \) (Phi) and the higher the ratio \( \frac{\mu}{\sigma} \) (Mean divided by Sigma). Figure 5 plots *Penance* for hedge fund indices with various \( \phi \).

[FIGURE 5 HERE]

These results introduce two interesting implications: First, hedge fund strategies are much riskier than what could be derived from performance metrics that rely on the ubiquitous IID assumption, such as Sharpe ratio, Sortino ratio, Treynor ratio, Information ratio, etc. (see Bailey and Lopez de Prado [2012] for a discussion). This leads to an over-allocation of capital by Markowitz-style approaches to hedge fund strategies. Second, PMs and strategies evaluated by those IID-based metrics are being stopped-out much earlier than it would be appropriate. A good PM running a strategy that delivers auto-correlated cashflows may be unnecessarily stopped-out because the firm assumed IID cashflows. This is a particularly bad decision, because one positive aspect about strategies with auto-correlated cashflows is that their *Penance* is shorter than in the IID case.

We would like to understand whether hedge funds intending to accept a probability \( a_1 \) of firing a truly skillful portfolio manager (a "false positive") are effectively taking a different probability \( a_2 \) as a result of assuming returns independence. Combining Propositions 1 and 4 we can compute the \( a_2 \) associated with \( \pi_t = \text{MaxQL}_{\sigma_1} = -\frac{(z_{\Delta t^* \sigma} \sigma)^2}{4 \mu} \) as

\[
\alpha_2 = \Phi \left[ \frac{- (Z_{\Delta t^* \sigma})^2 - \frac{e^\sigma_1}{\varphi - 1} - \frac{\varphi (\Delta \pi_0 - \mu) + \mu t_{\alpha_1}^*}{(\varphi - 1)^2} - 2 \frac{e_{\sigma_1}^*}{\varphi - 1} + t_{\alpha_1}^* + 1}{\sqrt{\frac{\sigma^2}{(\varphi - 1)^2} (\frac{e^{2(\sigma_1^*)^2}}{\varphi^2 - 1} - 2 \frac{e^{\sigma_1^*}}{\varphi - 1} + t_{\alpha_1}^* + 1)}} \right]
\]  

(17)

where \( \Phi \) is the cdf of the standard Normal distribution.\(^5\) Table 5 reports the effective proportion \( \alpha_2 \) of truly skillful portfolio managers fired by hedge funds, despite of aiming at a proportion \( \alpha_1 = 0.05 \). Again, this discrepancy arises because hedge funds assuming returns independence aim at a proportion \( \alpha_1 \), however they effectively get a proportion \( \alpha_2 \) because that assumption was false in most cases (see t-Stat values for \( \phi \) in Table 3).

\(^5\) Incidentally, Eq. (17) can be used to compute ITuw (Proposition 3) in the more general framework of first-order serial correlation. In order to do that, we simply have to input this \( \alpha_2 \) in the algorithm described in Appendix 10. For the reasons argued in Section 5, this would be a more effective way to communicate stop-out limits.
For all hedge fund styles, \( \alpha_2 > \alpha_1 \), which means that they are effectively firing a greater proportion of truly skillful portfolio managers than they originally intended. That proportion of over-firings is \( \alpha_2 - \alpha_1 \). Most hedge funds evaluate performance through traditional metrics, such as the Sharpe ratio, which assumes returns independence and would lead to the over-firing reported in Table 5. For example, hedge funds similar to those in the "HFRI RV: Fixed Income-Convertible Arbitrage Index" (code "HFRCAI Index") may be firing 3.38 times (0.1688 vs. 0.05) the number of truly skillful portfolio managers, compared to the number they were willing to accept under the assumption of returns independence. Skillful managers that are fired by mistake need to be replaced in order to preserve performance, which increases personnel turnover. Our framework explains how the excessive turnover experienced by some hedge funds may be the result of unrealistically expecting their portfolio managers to deliver independent returns.

11.- CONCLUSIONS
Following standard portfolio theory assumptions, we have computed analytically the maximum quantile-loss and quantile-time under water for a certain confidence level. We have shown how these concepts are intimately related through the "triple penance" rule. This rule states that, under standard portfolio theory assumptions, it takes three times longer to recover from the expected maximum quantile-loss than the time it takes to produce it, with the same confidence level. We have introduced a new downside-risk concept called Penance, which measures how long it takes to recover from the maximum quantile-loss, as a multiple of the time it took to reach the bottom. We have also demonstrated an effective way to communicate downside limits, via an implied time under water formulation.

According to this framework, for a certain confidence level, we should expect tighter stop-out limits imposed on portfolio managers with higher Sharpe ratios. That is rarely the case in practice. We have provided a theoretical justification to this observation, by recognizing that hedge funds must confront the risk of defection. They know that, should a good portfolio manager not receive a bonus within a certain period of time, he may try his luck at another firm and leave a loss behind. The consequence is that hedge funds assign greater confidence levels to portfolio managers with greater Sharpe ratios, and as a result they are quicker in stopping out portfolio managers with lower Sharpe ratios, just the opposite to what standard portfolio theory would have predicted.

We have complemented our study with a generalization of our framework to deal with the case of first-order auto-correlated cashflows. We derived a closed-form compact expression which estimates the downside potential of a strategy without having to assume IID random shocks. An empirical study of hedge fund indices reveals that ignoring the effect of serial correlation leads to a gross underestimation of the downside potential of hedge fund strategies, by as much as 70%. Although positive auto-correlation leads to greater downside potential, longer periods to reach the bottom and longer periods under
water, the *Penance* may be substantially smaller. We find that some hedge funds may be firing more than three times the number of skilled portfolio managers, compared to the number that they were willing to accept, as a result of evaluating their performance through traditional metrics, such as the Sharpe ratio. The excessive turnover experienced by some hedge funds may be the result of unrealistically expecting their portfolio managers to deliver independent returns.

We are aware that researchers have been compelled to adopt the IID assumption in past, in disregard of contradicting empirical evidence, solely for computational reasons. We hope that the expression we have derived in this paper will allow them to take serial correlation into account in risk management, portfolio optimization and capital allocation applications. The Python code included in the Appendix numerically confirms the accuracy of our solution.\(^6\)

---

\(^6\) Electronic copy available at www.QuantResearch.info/downloads/DD_Appendices.pdf
APPENDICES

A.I.- PROOF TO PROPOSITION 1

The quantile-loss function is defined as $QL_{\alpha,t} = \max\{0, -Q_{\alpha,t}\}$. We can compute its maximum as $\text{Max}QL_{\alpha} = \max\{0, -\text{Min}Q_{\alpha}\}$, where $\text{Min}Q_{\alpha} = \min_t Q_{\alpha,t}$ is the minimum value over $t$ of the quantile function for a significance level $\alpha$. Next, we derive the expression for $\text{Min}Q_{\alpha}$. We have seen that, in the case of independent and identically distributed Normal cashflows $\pi_t \sim N(\mu, \sigma^2)$, the quantile function is given by the expression:

$$Q_{\alpha,t} = \mu t + Z_{\alpha} \sigma \sqrt{t}$$

(18)

The maximum quantile-loss is $\text{Max}QL_{\alpha} = \max\{0, -\text{Min}Q_{\alpha}\}$. Differentiating Eq. (18), the first order necessary condition that determines the globally and unconstrained minimum value for $Q_{\alpha,t}$ is given by:

$$\frac{\partial Q_{\alpha,t}}{\partial t} = \mu + \frac{1}{2\sqrt{t}}Z_{\alpha} \sigma = 0$$

(19)

$$\frac{1}{2\sqrt{t}}Z_{\alpha} \sigma < 0$$

because $\alpha < \frac{1}{2}$ implies $Z_{\alpha} < 0$, thus this first order condition requires $\mu > 0$. Solving for $t$, we obtain the number of observations at which the lowest value of the function $Q_{\alpha,t}$ is realized,

$$t^*_\alpha = \left(\frac{Z_{\alpha} \sigma}{2\mu}\right)^2$$

(20)

The second order sufficient condition is verified in $\frac{\partial^2 Q_{\alpha,t}}{\partial t^2} = -\frac{1}{4} Z_{\alpha} \sigma t^{-3/2} > 0$. This second derivative of the quantile function with respect to $t$ is strictly positive. This means that $Q_{\alpha,t}$ is convex with respect to $t$, which guarantees the existence of a global minimum.

Combining both conditions, we can then evaluate $Q_{\alpha,t}$ at that optimized value $t^*_\alpha$ to obtain its minimum value with a significance level $\alpha$:

$$\text{Min}Q_{\alpha} = Q_{\alpha,t^*_\alpha} = \mu \left(\frac{Z_{\alpha} \sigma}{2\mu}\right)^2 + Z_{\alpha} \sigma \left|\frac{Z_{\alpha} \sigma}{2\mu}\right| = -\frac{(Z_{\alpha} \sigma)^2}{4\mu}$$

(21)

As expected, $\text{Min}Q_{\alpha}$ is not a function of $t$. Its negative value appears because, as we saw, $Z_{\alpha} < 0$ and $\mu > 0$ are sufficient conditions for the global maximum quantile-loss to exist. ■

---

7 We will treat $t$ as a continuous variable in $\frac{3}{2}^+$ for the purpose of differentiation.
A.2.- PROOF TO PROPOSITION 2
The time under water (TuW) is the number of observations (in terms of bets or time), \( t > 0 \), that elapse until we first observe that \( \pi_{i-1} < 0 \) and \( \pi_i \geq 0 \). We can determine its upper boundary for a significance level \( \alpha < \frac{1}{2} \) as the value \( t > 0 \) such that \( Q_{\alpha, t} = Q_{L, a, t} = 0 \). This condition is satisfied at

\[
\mu t + Z_{\alpha} \sigma \sqrt{t} = t (\mu \sqrt{t} + Z_{\alpha} \sigma) = 0 \tag{22}
\]

and because \( t > 0 \Rightarrow \mu Z_t + Z_{\alpha} \sigma = 0 \), we obtain

\[
TuW_{\alpha} = \left( \frac{Z_{\alpha} \sigma}{\mu} \right)^2 \tag{23}
\]

A.3.- PROOF TO PROPOSITION 3
Suppose that a portfolio manager experiences a cumulative performance at time \( t \) for an amount \( \bar{f}_t \), where \( \bar{f}_t < \pi_0 = 0 \). We can determine the implied significance level, \( \bar{\alpha} \), that is associated with such a performance after \( t \) independent observations. More precisely, we would like to compute the value of \( \bar{\alpha} \) that verifies

\[
\bar{\alpha} = Q_{\bar{\alpha}, t}
\]

Applying Eq. (18) on Eq. (24), we can solve for \( \bar{\alpha} \) as follows

\[
\bar{\alpha} = \Phi \left[ \frac{\bar{\pi}_t - \mu \sqrt{t}}{\sigma \sqrt{t}} \right] \tag{25}
\]

where \( \Phi \) is the cumulative distribution function for the Standard Normal distribution. The symbol \( \Phi \) represents a function and should not be confounded with a critical value \( Z_{\alpha} \).

For the sake of clarity, note that \( \alpha = \Phi (Z_{\alpha}, V \alpha \in (0,1)) \).

Eq. (25) tells us that, for any observed performance, we can compute its implied statistical significance, which can be understood as \( \text{Prob}[\bar{\pi}_t < \bar{f}_t] \). Hence, \( \bar{\alpha} \) can be interpreted as the ex-ante probability that this strategy would have performed below \( \bar{f}_t \) after \( t \) observations. Inserting Eq. (25) in Eq. (23), we obtain

\[
ITuW^\wedge = \frac{\bar{\pi}_t^2}{\mu^2 t} - 2 \frac{\bar{\pi}_t}{\mu} + t \tag{26}
\]

Eq. (26) gives us the implied time under water associated with the realized performance \( \bar{f}_t \), where \( \bar{\alpha} \) is implied by that same realized performance, \( \bar{f}_t \). This proposition allows us to communicate stop-out limits more effectively than a mere \( \text{MaxQL}_{\alpha} \) limit, because we do not need to wait until the maximum quantile-loss or time under water is reached. It suffices that \( TuW^\wedge > TuW_{\bar{\alpha}} \) for any observed \( \bar{\pi}_t \).
A.4.- PROOF TO THEOREM 1 ("TRIPLE PENANCE RULE")
Comparing Eqs. (20) and (23), we derive the expression
\[
t^*_\alpha = \frac{1}{4} T u W^\alpha
\] (27)

Eq. (27) is true for any value \( \alpha \in \left(0, \frac{1}{2}\right) \). These boundaries for \( \alpha \) first appeared in Eq. (19). We call it "triple penance rule" because it tells us that, in the case of independent and identically distributed normal cashflows \( \mathbf{A}_\pi \), it takes three times longer to recover from the expected maximum quantile-loss than the time it took to produce it, for the same significance level \( \alpha < \frac{1}{2} \).  

A.5.- PROOF TO PROPOSITION 4
This proposition generalizes the framework discussed in Section 3, by deriving the distribution of a cumulative function of a first-order auto-correlated random variable. Suppose an investment strategy which yields a sequence of cash inflows \( \mathbf{A}_\pi \) as a result of a sequence of bets \( \tau \in \{1, ..., \infty\} \), where
\[
\mathbf{A}_\pi = (1 - \phi) \mu + \phi \mathbf{A}_{\pi_{\tau-1}} + \sigma \varepsilon
\] (28)
such that the random shocks are IID distributed as \( \varepsilon \sim N(0,1) \). Recursively replacing the previous expression \( j \) times leads to
\[
\Delta \pi = (1 - \phi) \mu \sum_{i=0}^{\tau-1} \phi^i + \phi^j \Delta \pi_{\tau-j} + \sigma \sum_{i=0}^{\tau-1} \phi^i \varepsilon_{\tau-i}
\] (29)

For a process initialized at \( \Delta \pi_0 \), we can extend the recursion back to the origin, in which case \( j = \tau \), and Eq. (29) becomes
\[
\Delta \pi = (1 - \phi) \mu \sum_{i=0}^{\tau-1} \phi^i + \phi^\tau \Delta \pi_0 + \sigma \sum_{i=0}^{\tau-1} \phi^i \varepsilon_{\tau-i}
\] (30)

This expression evidences that \( \Delta \pi \) is a linear function of independent Gaussian random variables, hence \( \mathbf{A}_\pi \) is also Gaussian (Grinstead and Snell [1997]). From Eq. (30), we can derive its mean and variance
\[
E_0 [\mathbf{A}_\pi] = (1 - \phi) \mu \sum_{i=0}^{\tau-1} \phi^i + \phi^\tau \Delta \pi_0
\] (31)
\[
\nu_0 [\mathbf{A}_\pi] = \sigma^2 \sum_{i=0}^{2\tau-1} \phi^i
\]
Eq. (31) shows that a necessary and sufficient condition for $A\pi$ to be stationary is that $\phi E\left(-1,1\right)$, in which case the above mean and variance asymptotically converge to
\[
\lim_{t \to \infty} \frac{E\left[A\pi_t\right]}{\mu} = \mu \quad \text{and} \quad \lim_{t \to \infty} V\left[\Delta\pi_t\right] = \frac{\sigma^2}{1-\phi^2}.
\]
From Eq. (31) we obtain
\[
A\pi \sim N\left(\left(I - \phi\right)\mu \sum_{i=0}^{\tau-1} \phi^i + \phi^\tau \Delta\pi_0, \sigma^2 \sum_{i=0}^{\tau-1} \phi^{2i}\right)
\]
Equation (32) shows that
\[
\pi_t = \sum_{\tau=1}^{t} \Delta\pi_t = \left(1 - \phi\right)\mu \sum_{\tau=1}^{t} \sum_{i=0}^{\tau-1} \phi^i + \Delta\pi_0 \sum_{\tau=1}^{t} \phi^\tau + \sigma \sum_{\tau=1}^{t} \sum_{i=0}^{\tau-1} \phi^i \varepsilon_{t-i}.
\]
The last term can be conveniently operated as
\[
E_0\left[\sigma \sum_{\tau=1}^{t} \sum_{i=0}^{\tau-1} \phi^i \varepsilon_{t-i}\right] = 0
\]
and
\[
V_0\left[\sigma \sum_{\tau=1}^{t} \sum_{i=0}^{\tau-1} \phi^i \varepsilon_{t-i}\right] = \sigma^2 V_0\left[\sum_{\tau=1}^{t} \left(\varepsilon_{t} \left(\sum_{i=0}^{\tau-1} \phi^i\right)\right)\right].
\]
We conclude that $\pi_t$ is also a linear function of independent Gaussian random variables. This means that $\pi_t$ is also Gaussian. We can compute the mean and variance of $\pi_t$ as:
\[
E_0[\pi_t] = \left(1 - \phi\right)\mu \sum_{\tau=1}^{t} \sum_{i=0}^{\tau-1} \phi^i + \Delta\pi_0 \sum_{\tau=1}^{t} \phi^\tau
\]
and
\[
V_0[\pi_t] = \sigma^2 \sum_{\tau=1}^{t} \left(\sum_{i=0}^{\tau-1} \phi^i\right)^2.
\]
Equation (35) addresses the first limitation we discussed earlier (these are the moments on $\pi_t$, not $\Delta\pi_t$), however it is still unsatisfactory with respect to the second feature: In order to
analyze the expressions in Eq. (35), we need to compute their compact form, i.e. excluding the summation operators that involve $t$ and $\tau$. Applying the Theorem of Geometric Series on $E_0[\pi, j]$, we obtain:

$$
\sum_{\tau=1}^{t} \sum_{i=0}^{\tau-1} \varphi^i = \frac{\varphi^\tau - 1}{\varphi - 1} = \frac{1}{\varphi - 1} \left( -t + \sum_{\tau=1}^{t} \varphi^\tau \right) = \frac{1}{\varphi - 1} \left( -t + \frac{\varphi^{t+1} - 1}{\varphi - 1} - 1 \right) = \frac{\varphi^{t+1} - 1}{(\varphi - 1)^2} - \frac{t + 1}{\varphi - 1}
$$

(36)

Thus,

$$
E_0[\pi, j] = \frac{\varphi^{t+1} - \phi}{\varphi} - \frac{\phi}{t} (\Delta\pi_0 - \mu) + \mu t
$$

(37)

This new expression for $E_0[\pi, j]$ is easier to deal with from an Analysis perspective. Likewise, we can reduce $V_0[\pi, j]$ by recognizing that

$$
\left( \sum_{i=0}^{\tau-1} \varphi^i \right)^2 = \frac{\varphi^{2(t-\tau+1)} - 2\varphi^{t-\tau+1} + 1}{(\varphi - 1)^2}
$$

$$
\sum_{\tau=1}^{t} \left( \sum_{i=0}^{\tau-1} \varphi^i \right)^2 = \frac{1}{(\varphi - 1)^2} \left( \sum_{\tau=1}^{t} \varphi^{2(t-\tau+1)} - 2\sum_{\tau=1}^{t} \varphi^{t-\tau+1} + t \right)
$$

(38)

Thus,

$$
V_0[\pi, t] = \frac{\varepsilon^2}{(\varphi - 1)^2} \left( \frac{\varphi^{2(t+\mu)} - 1}{\varphi^2 - 1} - 2\frac{\varphi^{t+1} - 1}{\varphi - 1} + t + 1 \right)
$$

(39)

Finally, we conclude that
\[ \pi_t \sim N \left( \frac{\varphi^{t+1} - \varphi}{\varphi - 1} (\Delta \pi_0 - \mu) \right) \]

\[ + \mu_t, \frac{\sigma^2}{(\varphi - 1)} \left( \frac{\varphi^{2(t+1)} - 1}{\varphi^2 - 1} - 2 \frac{\varphi^{t+1} - 1}{\varphi - 1} + t + 1 \right) \]

Eq. (40) has the two features we were looking for: First, it will allow us to compute the quantile-loss of a strategy with performance \( \pi_t \). Second, we have been able to express this result in a compact form, amenable to Analysis. \( \blacksquare \)

**A.6.- PROOF TO PROPOSITION 5**

We can use the analytical result obtained in Proposition 4 to study the behavior of the quantile-loss and maximum quantile-loss functions in the more general case of first-order auto-correlated cashflows. The quantile function in this case is

\[ Q_{\alpha,t} = \frac{\varphi^{t+1} - \varphi}{\varphi - 1} (\Delta \pi_0 - \mu) + \mu_t \]

\[ + Z_\alpha \frac{\sigma}{|\varphi - 1|} \left( \frac{\varphi^{2(t+1)} - 1}{\varphi^2 - 1} - 2 \frac{\varphi^{t+1} - 1}{\varphi - 1} + t + 1 \right)^{1/2} \]

where \( Z_{\alpha} \) is the critical value of the Standard Normal distribution associated with a probability \( \alpha \) of performing worse than \( Q_{\alpha,t} \), i.e. \( \alpha = \text{Prob} \left[ r_{t+1} \leq Q_{\alpha,t} \right] \). Differentiating \( Q_{\alpha,t} \) with respect to \( t \):

\[ \frac{d^i Q_{\alpha,t}}{dt^i} = \frac{d^i E_0[\pi, t]}{dt^i} + Z_{\alpha} \frac{d^i V_{\alpha}[\pi, t]}{dt^i} \]

for \( i = 1,2 \), where

\[ \frac{dE_0[\pi, t]}{dt} = \frac{\text{Ln} \left( \left( \mu \right) \left( \varphi^{-1} \right) \left( \Delta \pi_0 - \mu \right) \right)}{\varphi - 1} + \mu \]

\[ \frac{\partial}{\partial t} \frac{\sqrt{V_0[\pi, t]}}{dt} = \frac{\sigma}{\sqrt{2 |\varphi - 1|}} \sqrt{\frac{\varphi^{2(t+1)} - 1}{\varphi^2 - 1} - 2 \frac{\varphi^{i+1} - 1}{\varphi - 1} + t + 1} \]

\[ \left( 2 \text{Ln} \left( \left( \mu \right) \left( \varphi^{i+1} \right) \right) \right) \left( \frac{\varphi^{2(t+1)} - 1}{\varphi^2 - 1} - 2 \frac{\varphi^{i+1} - 1}{\varphi - 1} + t + 1 \right) \]

\[ \left( \frac{\varphi^{2(t+1)} - 1}{\varphi^2 - 1} - 2 \frac{\varphi^{i+1} - 1}{\varphi - 1} + t + 1 \right) \]

\[ (43) \]
Note that these derivatives only have real solutions for $\varphi > 0$.\(^8\) That, combined with the stationarity condition, limits our range of study to $\varphi \in (0,1)$. For that range of first-order autocorrelation, $\frac{2 \ln |\varphi| \varphi^{2(1+\varphi)}}{\varphi^2 - 1} - \frac{2 \ln |\varphi| \varphi^{1+\varphi}}{\varphi^1 - 1} + 1 > 0$, thus $\frac{\partial \sqrt{V_0[\pi_t]}}{\partial t} > 0$. Risk is monotonically increasing over time, however the rate of increases converges to zero over time, as $\lim_{t \to \infty} \frac{\partial \sqrt{V_0[\pi_t]}}{\partial t} = 0$ (a convergence to zero from the right). $\frac{\partial E_0[\pi_t]}{\partial t}$ converges asymptotically to $\lim_{t \to \infty} \frac{\partial E_0[\pi_t]}{\partial t} = \mu$, starting below that level if $\Delta \pi_0 < \mu$, and starting above that level if $\Delta \pi_0 > \mu$. If $\mu$ is so large that at $t=1$ occurs that $\frac{\partial E_0[\pi_t]}{\partial t} > -Z_0 \frac{\partial \sqrt{V_0[\pi_t]}}{\partial t}$, $Q_{\alpha,t}$ will have its minimum at $Q_{\alpha,1}$ and $QL_\alpha = 0$. For a smaller but still positive $\mu$, $Q_{\alpha,t}$ will be initially a decreasing function of $t$, but it will eventually become an increasing function of $t$, once the effect of $\frac{\partial \sqrt{V_0[\pi_t]}}{\partial t}$ overcomes the effect of $\frac{\partial E_0[\pi_t]}{\partial t}$.

\[ a \in (0,1) \Rightarrow Z_\alpha < 0, \text{ so in order to guarantee a global minimum we would like to find} \]
\[ \frac{\partial^2 E_0[\pi_t]}{\partial t^2} \geq 0 \text{ and } \frac{\partial^2 \sqrt{V_0[\pi_t]}}{\partial t^2} < 0. \]

If we differentiate again the first expression in Eq. (43), we get:

\[ \frac{d^2 E_0[\pi_t]}{dt^2} = \left( \frac{L_n[\varphi]}{\varphi - 1} \right)^2 (\Delta \pi_0 - \mu) \]

We can see that $\Delta \pi_0 < \mu \Rightarrow \frac{\partial^2 E_0[\pi_t]}{\partial t^2} > 0$, and $\Delta \pi_0 > \mu \Rightarrow \frac{\partial^2 E_0[\pi_t]}{\partial t^2} < 0$, but in any case this component wears out, since $\lim_{t \to \infty} \frac{d^2 E_0[\pi_t]}{dt^2} = 0$. If we differentiate again the second expression in Eq. (43), after some operations we get:

\[ \frac{\partial^2 \sqrt{V_0[\pi_t]}}{\partial t^2} = \sigma (L_n[\varphi])^2 \left( \frac{2 \varphi^{2(1+\varphi)}}{\varphi^2 - 1} - \frac{\varphi^{t+1}}{\varphi - 1} \right) \]

\[ \left| \varphi - 1 \right| \left( \frac{2 \varphi^{2(1+\varphi)}}{\varphi^2 - 1} - \frac{\varphi^{t+1}}{\varphi - 1} + t + 1 \right) \]

\[ \left( \frac{2 \sigma \varphi^{2(t+1)} L_n[\varphi] - 2 \varphi^{t+1} L_n[\varphi] + 1}{\varphi^2 - 1} \right)^2 \]

\[ 4 \left| \varphi - 1 \right| \left( \frac{\varphi^{2(t+1)}}{\varphi^2 - 1} - \frac{\varphi^{t+1}}{\varphi - 1} + t + 1 \right)^{3/2} \]

\[ \ categorie}8\] The case $\varphi = 0$ would be easy to handle, by simplifying $\Phi$ in $Q_{\alpha,t}$ before differentiating. However, we do not need to concern ourselves with an analysis of that case, since that was already addressed by Propositions 1 and 2. The case $-1 < \Phi < 0$ introduces solutions in the complex domain, which would require a separate treatment. Numerical experiments show that even in this case the quantile function is unimodal and is endowed with a global minimum.
Because \( \frac{q^{2(t+1)}}{q^2-1} - \frac{q^{t+1}}{q-1} > 0 \) within the range \( \phi E (0,1) \), \( \frac{d^2 \sqrt{V_0[\eta_t]} }{dt^2} \) can be positive for a small \( t \). So \( Q_{\alpha,t} \) could be initially concave, either because of \( \frac{q^{2E_0[\eta_t]} }{\partial t^2} < 0 \), \( \frac{d^2 \sqrt{V_0[\eta_t]} }{dt^2} \) or both. As \( t \) increases, \( \phi' \to 0 \) and \( \frac{d^2 \sqrt{V_0[\eta_t]} }{dt^2} \) \( \sigma \) \( 4|\phi^{-1}| \left( \frac{1}{\phi^{2-1}} + \frac{1}{\phi^{-1} + t+1} \right) \) \( \frac{d^2 q^{t+1}}{dt^2} \approx 0 \), so in any case \( Q_{\alpha,t} \) is convex after a sufficient number of bets. \( Q_{\alpha,t} \) remains convex but increasingly linear, with \( \lim_{t \to \infty} \frac{d^2 \sqrt{V_0[\eta_t]} }{dt^2} = 0^- \) (a convergence to zero from the left).

Putting all the pieces together, we know that \( Q_{\alpha,t} \) could be initially a concave function of \( t \), but eventually it becomes convex. It is guaranteed that \( Q_{\alpha,t} \) has either zero or one inflexion point. As long as \( \mu > 0 \), \( Q_{\alpha,t} \) is unimodal and a global minimum exists. The following section shows how to compute \( \text{MaxQL}_\alpha = \max \{ 0, -\text{Min} Q_{\alpha,t} \} \).

The Python code that numerically verifies the accuracy of our analytical solution can be found at http://srb.com/abstract=2511599 and www.O<antResearch.liifo/Software.h.
Table 1 - Trading parameters for evaluating the stop-out

<table>
<thead>
<tr>
<th>PARAMETERS</th>
<th>PM1</th>
<th>PM2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Expected annual PnL</td>
<td>10,000,000</td>
<td>15,000,000</td>
</tr>
<tr>
<td>Expected Std annual PnL</td>
<td>10,000,000</td>
<td>10,000,000</td>
</tr>
<tr>
<td># Independent trades per year</td>
<td>12</td>
<td>12</td>
</tr>
<tr>
<td>Confidence</td>
<td>95%</td>
<td>95%</td>
</tr>
</tbody>
</table>

Table 2 - Selected Hedge Fund Research Indices

<table>
<thead>
<tr>
<th>HFR Index</th>
<th>Bloomberg Code</th>
<th>Currency</th>
</tr>
</thead>
<tbody>
<tr>
<td>HFR Fund of Funds Composite Index</td>
<td>HFRIFOF Index</td>
<td>USD</td>
</tr>
<tr>
<td>HFR Fund Weighted Composite Index</td>
<td>HFRIFWI Index</td>
<td>USD</td>
</tr>
<tr>
<td>HFR Equity Hedge (Total) Index</td>
<td>HFRIEHI Index</td>
<td>USD</td>
</tr>
<tr>
<td>HFR Macro (Total) Index</td>
<td>HFRIMI Index</td>
<td>USD</td>
</tr>
<tr>
<td>HFR FOF: Diversified Index</td>
<td>HFRIFOFD Index</td>
<td>USD</td>
</tr>
<tr>
<td>HFR ED: Distressed/Restructuring Index</td>
<td>HFRIDSI Index</td>
<td>USD</td>
</tr>
<tr>
<td>HFR EH: Equity Market Neutral Index</td>
<td>HFRIEMNI Index</td>
<td>USD</td>
</tr>
<tr>
<td>HFR FOF: Conservative Index</td>
<td>HFRIFOFC Index</td>
<td>USD</td>
</tr>
<tr>
<td>HFR Event-Driven (Total) Index</td>
<td>HFRIEDI Index</td>
<td>USD</td>
</tr>
<tr>
<td>HFR Macro: Systematic Diversified Index</td>
<td>HFRIMTI Index</td>
<td>USD</td>
</tr>
<tr>
<td>HFR RV: Fixed Income-Corporate Index</td>
<td>HFRIFHY Index</td>
<td>USD</td>
</tr>
<tr>
<td>HFR RV: Multi-Strategy Index</td>
<td>HFRIFI Index</td>
<td>USD</td>
</tr>
<tr>
<td>HFR Relative Value (Total) Index</td>
<td>HFRIRVA Index</td>
<td>USD</td>
</tr>
<tr>
<td>HFR ED: Merger Arbitrage Index</td>
<td>HFRIMAI Index</td>
<td>USD</td>
</tr>
<tr>
<td>HFR RV: Fixed Income-Convertible Arbitrage Index</td>
<td>HFRICAI Index</td>
<td>USD</td>
</tr>
<tr>
<td>HFR Emerging Markets (Total) Index</td>
<td>HFRIEM Index</td>
<td>USD</td>
</tr>
<tr>
<td>HFR Emerging Markets: Asia ex-Japan Index</td>
<td>HFRIEMA Index</td>
<td>USD</td>
</tr>
<tr>
<td>HFR EH: Short Bias Index</td>
<td>HFRISHSE Index</td>
<td>USD</td>
</tr>
<tr>
<td>HFR Emerging Markets: Latin America Index</td>
<td>HFRIELMA Index</td>
<td>USD</td>
</tr>
<tr>
<td>HFR FOF: Strategic Index</td>
<td>HFRIFOS Index</td>
<td>USD</td>
</tr>
<tr>
<td>HFR EH: Quantitative Directional</td>
<td>HFRIENHI Index</td>
<td>USD</td>
</tr>
<tr>
<td>HFR Fund Weighted Composite Index - GBP</td>
<td>HFRIFWIG Index</td>
<td>GBP</td>
</tr>
<tr>
<td>HFR EH: Sector - Technology/Healthcare Index</td>
<td>HFRISTI Index</td>
<td>USD</td>
</tr>
<tr>
<td>HFR FOF: Market Defensive Index</td>
<td>HFRIFOFM Index</td>
<td>USD</td>
</tr>
<tr>
<td>HFR Fund Weighted Composite Index - CHF</td>
<td>HFRIFWIC Index</td>
<td>CHF</td>
</tr>
<tr>
<td>HFR Fund Weighted Composite Index - JPY</td>
<td>HFRIFWJ Index</td>
<td>JPY</td>
</tr>
</tbody>
</table>

Table 2 lists the hedge fund indices in the HFR database with a history from 01/01/1990 to 01/01/2013. These are the indices that we have used in the empirical study presented in Section 10.
Table 3 - Maximum Quantile-Loss and Time under Water considering first-order serial correlation

Table 3 reports the descriptive statistics computed on the hedge fund indices listed in Table 2, when we take into account first-order auto-correlation. MaxQL is the maximum quantile-loss at a $\alpha = 0.05$ significance level, which occurs after $t^*$ observations. At that same significance level, these hedge fund indices remain under water for the period reported in column $TuW$. Penance measures how long it takes to recover from the maximum quantile-loss as a multiple of the time it took to reach the bottom. As we can appreciate, Penance ranges between 1.6 and 3, and it is smaller the higher $\phi$ (Phi) and the higher the ratio $\frac{\mu}{\sigma}$ (Mean divided by Sigma). Although positive serial correlation leads to greater downside potential, longer $t_\alpha^*$ and longer periods under water, the Penance is smaller.
<table>
<thead>
<tr>
<th>Code</th>
<th>Mean</th>
<th>Phi</th>
<th>Sigma</th>
<th>MaxQL</th>
<th>t*</th>
<th>TuW</th>
<th>Penance</th>
</tr>
</thead>
<tbody>
<tr>
<td>HFRIFOF Index</td>
<td>0.0055</td>
<td>0.0000</td>
<td>0.0170</td>
<td>3.53%</td>
<td>6.3996</td>
<td>25.5985</td>
<td>3.0000</td>
</tr>
<tr>
<td>HFRIFWI Index</td>
<td>0.0099</td>
<td>0.0000</td>
<td>0.0264</td>
<td>4.80%</td>
<td>4.8667</td>
<td>19.4669</td>
<td>3.0000</td>
</tr>
<tr>
<td>HFRIMI Index</td>
<td>0.0095</td>
<td>0.0000</td>
<td>0.0215</td>
<td>3.28%</td>
<td>3.4435</td>
<td>13.7740</td>
<td>3.0000</td>
</tr>
<tr>
<td>HFRIFOFD Index</td>
<td>0.0052</td>
<td>0.0000</td>
<td>0.0174</td>
<td>3.96%</td>
<td>7.6477</td>
<td>30.5909</td>
<td>3.0000</td>
</tr>
<tr>
<td>HFRIDSI Index</td>
<td>0.0096</td>
<td>0.0000</td>
<td>0.0188</td>
<td>2.48%</td>
<td>2.5827</td>
<td>10.3309</td>
<td>3.0000</td>
</tr>
<tr>
<td>HFRIEMNI Index</td>
<td>0.0052</td>
<td>0.0000</td>
<td>0.0094</td>
<td>1.16%</td>
<td>2.2389</td>
<td>8.9554</td>
<td>3.0000</td>
</tr>
<tr>
<td>HFRIFOFI Index</td>
<td>0.0048</td>
<td>0.0000</td>
<td>0.0116</td>
<td>1.90%</td>
<td>3.9492</td>
<td>15.7968</td>
<td>3.0000</td>
</tr>
<tr>
<td>HFRIEDI Index</td>
<td>0.0095</td>
<td>0.0000</td>
<td>0.0192</td>
<td>2.63%</td>
<td>2.7554</td>
<td>11.0216</td>
<td>3.0000</td>
</tr>
<tr>
<td>HFRIMITI Index</td>
<td>0.0085</td>
<td>0.0000</td>
<td>0.0216</td>
<td>3.69%</td>
<td>4.3218</td>
<td>17.2870</td>
<td>3.0000</td>
</tr>
<tr>
<td>HFRIFIH Index</td>
<td>0.0072</td>
<td>0.0000</td>
<td>0.0177</td>
<td>2.95%</td>
<td>4.1164</td>
<td>16.4656</td>
<td>3.0000</td>
</tr>
<tr>
<td>HFRIFI Index</td>
<td>0.0069</td>
<td>0.0000</td>
<td>0.0129</td>
<td>1.64%</td>
<td>2.3883</td>
<td>9.5530</td>
<td>3.0000</td>
</tr>
<tr>
<td>HFRIRVA Index</td>
<td>0.0080</td>
<td>0.0000</td>
<td>0.0130</td>
<td>1.42%</td>
<td>1.7701</td>
<td>7.0803</td>
<td>3.0000</td>
</tr>
<tr>
<td>HFRIMAI Index</td>
<td>0.0071</td>
<td>0.0000</td>
<td>0.0104</td>
<td>1.03%</td>
<td>1.4444</td>
<td>5.7777</td>
<td>3.0000</td>
</tr>
<tr>
<td>HFRICAI Index</td>
<td>0.0071</td>
<td>0.0000</td>
<td>0.0200</td>
<td>3.79%</td>
<td>5.3200</td>
<td>21.2800</td>
<td>3.0000</td>
</tr>
<tr>
<td>HFRIEM Index</td>
<td>0.061</td>
<td>0.0000</td>
<td>0.0410</td>
<td>10.98%</td>
<td>10.6100</td>
<td>42.4399</td>
<td>3.0000</td>
</tr>
<tr>
<td>HFRIEMA Index</td>
<td>0.0800</td>
<td>0.0000</td>
<td>0.0382</td>
<td>12.38%</td>
<td>15.4963</td>
<td>61.9851</td>
<td>3.0000</td>
</tr>
<tr>
<td>HFRISHSE Index</td>
<td>0.0017</td>
<td>0.0000</td>
<td>0.0535</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>HFRIEMLA Index</td>
<td>0.011</td>
<td>0.0000</td>
<td>0.0508</td>
<td>15.79%</td>
<td>14.2615</td>
<td>57.0458</td>
<td>3.0000</td>
</tr>
<tr>
<td>HFRIOFS Index</td>
<td>0.0068</td>
<td>0.0000</td>
<td>0.0248</td>
<td>6.09%</td>
<td>8.9946</td>
<td>35.6185</td>
<td>3.0000</td>
</tr>
<tr>
<td>HFRIENHII Index</td>
<td>0.0101</td>
<td>0.0000</td>
<td>0.0367</td>
<td>9.02%</td>
<td>8.9357</td>
<td>35.7430</td>
<td>3.0000</td>
</tr>
<tr>
<td>HFRIFWIG Index</td>
<td>0.0094</td>
<td>0.0000</td>
<td>0.0360</td>
<td>9.33%</td>
<td>9.9416</td>
<td>39.7662</td>
<td>3.0000</td>
</tr>
<tr>
<td>HFRIFOFM Index</td>
<td>0.0056</td>
<td>0.0000</td>
<td>0.0159</td>
<td>3.05%</td>
<td>5.4422</td>
<td>21.7686</td>
<td>3.0000</td>
</tr>
<tr>
<td>HFRIFVIC Index</td>
<td>0.0089</td>
<td>0.0000</td>
<td>0.0390</td>
<td>11.50%</td>
<td>12.8580</td>
<td>51.4319</td>
<td>3.0000</td>
</tr>
<tr>
<td>HFRIFWJ Index</td>
<td>0.0084</td>
<td>0.0000</td>
<td>0.0363</td>
<td>10.58%</td>
<td>12.5579</td>
<td>50.2317</td>
<td>3.0000</td>
</tr>
<tr>
<td>HFRISTI Index</td>
<td>0.011</td>
<td>0.0000</td>
<td>0.0464</td>
<td>13.17%</td>
<td>11.8933</td>
<td>47.5731</td>
<td>3.0000</td>
</tr>
</tbody>
</table>

Table 4 - Maximum Quantile-Loss and Time under Water ignoring serial correlation

It is common to assume returns independence (thus disregarding evidence of serial correlation) to simplify calculations. Table 4 reports the maximum quantile-loss (MaxQL), the observation at which the maximum quantile-loss occurs (t*) and the quantile-time under water (TuW) for a significance level of α = 0.05. Wrongly assuming φ = 0 (see column Phi) may lead to a gross underestimation of the downside potential, in some cases by as much as 70%. As predicted by the "triple penance rule", the number of observations it takes to resurface after the maximum quantile-loss is exactly 3 times the number of observations it took to reach that maximum quantile-loss, with the same confidence level.
<table>
<thead>
<tr>
<th>Code</th>
<th>MaxQL</th>
<th>$t^*$</th>
<th>Alpha1</th>
<th>Mean2</th>
<th>Phi2</th>
<th>Sigma2</th>
<th>Alpha2</th>
</tr>
</thead>
<tbody>
<tr>
<td>HFRIFOF Index</td>
<td>0.0353</td>
<td>6.3996</td>
<td>0.0500</td>
<td>0.0055</td>
<td>0.3594</td>
<td>0.0158</td>
<td>0.1205</td>
</tr>
<tr>
<td>HFRIFWI Index</td>
<td>0.0310</td>
<td>3.4905</td>
<td>0.0500</td>
<td>0.0089</td>
<td>0.3048</td>
<td>0.0192</td>
<td>0.1014</td>
</tr>
<tr>
<td>HFRIEHI Index</td>
<td>0.0480</td>
<td>4.8667</td>
<td>0.0500</td>
<td>0.0099</td>
<td>0.2651</td>
<td>0.0255</td>
<td>0.0975</td>
</tr>
<tr>
<td>HFRIMI Index</td>
<td>0.0328</td>
<td>3.4435</td>
<td>0.0500</td>
<td>0.0095</td>
<td>0.1844</td>
<td>0.0211</td>
<td>0.0796</td>
</tr>
<tr>
<td>HFRIFOFD Index</td>
<td>0.0396</td>
<td>7.6477</td>
<td>0.0500</td>
<td>0.0052</td>
<td>0.3535</td>
<td>0.0163</td>
<td>0.2107</td>
</tr>
<tr>
<td>HFRIDSI Index</td>
<td>0.0248</td>
<td>2.5827</td>
<td>0.0500</td>
<td>0.0096</td>
<td>0.5458</td>
<td>0.0158</td>
<td>0.1312</td>
</tr>
<tr>
<td>HFRIEIIN Index</td>
<td>0.0116</td>
<td>2.2389</td>
<td>0.0500</td>
<td>0.0052</td>
<td>0.1644</td>
<td>0.0093</td>
<td>0.0728</td>
</tr>
<tr>
<td>HFRIFOFC Index</td>
<td>0.0190</td>
<td>3.9492</td>
<td>0.0500</td>
<td>0.0048</td>
<td>0.4557</td>
<td>0.0103</td>
<td>0.1331</td>
</tr>
<tr>
<td>HFRIEDI Index</td>
<td>0.0263</td>
<td>2.7554</td>
<td>0.0500</td>
<td>0.0095</td>
<td>0.3916</td>
<td>0.0177</td>
<td>0.1114</td>
</tr>
<tr>
<td>HFRIMTI Index</td>
<td>0.0369</td>
<td>4.3218</td>
<td>0.0500</td>
<td>0.0085</td>
<td>-0.0188</td>
<td>0.0216</td>
<td>-</td>
</tr>
<tr>
<td>HFRIFIHY Index</td>
<td>0.0295</td>
<td>4.1164</td>
<td>0.0500</td>
<td>0.0072</td>
<td>0.4838</td>
<td>0.0155</td>
<td>0.4000</td>
</tr>
<tr>
<td>HFRIFI Index</td>
<td>0.0164</td>
<td>2.3883</td>
<td>0.0500</td>
<td>0.0069</td>
<td>0.5059</td>
<td>0.0111</td>
<td>0.1224</td>
</tr>
<tr>
<td>HFRIRVA Index</td>
<td>0.0142</td>
<td>1.7701</td>
<td>0.0500</td>
<td>0.0080</td>
<td>0.4528</td>
<td>0.0116</td>
<td>0.1029</td>
</tr>
<tr>
<td>HFRIMAI Index</td>
<td>0.0103</td>
<td>1.4444</td>
<td>0.0500</td>
<td>0.0071</td>
<td>0.2982</td>
<td>0.0100</td>
<td>0.0814</td>
</tr>
<tr>
<td>HFRICAII Index</td>
<td>0.0379</td>
<td>5.3200</td>
<td>0.0500</td>
<td>0.0071</td>
<td>0.5780</td>
<td>0.0163</td>
<td>0.1688</td>
</tr>
<tr>
<td>HFRIEM Index</td>
<td>0.1098</td>
<td>10.6100</td>
<td>0.0500</td>
<td>0.0104</td>
<td>0.3539</td>
<td>0.0383</td>
<td>0.1243</td>
</tr>
<tr>
<td>HFRIBEF Index</td>
<td>0.1238</td>
<td>15.4963</td>
<td>0.0500</td>
<td>0.0080</td>
<td>0.3112</td>
<td>0.0363</td>
<td>0.1139</td>
</tr>
<tr>
<td>HFRISHE Index</td>
<td>-</td>
<td>-</td>
<td>0.0500</td>
<td>-0.0017</td>
<td>0.0907</td>
<td>0.0533</td>
<td>-</td>
</tr>
<tr>
<td>HFRIMEFLA Index</td>
<td>0.1579</td>
<td>14.2615</td>
<td>0.0500</td>
<td>0.0111</td>
<td>0.1969</td>
<td>0.0499</td>
<td>0.0873</td>
</tr>
<tr>
<td>HFRIFOS Index</td>
<td>0.0609</td>
<td>8.9046</td>
<td>0.0500</td>
<td>0.0068</td>
<td>0.3231</td>
<td>0.0235</td>
<td>0.1415</td>
</tr>
<tr>
<td>HFRINHII Index</td>
<td>0.0902</td>
<td>8.9357</td>
<td>0.0500</td>
<td>0.0101</td>
<td>0.2011</td>
<td>0.0359</td>
<td>0.0872</td>
</tr>
<tr>
<td>HFRIFWIG Index</td>
<td>0.0933</td>
<td>9.9416</td>
<td>0.0500</td>
<td>0.0094</td>
<td>0.2314</td>
<td>0.0350</td>
<td>0.0940</td>
</tr>
<tr>
<td>HFRIFOMF Index</td>
<td>0.0305</td>
<td>5.4422</td>
<td>0.0500</td>
<td>0.0056</td>
<td>0.0422</td>
<td>0.0159</td>
<td>0.0567</td>
</tr>
<tr>
<td>HFRIFWIC Index</td>
<td>0.1150</td>
<td>12.8580</td>
<td>0.0500</td>
<td>0.0089</td>
<td>0.0505</td>
<td>0.0390</td>
<td>0.0586</td>
</tr>
<tr>
<td>HFRIFWIJ Index</td>
<td>0.1058</td>
<td>12.5579</td>
<td>0.0500</td>
<td>0.0084</td>
<td>0.0954</td>
<td>0.0361</td>
<td>0.0667</td>
</tr>
<tr>
<td>HFRISTI Index</td>
<td>0.1317</td>
<td>11.8933</td>
<td>0.0500</td>
<td>0.0111</td>
<td>0.1608</td>
<td>0.0458</td>
<td>0.0795</td>
</tr>
</tbody>
</table>

Table 5 - Intended vs. actual probability of false positives

Table 5 reports the effective probability of false positives when portfolio managers or strategies are stopped-out based on quantile-loss limits that ignore first-order autocorrelation. For all hedge fund styles, the actual probability of false positives is considerably greater than the one intended. Because many firms evaluate their managers’ performance assuming independent returns (e.g., Sharpe ratio), they are improperly stopping them out. In some cases, they may be firing more than three times the number of skillful portfolio managers, compared to the number they were willing to accept under the (wrong) assumption of returns independence.
FIGURES

Figure 1 - The Triple Penance rule

Figure 1 provides a graphical representation of the Triple Penance rule. It takes three time longer to recover from the maximum quantile-loss \((\text{MaxTuW}_\alpha)\) than the time it took to produce it \((t^*_\alpha)\), for a given significance level \(\alpha < \frac{1}{2}\), regardless of the PM's Sharpe ratio.
Figure 2 plots the quantile-loss function $Q_{a,t}$ as time passes, for $\alpha = 0.05$, where PM1 has an annualized Sharpe ratio of 1 (annual mean and standard deviation of US$10m), and PM2 has an annualized Sharpe ratio of 1.5 (annual mean of US$15m, and annual standard deviation of US$10m). For that 95% confidence level, PM1 reaches a maximum quantile-loss at US$6,763,858.64 after 0.676 years, and remains up to 2.706 years under water, whereas PM2 reaches a maximum quantile-loss at US$4,509,239.09 after 0.3 years, and remains 1.202 years under water. These results are consistent with the "triple penance" rule.
Figure 3 plots the quantile-loss function $Q_{a,t}$ as time passes, where $a$ has been computed individually to meet the goal of being under water a maximum of 1 year. As a result, whereas before we had a tighter stop-out for the portfolio manager with higher Sharpe ratio, now the stricter stop-out is imposed on the portfolio manager with lower Sharpe ratio. This is consistent with the business reality that a hedge fund faces the risk of seeing good portfolio managers defecting if a performance bonus cannot be paid within a certain period.
Figure 4 plots the quantile-loss function $Q_{\alpha t}$ as time passes, where $\bar{a}$ has been computed to meet the goal of being under water a maximum of 2 years. As a result of the longer maximum period (2 years instead of 1), we are even more permissive of in setting stop-out levels for PM2 than we were in Figure 2.
Figure 5 - The effect of higher serial correlation on Penance

Figure 5 plots \textit{Penance} for hedge fund indices with various $\Phi$. Although positive serial correlation leads to greater quantile-losses, longer $t^*_\alpha$ and longer periods underwater, the \textit{Penance} may be substantially smaller. In particular, \textit{Penance} is smaller the higher $\Phi$ (\textit{Phi}) and the higher the ratio $\frac{\mu}{\sigma}$ (\textit{Mean} divided by \textit{Sigma}).
REFERENCES

• Baillie, DD. and MM. Llopperzz dz elé PPrroaddoo ((22001122)):: "TThhee SSIiiaaahmaapppee RRarraaaiisssiooonn EErrfficiieenncciieerrrnnn FFrooommiitttuireerrrnnn." JJoouurrnnaall ooff RRrriisskk, 1155((22)), WWüünnnttteeerr, 33---4444. AAVVaaSsaaailallbbliiee aatt http://lissvora.economics.aub.edu.lb/ \( 1 \times 2 = 4 \), 164 3.

• Baillie, DD. and MM. Llopperzz dz elé PPrroaddoo ((22001133)):: "TThhee SSIiiaaahmaapppee RRarraaaiisssiooonn EErrfficiieenncciieerrrnnn FFrooommiitttuireerrrnnn." JJoouurrnnaall ooff RRrriisskk, 1155((22)), WWüünnnttteeerr, 33---4444. AAVVaaSsaaailallbbliiee aatt http://lissvora.economics.aub.edu.lb/ \( 1 \times 2 = 4 \), 164 3.

• CCaaaann, PP. , HH. ZZëëhhaaanneeggg asmmdd DD. HHaaddjgjiiillliiaaddddd ((220011)):: ""MMMaaxxxiiiummmmm DDrрааawwwddoownn ooff HTThheessrrrееetttrrrrееetttrrciiull aammdd AAQQppplliieedd FFiiinnnaanncciieerrrnnn., 114466(88), pppp; 1119955---11223300.

• CChheerkkhhkkllldddooovv, AA.. SS. UUnnrryyaassseerrv aammdd MM. ZZääabbaarrraaannkkkkkiinn ((22000033)):: ""PPhoоорrttffiiillooiooo ooootttpppiimmiitttiiimmazzaatttiooonn wwiitthh ddrraawwwddoownn ocommsssttrrraaaiinnitttt,"", iiinn bbh.. SScchheeurreerreeerr ((IEEddddd)): ""AAAssssserrtt aammdd liiisstaaabbiillliitttyy mmaannnaaagggjeemmeerennntt tttccooolllss."" RRiisskk BBloooookkkss.

• CChheerkkhhkkllldddooovv, AA.. SS. UUnnrryyaassseerrv aammdd MM. ZZääabbaarrraaannkkkkkiinn ((22000055)):: ""DDrрааawwwddoownn rrmmeennaaasssuurrreee ainnn ppoorrttffiiillooiooo ooootttpppiimmiitttiiimmazzaatttiooonn."" JJoouurrnnaall ooff IIiinnnnteerrnaaattiioonnaall FFiiinnnaanncciieerrrnnn ooff TTihheessrrrееetttrrrrееetttrrciiull aammdd AAQQppplliieedd FFiiinnnaanncciieerrrnnn., 114466(88), pppp; 1119955---11223300.

• CChheerkkhhkkllldddooovv, AA.. SS. UUnnrryyaassseerrv aammdd MM. ZZääabbaarrraaannkkkkkiinn ((22000066)):: ""DDrрааawwwddoownn rrmmeennaaasssuurrreee ainnn ppoorrttffiiillooiooo ooootttpppiimmiitttiiimmazzaatttiooonn."" JJoouurrnnaall ooff IIiinnnnteerrnaaattiioonnaall FFiiinnnaanncciieerrrnnn ooff TTihheessrrrееetttrrrrееetttrrciiull aammdd AAQQppplliieedd FFiiinnnaanncciieerrrnnn., 114466(88), pppp; 1119955---11223300.

• GGGerrtttmmmaannssoonnkkkkyy, MM.. AA. LLoovv aammdd FF. MMaaakkkaarrroovv ((22000044)):: ""AAAm mmeeccccoooonnnmmmeerrrrtteerrriiccc mmooollleerrr ooff sseerrrriiimaaall ccoorrreeelllaattiiooonn aammdd iiiiillliiiiiqqquuuuiidddiitttyy iinnn hheedddeeegggg fffiiiaannnddd mmeettiiuurrrmmss.."" JJoouurrnnaall ooff FFiiinnnaanncciieerrrnnn EEeccccoonnnnmmtooocccciieerrrnnn, 335555, pppp; 55522299---5556660099.

• GGGrrimmmssootttteerrzaallddd, CC. aammdd SSnnneerrrillll ((1119999977)):: ""Illeennnntttttooooddiiuaaccttttiiooonn ttttoo PPnoobbbaabbiiillliitttyy."" AAAm mmeeccccoooonnnmm MMaattaallthheerrmmeennaatttiiicccciiaall SSloooocccciieerrtttyy, CChhaaupppttreerr 77, 2 2 EEddiiittttiiiooonn.

• GGGrrooosssoonnmmmaann, SS. aammdd ZZ. ZZëëhhaaanneeggg ((1119999933)):: ""ODDpppttimmmaall IInnnssummeeerrsstttteerrmmmeennntt SSIiiaaahmaapppee RRarraaaiisssiooonn ffoorr ccooommmittttttooollliiinnngg ddrraawwwddoownn ooffs."" MMaaattthheerrmmeennaatttiiicccciiaall FFiiinnnaanncciieerrrnnn, VVooolll.. 33, pppp; 22244411---22277766.

• HHHaammmmllllltttoonn, JJ. ((1119999944)):: ""TTiimmmeerr SSeeerrrrriieessoo SSAAmmayyyllysstts.."" PPPrrriimmncceerrttiooonn, CChhaaupppttreerr 44.

• HHHlaayyeerr, bb. ((22000066)):: ""MMMaaxxxiiiummmmm DDrрааawwwddoownn ooffs hheedddeeegggg fffiiiaannnddd wwiitthh sseerrrriiimaaall ccoorrreeelllaattiiooonn."" JJoouurrnnaall ooff FFiiinnnaanncciieerrrnnn ooff AAIIllllettteerrrnnaaatttiiivvee IInnnssummeeerrstttteerrmmeeennntts, VVooolll.. 88(44), pppp; 222666---22233388.

• JJoouurrnnoonn, PP. ((22000066)):: ""VVaalluuueerr aattt RRiisskk: TTiimmee mmezzzzw wbbrennncchhhmmeennaaarrkkkk fffoorr mmaannnaaagggjiiinnngg fffiiiaannncceaaicciiaall rriisskk."" MMmccCGGrraaamm--WWllllllhheeeilllll, 3 3 EEddiiittttiiiooonn.

• LLiiinn, CC.. WW. OObbroowwennn aammdd II. OObbroowwennn ((22001100)):: ""TThheeesssee aassssoooccchiiaattiiooonn bbeerrttteerrrveerrmm mmaarrnaakkttt rriisskk ddiissccchhooossuurrreerr reeccppoorrtttinggg aammdd fffaaarrmmm mmiisskk: iiinnn iiinnnnpppaaccxtt ooff SSEEECC FFRRRR NNoo.. 4488", ""TTiimmee HHHoossuurrrr mmaaiff FFiiinnnaanncciieerrrnnn ooff AAQQppplliieedd FFiiinnnaanncciieerrrnnn., 1122556666((44)), JJttuuulllyy/A/Auuuggggsuuss, pppp; 333555---444666.

• LLiioo, AA. ((22000022)):: ""TThheeesssee SSIIitttaattttiisssiieessccoo ooff SSIiiaaahmaapppee RRarraaaiisssiooonn."" JJoouurrnnaall ooff FFiiinnnaanncciieerrrnnn AAAnnnnaallllyyssstteess, VVooolll.. 5588, NNoo.. 44, JJttuuulllyy/A/Auuuggggsuuss.

• LLloopppeerrzz dz elé PPrroaddoo, MM. aammdd AA. PPrrreiiijjjuummm ((22000044)):: ""MMMeexxxaaasssuurrreeinnngg tttbhhee LLoossosss PPPoorrttteeerrrnttiitssiiiaall ooff HHeedddeeegggg FFFiiiaannnddd SSIiiaaahmaapppee RRarraaaiisssiooonn."" JJoouurrnnaall ooff AAIIllllettteerrrnnaaatttiiivvee IInnnssummeeerrstttteerrmmeeennntts, VVooolll.. 77(11), pppp; 77---333111. AAVVaaSsaaailallbbliiee aatt at http://ssrn.com/abstract=641702.

• MMagdooonn, IInnsuall, M. and A. AAtttiiyya ((220044)):: ""MMMaaxxxiimmumuurnn dddrraawwwddoownn."" RRiisskk MMaggggaaanneerrr, October.
THE DEFLATED SHARPE RATIO: CORRECTING FOR SELECTION BIAS, BACKTEST OVERFITTING AND NON-NORMALITY

David H. Bailey †
Marcos Lopez de Prado ‡

First version: April 15, 2014
This version: July 31, 2014

Journal of Portfolio Management, Forthcoming, 2014

† Recently retired from Lawrence Berkeley National Laboratory, Berkeley, CA 94720. Research Fellow at the University of California, Davis, Department of Computer Science. E-mail: davd@davdbbailey.cooi
‡ Senior Managing Director, Guggenheim Partners, New York, NY 10017. Research Affiliate, Computational Research Division, Lawrence Berkeley National Laboratory, Berkeley, CA 94720. E-mail: lop-zd-prado@lbl.gov

Special thanks are owed to Prof. David J. Hand (Royal Statistical Society), who reviewed an early version of this paper and suggested various extensions. We are also grateful to Matthew Beddall (Winton Capital), Jose Blanco (Credit Suisse), Jonathan M. Borwein (University of Newcastle, Australia), Sid Browne (Credit Suisse), Peter Carr (Morgan Stanley, NYU), Marco Dion (J.P. Morgan), Matthew D. Foreman (University of California, Irvine), Stephanie Ger (Berkeley Lab), Campbell Harvey (Duke University), Kate Land (Winton Capital), Jeffrey S. Lange (Guggenheim Partners), Attilio Meucci (KKR, NYU), Philip Protter (Columbia University), Riccardo Rebonato (PIMCO, University of Oxford), Mark Roulston (Winton Capital), Luis Viceira (HBS), John Wu (Berkeley Lab) and Jim Qiji Zhu (Western Michigan University).

The statements made in this communication are strictly those of the authors and do not represent the views of Guggenheim Partners or its affiliates. No investment advice or particular course of action is recommended. All rights reserved.
THE DEFLATED SHARPE RATIO: CORRECTING FOR SELECTION BIAS, BACKTEST OVERFITTING AND NON-NORMALITY

ABSTRACT

With the advent in recent years of large financial data sets, machine learning and high-performance computing, analysts can backtest millions (if not billions) of alternative investment strategies. Backtest optimizers search for combinations of parameters that maximize the simulated historical performance of a strategy, leading to backtest overfitting.

The problem of performance inflation extends beyond backtesting. More generally, researchers and investment managers tend to report only positive outcomes, a phenomenon known as selection bias. Not controlling for the number of trials involved in a particular discovery leads to over-optimistic performance expectations.

The Deflated Sharpe Ratio (DSR) corrects for two leading sources of performance inflation: Selection bias under multiple testing and non-Normally distributed returns. In doing so, DSR helps separate legitimate empirical findings from statistical flukes.

Keywords: Sharpe ratio, Non-Normality, Probabilistic Sharpe ratio, Backtest overfitting, Minimum Track Record Length, Minimum Backtest Length.

JEL Classification: GO, Gl, G2, G15, G24, E44.
AMS Classification: 91G10, 91G60, 91G70, 62C, 60E.
Today's quantitative teams routinely scan through petabytes of financial data, looking for patterns invisible to the naked eye. In this endeavor, they are assisted by a host of technologies and advancing mathematical fields. Big data, machine learning, cloud networks and parallel processing have meant that millions (if not billions) of analyses can be carried out on a given dataset, searching for profitable investment strategies. To put this in perspective, the amount of data used by most quant teams today is comparable to the memory stored by Netflix to support its video-streaming business nationwide. This constitutes a radical change compared to the situation a couple of decades ago, when the typical financial analyst would run elementary arithmetic calculations on a spreadsheet containing a few thousand datapoints. In this paper we will discuss some of the unintended consequences of utilizing scientific techniques and high-performance computing without controlling for selection bias. While these problems are not specific to finance, examples are particularly abundant in financial research.

Backtests are a case in point. A backtest is a historical simulation of how a particular investment strategy would have performed in past. Although backtesting is a powerful and necessary research tool, it can also be easily manipulated. In this article we will argue that the most important piece of information missing from virtually all backtests published in academic journals and investment offerings is the number of trials attempted. Without this information, it is impossible to assess the relevance of a backtest. Put bluntly, a backtest where the researcher has not controlled for the extent of the search involved in his or her finding is worthless, regardless of how excellent the reported performance might be. Investors and journal referees should demand this information whenever a backtest is submitted to them, although even this will not remove the danger completely.

**MULTIPLE TESTING**

Investment strategies are typically judged according to performance statistics. Because any measurement is associated with a margin of error, the process of selecting a statistical model is an instance of decision-making under uncertainty. We can never be certain that the true performance is above a certain threshold, even if the estimated performance is. Two sorts of errors arise: The *Type I Error*, with probability $\alpha$ (also called "significance level"), and the *Type II Error*, with probability $\beta$. The Type I Error occurs when we choose a strategy that should have been discarded (a "false positive"), and the Type II Error occurs when we discard a strategy that should have been chosen (a "false negative"). Decision makers are often more concerned with "false positives" than with "false negatives". The reason is, they would rather exclude a true strategy than risking the addition of a false one. For risk-averse investors, a lost opportunity is less worrisome than an actual loss. For this reason, the standard practice is to design statistical tests which set the Type I Error probability to a low threshold (e.g., $\alpha = 5\%$), while maximizing the power of the test, defined as $1 - \beta$.

Suppose now that we are interested in analyzing multiple strategies on the same dataset, with the aim of choosing the best, or at least a good one, for future application. A curious problem then emerges: As we test more and more strategies, each at the same significance level $\alpha$, the overall probability of choosing at least one poor strategy grows. This is called the multiple testing problem, and it is so pervasive and notorious that the American Statistical Society explicitly warns against it its Ethical Guidelines (American Statistical Association, 1997, guideline #8):
Running multiple tests on the same data set at the same stage of an analysis increases the chance of obtaining at least one invalid result. Selecting the one "significant" result from a multiplicity of parallel tests poses a grave risk of an incorrect conclusion. Failure to disclose the full extent of tests and their results in such a case would be highly misleading.

SELECTION BIAS
Researchers conducting multiple tests on the same data tend to publish only those that pass a statistical significance test, hiding the rest. Because negative outcomes are not reported, investors are only exposed to a biased sample of outcomes. This problem is called "selection bias", and it is caused by multiple testing combined with partial reporting, see Roulston and Hand [2013]. It appears in many different forms: Analysts who do not report the full extent of the experiments conducted ("file drawer effect"), journals that only publish "positive" outcomes ("publication bias"), indices that only track the performance of hedge funds that didn't blow up ("survivorship bias"), managers who only publish the history of their (so far) profitable strategies ("self-selection bias", "backfilling"), etc. What all these phenomena have in common is that critical information is hidden from the decision-maker, with the effect of a much larger than anticipated Type I Error probability. Ignoring the full extent of trials makes the improbable more probable, see Hand [2014].

The danger of encountering false positives is evident in High-Throughput Screening (HTS) research projects, such as those utilized to discover drug treatments, the design of chemical compounds or microarrays genetic testing. Bennett et al. [2010] were awarded the 2012 Ig Nobel prize for showing that even a salmon's dead brain can appear to show significant activity under multiple MRI testing. More seriously, the pharmaceutical field has been stung with numerous recent instances of products that look great based on published trials, but which disappoint when actually fielded. The problem here, in many cases, is that only the results of successful tests are typically published, thereby introducing a fundamental bias into the system. Such experiences have led to the AllTrials movement (see http://ailtriak.net), which would require the results of all trials to be made public.

In spite of the experience of HTS projects, it is rare to find financial studies that take into account the increased false positive rates that result from hiding multiple testing. The extent of this problem has led some researchers to paraphrase Ioannidis [2005] and conclude that "most claimed research findings in financial economics are likely false", see Harvey et al. [2013].

BACKTEST OVERFITTING
What constitutes a legitimate empirical finding? After a sufficient number of trials, it is guaranteed that a researcher will always find a misleadingly profitable strategy, a false positive. Random samples contain patterns, and a systematic search through a large space of strategies will eventually lead to identifying one that profits from the chance configuration of how the random data have fallen. When a set of parameters are optimized to maximize the performance of a backtest, an investment strategy is likely to be fit to such flukes. This phenomenon is called
backtest overfitting, and we refer the reader to Bailey et al. [2014] for a detailed discussion. Although the historical performance of an optimized backtest may seem promising, the random pattern that fuels it is unlikely to repeat itself in the future, hence rendering the strategy worthless. Structural breaks have nothing to do with the failure of the strategy in this situation.

Let us elucidate this point with an example. After tossing a fair coin ten times we could obtain by chance a sequence such as \(+,+,+,+,-,-,-,-,\), where \(+\) means head and \(-\) means tail. A researcher could determine that the best strategy for betting on the outcomes of this coin is to expect \(+\) on the first five tosses, and for \(-\) on the last five tosses (a typical "seasonal" argument in the investment community). When we toss that coin ten more times, we may obtain a sequence such as \(\ldots+,-,+,-,-,+,-\), where we win 5 times and lose 5 times. That researcher's betting rule was overfit, because it was designed to profit from a random pattern present only in the past. The rule has null power over the future, regardless of how well it appears to have worked in the past.

Competition among investment managers means that the ratio of signal to noise in financial series is low, increasing the probability of "discovering" a chance configuration, rather than an actual signal. The implication is that backtest overfitting is hard to avoid. Clearly this is a critical issue, because most investment decisions involve choosing among multiple candidates or alternatives.

AANN OONLLIIINNEE TTOOOOLL TTOOO EEXXPPPLLOOOREEE BBAAACCKKEETTTEEESSSTT OOVVEERRFFIITTITTTTIINNNGG

BACKTEST OVERFITTING UNDER MEMORY EFFECTS

In order to understand the effects of backtest overfitting on out-of-sample performance, we must introduce an important distinction: Financial processes with and without memory. A coin, whether fair or biased, does not have memory. The 50% heads ratio does not arise as a result of the coin "remembering" previous tosses. Patterns emerge, but they are "diluted away" as additional sequences of tosses are produced. Now suppose that we add a memory chip to that coin, such that it remembers the previous tosses and distributes its mass to compensate its outcomes. This memory actively "undoes" recent historical patterns, such that the 50% heads ratio is quickly recovered. Just as a spring memorizes its equilibrium position, financial variables that have acquired a high tension will return to equilibrium violently, undoing previous patterns.

The difference between "diluting" and "undoing" a historical pattern is enormous. Diluting does not contradict your bet, but undoing generates outcomes that systematically go against your bet!
Backtest overfitting tends to identify the trading rules that would profit from the most extreme random patterns in sample. In presence of memory effects, those extreme patterns must be undone, which means that backtest overfitting will lead to loss maximization. See Bailey et al. (2014) for a formal mathematical proof of this statement. Unfortunately, most financial series exhibit memory effects, a situation that makes backtest overfitting a particularly onerous practice, and may explain why so many systematic funds fail to perform as advertised.

BACKTEST OVERFITTING AND THE HOLDOUT METHOD
Practitioners attempt to validate their backtests using several approaches. The holdout method is perhaps the best known example, see Schorfheide and Wolpin [2012] for a description. A researcher splits the available sample into two non-overlapping subsets: The in-sample subset (IS), and the out-of-sample subset (OOS). The idea is to discover a model using the IS subset, and then validate its generality on the OOS subset. The k-fold cross-validation method repeats the process of sample splitting k times, which contributes to the reduction of the estimation error's variance. Then, OOS results are tested for statistical significance. For example, we could reject models where the OOS performance is inconsistent with the IS performance.

From our earlier discussion, the reader should understand why the holdout method cannot prevent backtest overfitting: Holdout assesses the generality of a model as if a single trial had taken place, again ignoring the rise in false positives as more trials occur. If we apply the holdout method enough times (say 20 times for a 95% confidence level), false positives are no longer unlikely: They are expected. The more times we apply holdout, the more likely an invalid strategy will pass the test, which will then be published as a single-trial outcome. While model validation techniques are relevant for guarding against testing hypotheses suggested by the data (called "Type III errors"), they do not control for backtest overfitting.

GENERAL APPROACHES TO MULTIPLE TESTING
In the previous sections we have introduced the problem of multiple testing. We have explained how multiple testing leads to an increased probability of false positives, and hiding negative outcomes of multiple testing leads to selection bias. We have seen that backtest overfitting is a particularly expensive form of selection bias, as a result of memory effects present in financial series. Finally, we have discussed why popular model validation techniques fail to address these problems. So what would constitute a proper strategy selection method?

Statisticians have been aware of the multiple testing and selection bias problems since the early part of the twentieth century, and have developed methods to tackle them (e.g. the classic Bonferroni approach to multiple testing, and Heckman’s work on selection bias - which won him a Nobel Prize). Recently, however, with the increase in large data sets and, in particular, challenges arising from bioinformatics, tackling multiple testing problems has become a hot research topic, with many new advances being made. See, for example, Dudoit and van der Laan [2008] and Dmitrienko et al. [2010]. For controlling familywise error rate - the probability that at least one test of multiple null hypotheses will be falsely rejected - a variety of methods have been developed, but researchers have also explored alternative definitions of error rate (e.g., Holm [1979]). In particular, the complementary false discovery rate (e.g. Benjamini and
Hochberg [1995]) has attracted a great deal of attention. Instead of looking at the probability of falsely rejecting a true null hypothesis, this looks at the probability that a rejected hypothesis is null (which is arguably more relevant in most scientific situations, if not in the manufacturing situations for which Neyman and Pearson originally developed their approach to hypothesis testing).

Bailey et al. [2013] introduce a new cross-validation technique to compute the Probability of Backtest Overfitting (PBO). PBO assesses whether the strategy selection process has been conducive to overfitting, in the sense that selected strategies tend to underperform the median of trials out of sample. PBO is non-parametric and can be applied to any performance statistic, however it requires a large amount of information. We dedicate the remainder of the paper to develop a new parametric method to correct the Sharpe ratio for the effects of multiple testing, inspired by the false discovery rate approach.

EXPECTED SHARPE RATIOS UNDER MULTIPLE TRIALS
The Sharpe Ratio (SR) is the most widely used performance statistic (Sharpe [1966, 1975, 1994]). It evaluates an investment in terms of returns on risk, as opposed to return on capital. Portfolio managers are keen to improve their SRs, in order to rank higher in databases such as Hedge Fund Research, and receive greater capital allocations. Setting a constant cut-off threshold for SR above which portfolio managers or strategies are considered for capital allocation leads to the same selection bias discussed earlier: As more candidates are considered, the false positive rate keeps growing.

More formally, consider a set of $N$ independent backtests or track records associated with a particular strategy class (e.g., Discretionary Macro). Each element of the set is called a trial, and it is associated with a SR estimate, $SR_n$, with $n = 1, ..., N$. Suppose that these trials’ $\{SR_n\}$ follow a Normal distribution, with mean $E[\{\tilde{SR}_n\}]$ and variance $V[\{\tilde{SR}_n\}]$. This is not an unreasonable assumption, since the concept of "strategy class" implies that the trials are bound by some common characteristic pattern. In other words, we assume that there is a mean and variance associated with the trials’ $\{SR_n\}$ for a given strategy class. For example, we would expect the $E[\{\tilde{SR}_n\}]$ from High Frequency Trading trials to be greater than $E[\{\tilde{SR}_n\}]$ from Discretionary Macro. Appendix 1 proves that, under these assumptions, the expected maximum of $\{SR_n\}$ after $N = 1$ independent trials can be approximated as:

$$E[\max(\tilde{SR}_n)] \approx E[\{\tilde{SR}_n\}] + \sqrt{V[\{\tilde{SR}_n\}]} \left( (1-\gamma)Z^{-1} \left[ 1 - \frac{1}{N} \right] + \gamma Z^{-1} \left[ 1 - \frac{1}{N} e^{-1} \right] \right)$$

where $\gamma$ (approx. 0.5772) is the Euler-Mascheroni constant, $Z$ is the cumulative function of the standard Normal distribution, and $e$ is Euler’s number. Appendix 2 runs numerical experiments that contrast the accuracy of this solution. Appendix 3 shows how $N$ can be determined when the trials are not independent.
Equation 1 tells us that, as the number of independent trials \( (N) \) grows, so will grow the expected maximum of \( \{SR_n\} \). Exhibit 1 illustrates this point for \( E[S\tilde{R}_n] = 0, V[S\tilde{R}_n] = 1 \) and \( N \in [1,000] \).

[EXHIBIT 1 HERE]

Consequently, it is not surprising to obtain good backtest results or meet better portfolio managers as we parse through more candidates. This is a consequence of purely random behavior, because we will observe better candidates even if there is no investment skill associated with this strategy class \( (E[S\tilde{R}_n] = 0, V[S\tilde{R}_n] > 0) \). In the following section we will use this fact to adjust the strategy rejection threshold as the number of independent trials increases.

**THE DEFLATED SHARPE RATIO**

When an investor selects the best performing strategy over a large number of alternatives, she exposes herself to "the winner's curse." As we have shown in the previous section, she is likely to choose a strategy with an inflated Sharpe ratio. Performance out of sample is likely to disappoint, a phenomenon called "regression to the mean" in the shrinkage estimation literature, see Efron [2001]. In what follows we will provide an estimator of the Sharpe ratio that undoes the selection bias introduced by multiple testing, while also correcting for the effects of Non-Normal returns.

The Probabilistic Sharpe Ratio (PSR), developed in Bailey and Lopez de Prado [2012a], computes the probability that the true SR is above a given threshold. This rejection threshold is determined by the user. PSR takes into account the sample length and the first four moments of the returns' distribution. The reason for this, as several studies have demonstrated, is the inflationary effect of short samples and samples drawn from non-Normal returns distributions. We refer the interested reader to Lo [2002], Mertens [2002], Lopez de Prado and Peijan [2004], Ingersoll et al. [2007] for a discussion.

Our earlier analysis has shown a second source of inflation, caused by selection bias. Both sources of inflation, when conflated, can lead to extremely high estimated values \( \tilde{SR} \), even when the true SR may be null. In this paper we propose a Deflated Sharpe Ratio (DSR) statistic that corrects for both sources of \( \tilde{SR} \) inflation, defined as:

\[
DSR \equiv PSR(\tilde{SR}_0) = Z \left[ \frac{(\tilde{SR} - \tilde{SR}_0)\sqrt{T - 1}}{\sqrt{1 - \gamma_3^{\tilde{SR}} + \tilde{\gamma}_4 - 1} \tilde{SR}^2} \right]
\]

where \( \tilde{SR}_0 = \sqrt{V[S\tilde{R}_n]} \bigg] \left( (1 - \gamma)Z^{-1} \left[ 1 - \frac{1}{N} \right] + \gamma Z^{-1} \left[ 1 - \frac{1}{N} e^{-1} \right] \right) \), \( V[S\tilde{R}_n] \) is the variance across the trials' estimated SR and \( N \) is the number of independent trials. We also use information concerning the selected strategy: \( \tilde{SR} \) is its estimated SR, \( T \) is the sample length, \( \tilde{\gamma}_3 \) is
the skewness of the returns distribution and $f_4$ is the kurtosis of the returns distribution for the selected strategy. $Z$ is the cumulative function of the standard Normal distribution.

Essentially, DSR is a PSR where the rejection threshold is adjusted to reflect the multiplicity of trials. The rationale behind DSR is the following: Given a set of SR estimates, $\{SR_n\}$, its expected maximum is greater than zero, even if the true SR is zero. Under the null hypothesis that the actual Sharpe ratio is zero, $H_0:SR = 0$, we know that the expected maximum $\overline{SR}$ can be estimated as the $\overline{SR}_0$ in Eq. (2). Indeed, $\overline{SR}_0$ increases as more independent trials are attempted ($N$), or the trials involve a greater variance ($\sqrt{\frac{1}{N}\{SR_n\}}$).

Note that the standard $\overline{SR}$ is computed as a function of two estimates: Mean and standard deviation of returns. DSR deflates SR by taking into consideration five additional variables: The non-Normality of the returns ($\tilde{\gamma}_3, \tilde{\gamma}_4$), the length of the returns series ($T$), the variance of the SRs tested ($\sqrt{\frac{1}{N}\{SR_n\}}$), as well as the number of independent trials involved in the selection of the investment strategy ($N$).

In an excellent recent study, Harvey and Liu [2014], henceforth denoted HL, compute the threshold that a new strategy’s Sharpe ratio must overcome in order to evidence greater performance. HL’s solution is based on Benjamini and Hochberg’s framework. The role of HL’s threshold is analogous to the role played by our $E[\max\{5\overline{SR}_n\}]$ in Eq. (1), which we derived through Extreme Value Theory. DSR uses this threshold to deflate a particular Sharpe ratio estimate (see Eq. (2)). In other words, DSR computes how statistically significant a particular $\overline{SR}$ is, considering the set of trials carried out so far. In the current paper we apply DSR to the $E[\max\{5\overline{SR}_n\}]$ threshold, but DSR could also be computed on HL’s threshold. From that perspective, these two methods are complementary, and we encourage the reader to compute DSR using both thresholds, $E[\max\{5\overline{SR}_n\}]$ as well as HL’s.

### A NUMERICAL EXAMPLE

Suppose that a strategist is researching seasonality patterns in the treasury market. He believes that the U.S. Treasury’s auction cycle creates inefficiencies that can be exploited by selling off-the-run bonds a few days before the auction, and buying the new issue a few days after the auction. He backtests alternative configurations of this idea, by combining different pre-auction and post-auction periods, tenors, holding periods, stop-losses, etc. He uncovers that many combinations yield an annualized $\overline{SR}$ of 2, with a particular one yielding a $\overline{SR}$ of 2.5 over a daily sample of 5 years.

Excited by this result, he calls an investor asking for funds to run this strategy, arguing that an annualized $\overline{SR}$ of 2.5 must be statistically significant. The investor, who is familiar with a paper recently published by the *Journal of Portfolio Management*, asks the strategist to disclose: i) The number of independent trials carried out ($TV$); ii) the variance of the backtest results ($\sqrt{\frac{1}{N}\{\overline{SR}_n\}}$); iii) the sample length ($T$); and iv) the skewness and kurtosis of the returns ($\tilde{\gamma}_3, \tilde{\gamma}_4$). The analyst responds that $N = 100$, $\sqrt{\frac{1}{N}\{\overline{SR}_n\}} = \frac{1}{2}$, $T=1250$, $\tilde{\gamma}_3 = -3$ and $\tilde{\gamma}_4 = 10$.
Shortly after, the investor declines the analyst's proposal. Why? Because the investor has determined that this is not a legitimate empirical discovery at a 95% confidence level. In particular, 
\[ S_R = \frac{1}{\sqrt{250}} \left( (1 - \gamma)Z^{-1}[1 - \frac{1}{100}] + \gamma Z^{-1}[1 - \frac{1}{100} e^{-1}] \right) \approx 0.1132 \], non-annualized (with 250 observations per year), and \( DSR \approx Z \left[ \frac{2.5}{\sqrt{250}} - \frac{0.1132}{\sqrt{250}} \right] = 0.9004 < 0.95. \]

[EXHIBIT 2 HERE]

Exhibit 2 plots how the rejection threshold \( S_R \) increases with \( N \), and consequently \( DSR \) decreases. The investor has recognized that there is only a 90% chance that the true SR associated with this strategy is greater than zero. Should the strategist have made his discovery after running only \( N=46 \) independent trials, the investor may have allocated some funds, as \( DSR \) would have been 0.9505, above the 95% confidence level.

Non-Normality also played a role in discarding this investment offer. If the strategy had exhibited Normal returns \( (f_3 = 0, \hat{p}_d = 3) \), \( DSR = 0.9505 \) after \( N=88 \) independent trials. If non-Normal returns had not inflated the performance so much, the investor would have been willing to accept a much larger number of trials. This example illustrates that it is critical for investors to account for both sources of performance inflation jointly, as DSR does.

**WHEN SHOULD WE STOP TESTING?**

One important practical implication of this research is that multiple testing is a useful tool that should not be abused. Multiple testing exercises should be carefully planned in advance, so as to avoid running an unnecessarily large number of trials. Investment theory, not computational power, should motivate what experiments are worth conducting. This begs the question, what is the optimal number of trials that should be attempted?

An elegant answer to this critical question can be found in the theory of optimal stopping, more concretely the so-called "secretary problem", or 1/e-law of optimal choice, see Brass [1984]. There are many versions of this problem, but the key notion is that we wish to impose a cost to the number of trials conducted, because every additional trial irremediably increases the probability of a false positive.\(^1\)

In the context of our discussion, it translates as follows: From the set of strategy configurations that are theoretically justifiable, sample a fraction \( \frac{1}{e} \) of them (roughly 37%) at random and measure their performance. After that, keep drawing and measuring the performance of additional configurations from that set, one by one, until you find one that beats all of the

---

\(^1\) In the original "secretary problem" the decision-maker had no possibility of choosing a past candidate. This is not necessarily the case when selecting a strategy. However the key similarity is that in both cases the counter of trials cannot be turned back.
previous. That is the optimal number of trials, and that "best so far" strategy the one that should be selected.

This result provides a useful rule of thumb, with applications that go beyond the number of strategy configurations that should be backtested. It can be applied to situations where we test multiple alternatives with the goal of choosing a near-best as soon as possible, so as to minimize the chances of a false positive.

CONCLUSIONS
Machine learning, high-performance computing and related technologies have advanced many fields of the sciences. For example, the U.S. Department of Energy's SciDAC (Scientific Discovery through Advanced Computing) program uses terascale computers to "research problems that are insoluble by traditional theoretical and experimental approaches, hazardous to study in the laboratory, or time-consuming or expensive to solve by traditional means."²

Many of these techniques have become available to financial analysts, who use them to search for profitable investment strategies. Academic journals often publish backtests that report the performance of such simulated strategies. One problem with these innovations is that, unless strict scientific protocols are followed, there is a substantial risk of selecting and publishing false positives.

We have argued that selection bias is ubiquitous in the financial literature, where backtests are often published without reporting the full extent of the trials involved in selecting that particular strategy. To make matters worse, we know that backtest overfitting in the presence of memory effects leads to negative performance out-of-sample. Thus, selection bias combined with backtest overfitting misleads investors into allocating capital to strategies that will systematically lose money. The customary disclaimer that "past performance does not guarantee future results" is too lenient when in fact adverse outcomes are very likely.

In this paper we have proposed a test to determine whether an estimated SR is statistically significant after correcting for two leading sources of performance inflation: Selection bias and non-Normal returns. The Deflated Sharpe Ratio (DSR) incorporates information about the unselected trials, such as the number of independent experiments conducted and the variance of the SRs, as well as taking into account the sample length, skewness and kurtosis of the returns' distribution.

² http://outreach.scidac.gov/scidac-overview
APPENDICES

A.1. DERIVING THE EXPECTED MAXIMUM SHARE RATIO

We would like to derive the expected Sharpe Ratio after \( N \) independent trials. Consider a set \( \{ y_n \} \) of independent and identically distributed random variables drawn from a Normal Distribution, \( y_n \sim \mathcal{N} [\mu, \sigma^2] \), \( n = 1, \ldots, N \). We can build a standardized set \( \{ x_n \} \) by computing \( x_n = \frac{y_n - \mu}{\sigma} \), where \( x_n \sim \mathcal{N} [0,1] = Z \). The set \( \{ y_n \} \) is therefore identical to the set \( \{ \mu + \sigma x_n \} \). For \( \sigma > 0 \), the order of the set \( \{ \mu + \sigma x_n \} \) is unchanged, consequently

\[
\max \{ y_n \} = \max \{ \mu + \sigma x_n \} = \mu + \sigma \max \{ x_n \}
\]

where the same element is the maximum in both sets. Because the mathematical expectation operator, \( E[. ] \), is linear, we know that

\[
E[\max \{ y_n \}] = \mu + \sigma E[\max \{ x_n \}]
\]

Bailey et al. [2014a] prove that given a series of independent and identically distributed standard normal random variables, \( x_n \sim Z \), \( n = 1, \ldots, N \), the expected maximum of that series, \( E[\max x_n] = E[\max x_n] \), can be approximated for a large \( N \) as

\[
E[\max x_n] \approx (1 - \gamma)Z^{-1}\left[ 1 - \frac{1}{N} \right] + \gamma Z^{-1}\left[ 1 - \frac{1}{N} e^{-1} \right]
\]

where \( \gamma \) (approx. 0.5772) is the Euler-Mascheroni constant, \( e \) is Euler's number, and \( N \gg 1 \). Combining both results, we obtain that

\[
E[\max y_n] \approx \mu + \sigma \left( (1 - \gamma)Z^{-1}\left[ 1 - \frac{1}{N} \right] + \gamma Z^{-1}\left[ 1 - \frac{1}{N} e^{-1} \right] \right)
\]

A.2. EXPERIMENTAL VERIFICATION

We can evaluate experimentally the accuracy of the previous result. First, for a given set of parameters \( \{ E[5 \tilde{R}_n], V[5 \tilde{R}_n]/N \} \) we compute \( E[\max 5 \tilde{R}_n] \) analytically, i.e. applying Eq. (6). Second, we would like to compare that value with a numerical estimation. That numerical estimation is obtained by drawing \( Q \) random sets of \( \{ SR_n \} \) of size \( N \) from a Normal distribution with mean \( E[5 \tilde{R}_n] \) and variance \( V[5 \tilde{R}_n] \), computing the maximum of those sets, and estimating their mean, \( Q^{-1} \sum_{q=1}^{Q} \max \{ SR_n \}/q \). Third, we compute the estimation error as

\[
\varepsilon = E[\max \{ SR_n \}] - Q^{-1} \sum_{q=1}^{Q} \max \{ SR_n \}/q \]

where we expect \( \varepsilon \approx 0 \). Fourth, we can repeat the previous three steps for a wide variety of combinations of \( \{ E[5 \tilde{R}_n], V[5 \tilde{R}_n]/N \} \), and evaluate the magnitude and patterns of the resulting \( \varepsilon \) under various scenarios.

[EXHIBIT 3.1 HERE]
Exhibit 3.1 plots a heat map of ε values where $E[\{S_{R_n}\}] \in [-10, -10]$, $N \in [10,1000]$, $Q = 10^4$ and $V[\{S_{R_n}\}] = 1$. Exhibit 3.2 plots the analogous heat map, where we have set $V[\{S_{R_n}\}] = 4$. It is easy to verify that alternative values of $V[\{S_{R_n}\}] > 0$ generate similar outcomes. Snippet 1 implements in Python the experiment discussed earlier.

```
#!/usr/bin/env python
# On 20140607 by lopezdeprado@lbi.gov
import numpy as np, scipy.stats as ss, pandas as pd
from itertools import product

def getExpMaxSR(mu,sigma,numTrials):
    # Compute the expected maximum Sharpe ratio (Analytically)
    emc=0.5772156649 # Euler-Mascheroni constant
    maxZ=(1-emc)*ss.norm.pdf(1-1./numTrials)+emc*ss.norm.pdf(1-1./(numTrials*np.e))
    return mu+sigma*maxZ

def getDistMaxSR(mu,sigma,numTrials,numIters):
    # Compute the expected maximum Sharpe ratio (Numerically)
    maxSR,count=0,0
    while count<numIters:
        count+=1
        series=np.random.normal(mu,sigma,numTrials)
        maxSR=max(maxSR,max(series))
        return np.mean(maxSR),np.std(maxSR)

def simulate(mu,sigma,numTrials,numIters):
    #1) Compute Expected[Max[Sharpe ratio]] ANALYTICALLY
    expMaxSR=getExpMaxSR(mu,sigma,numTrials)
    #2) Compute Expected[Max[Sharpe ratio]] NUMERICALLY
    meanMaxSR, stdMaxSR = getDistMaxSR(mu,sigma,numTrials,numIters)
    return expMaxSR, meanMaxSR, stdMaxSR

def main():
    numIters,sigma,ouput,count=1e4,1.0
    for prod__ in product(np.linspace(-100,100,101),range(10,1001,10)):
        mu,numTrials=prod__,prod__
        expMaxSR,meanMaxSR, stdMaxSR=simulate(mu,sigma,numTrials,numIters)
        err=expMaxSR-meanMaxSR
        output.append({mu,sigma,numTrials,numIters,expMaxSR,meanMaxSR,\
            stdMaxSR, err})
    output.to_csv('DSR.csv')

if __name__ == '__main__':
    main()
```

Snippet 1 - Code for the experimental verification
A.3. ESTIMATING THE NUMBER OF INDEPENDENT TRIALS

It is critical to understand that the $N$ used to compute $E[\text{max}\{SR_n\}]$ corresponds to the number of *independent* trials. Suppose that we run $M$ trials, where only $\bar{N}$ trials are independent, $\bar{N}<M$. Clearly, using $M$ instead of $\bar{N}$ will overstate $E[\text{max}\{SR_n\}]$. So given $M$ dependent trials we need to derive the number of "implied independent trials", $\bar{N}$.

One path to accomplish that is by taking into account the average correlation between the trials, $p$. First, consider an $M\times M$ correlation matrix $C$ with real-valued entries $\{p_{i,j}\}$, where $i$ is the index for the rows and $j$ is the index for the columns. Let $\hat{C}$ be a modified correlation matrix, where all off-diagonal correlations have been replaced by a constant value $p$, i.e. $p_{i,j} = p, \forall i \neq j$. Then we define the weighted average correlation as the value $p$ such that

$$x'\hat{C}x = x'\begin{bmatrix} 1 & \rho_{1,j} & \rho_{1,M} \\ \rho_{i,1} & 1 & \rho_{i,M} \\ \rho_{M,1} & \rho_{M,j} & 1 \end{bmatrix} x = x'\begin{bmatrix} 1 & \rho & \rho \\ \rho & 1 & \rho \\ \rho & \rho & 1 \end{bmatrix} x = x'\hat{C}x \quad (7)$$

In words, we are interested in finding the constant value $p$ such that, if we make all off-diagonal correlations equal to $p$, the above quadratic form remains unchanged. For the case where $x$ equals a unit vector, $x = 1_M$, the quadratic form reduces to the sum of all the entries $\{p_{i,j}\}$, leading to the equal-weighted average correlation:

$$\rho = \frac{\sum_{i=1}^{M} \sum_{j=1}^{M} \rho_{i,j} - M}{M(M-1)} = \frac{2 \sum_{i=1}^{M} \sum_{j=i+1}^{M} \rho_{i,j}}{M(M-1)} \quad (8)$$

Second, a proper correlation matrix must be positive-definite, so it is guaranteed that all its quadratic forms are strictly positive, and in particular $I_M'\hat{C}I_M = I_M'CI_M > 0$. Then, $I_M'\hat{C}I_M = M + M(M-1)p$. The implication is that the average correlation is bounded by $p \in \left(\frac{1}{M-1}, 1\right]$, with $M > 1$ for a correlation to exist. The larger the number of trials, the more positive the average correlation is likely to be, and for a sufficiently large $M$ we have $\frac{1}{M-1} \approx 0 < p \leq 1$.

Third, we know that as $p \to 1$, then $N \to 1$. Similarly, as $p \to 0$, then $N \to M$. Given an estimated average correlation $\hat{p}$, we could therefore interpolate between these two extreme outcomes to obtain

$$\bar{N} = \hat{p} + (1 - \hat{p})M \quad (9)$$

Exhibit 4 plots the relationship between $\{M, \beta, N\}$. This method could be further enriched by incorporating Fisher's transform (see Fisher [1915]), thus controlling for the variance of the error in the estimation of $\hat{p}$.

[EXHIBIT 4 HERE]
This and other "average correlation" approaches are intuitive and convenient in practical applications. However, two problematic aspects should be highlighted in connection with "average correlation" formulations: First, correlation is a limited notion of linear dependence. Second, in practice $M$ almost always exceeds the sample length, $T$. Then the estimate of average correlation may itself be overfit. In general for short samples $T < \frac{1}{2} M(M-1)$, the correlation matrix will be numerically ill-conditioned, and it is not guaranteed that $\sum M_i < 0$. Estimating an average correlation is then pointless, because there are more correlations $\{p_{i,j}|i \neq j, i = 1, \ldots M\}$ than independent pairs of observations! One way to deal with this numerical problem is to reduce the dimension of the correlation matrix (see Bailey and Lopez de Prado [2012b] for one such algorithm), and compute the average correlation on that reduced definite-positive matrix.

An alternative and more direct path is to use information theory to determine $N$. Entropy relates to a much deeper concept of redundancy than correlation. We refer the interested reader to the literature on data compression, total correlation and multiinformation, e.g. Watabane [1960] and Studeny and Vejnarova [1999]. These and other standard information theory methods produce accurate estimates of the number $N$ of non-redundant sources among a set of $M$ random variables.
Exhibit 1 - Expected Maximum Sharpe Ratio as the number of independent trials grows, for $E[\{SR_n\}] = 0$ and $V[\{SR_n\}] \in \{1,4\}$
Exhibit 2 - Expected Maximum Sharpe Ratio (left-y-axis) and Deflated Sharpe Ratio (right-y-axis) as the number of independent trials grows
Exhibit 3.1 - Experimental verification of the analytical formula for estimating the expected maximum of \( \{SR_n\} \), where \( \text{V}[[\tilde{S}R_n]] = 1 \).

This heat map plots the difference between the expected maximum of \( \{SR_n\} \) estimated analytically and the average of maximum \( \{SR_n\} \) computed empirically, for various combinations of \( \mu \) and \( N \). We have set \( \text{V}[[\tilde{S}R_n]] = 1 \). When the number of trials is very small (e.g., less than 50), the analytical result sometimes overestimates the empirical result, by a very small margin (less than 0.05 for an underlying process with variance 1). As the number of trials increases, that level of overestimation converges to zero (by 1000 trials it is as small as 0.006 for a process with variance 1). This is consistent with our proof, where we required that \( N \gg 1 \).
Exhibit 3.2 - Experimental verification of the analytical formula for estimating the expected maximum of $\{SR_n\}$, where $V[\{SR_n\}] = 4$

An analogous result is obtained for alternative values of $V[\{5R_n\}]$. Now the maximum error is approx. 0.11, for an underlying process with variance $V[\{5R_n\}] = 4$. This is about double the maximum error observed in Exhibit 2.1, and consistent with the $\sigma = \sqrt{V[\{SR_n\}]}$ scaling in Eq.(6). The error quickly converges to zero as the number of trials grows, $N \gg 1$. 
Exhibit 4 - Implied number of independent trials ($\bar{N}$) for various values of average correlation ($p$) and number of trials ($M$)
REFERENCES


THE PROBABILITY OF BACKTEST OVERFITTING

David H. Bailey * Jonathan M. Borwein †
Marcos Lopez de Prado ‡ Qiji Jim Zhu §

February 27, 2015

Revised version: February 2015

*Lawrence Berkeley National Laboratory (retired), 1 Cyclotron Road, Berkeley, CA 94720, USA, and Research Fellow at the University of California, Davis, Department of Computer Science. E-mail: david@davidhbailey.com; URL: http://www.davidbailey.com

† Laureate Professor of Mathematics at University of Newcastle, Callaghan NSW 2308, Australia, and a Fellow of the Royal Society of Canada, the Australian Academy of Science, the American Mathematical Society and the AAAS. E-mail: jonathan.borwein@newcastle.edu.au; URL: http://www.carma.newcastle.edu.au/jon

‡ Senior Managing Director at Guggenheim Partners, New York, NY 10017, and Research Affiliate at Lawrence Berkeley National Laboratory, Berkeley, CA 94720, USA. E-mail: lopedz@prado@lbl.gov; URL: http://www.QuantRSSearch.info

§ Professor, Department of Mathematics, Western Michigan University, Kalamazoo, MI 49008, USA. Email: zhu@wmich.edu; URL: http://homepages.wmich.edu/~zhu/
THE PROBABILITY OF BACKTEST OVERFITTING

Abstract

Many investment firms and portfolio managers rely on backtests (i.e., simulations of performance based on historical market data) to select investment strategies and allocate capital. Standard statistical techniques designed to prevent regression overfitting, such as hold-out, tend to be unreliable and inaccurate in the context of investment backtests. We propose a general framework to assess the probability of backtest overfitting (PBO). We illustrate this framework with specific generic, model-free and nonparametric implementations in the context of investment simulations, which implementations we call combinatorially symmetric cross-validation (CSCV). We show that CSCV produces reasonable estimates of PBO for several useful examples.

Keywords. Backtest, historical simulation, probability of backtest overfitting, investment strategy, optimization, Sharpe ratio, minimum backtest length, performance degradation.

JEL Classification: GO, Gl, G2, G15, G24, E44.

AMS Classification: 91G10, 91G60, 91G70, 62C, 60E.

Acknowledgements. We are indebted to the Editor and three anonymous referees who peer-reviewed this article as well as a related article for the Notices of the American Mathematical Society [1]. We are also grateful to Tony Anagnostakis (Moore Capital), Marco Avellaneda (Courant Institute, NYU), Peter Carr (Morgan Stanley, NYU), Paul Embrechts (ETH Ziirich), Matthew D. Foreman (University of California, Irvine), Ross Garon (SAC Capital), Jeffrey S. Lange (Guggenheim Partners), Attilio Meucci (KKR, NYU), Natalia Nolde (University of British Columbia and ETH Ziirich) and Riccardo Rebonato (PIMCO, University of Oxford) for many useful and stimulating exchanges.

Sponsorship. Research supported in part by the Director, Office of Computational and Technology Research, Division of Mathematical, Information, and Computational Sciences of the U.S. Department of Energy, under contract number DE-AC02-05CH11231 and by various Australian Research Council grants.
“This was our paradox: No course of action could be determined by a rule, because every course of action can be made to accord with the rule.” Ludwig Wittgenstein [36].

1 INTRODUCTION

Modern investment strategies rely on the discovery of patterns that can be quantified and monetized in a systematic way. For example, algorithms can be designed to profit from phenomena such as “momentum,” i.e., the tendency of many securities to exhibit long runs of profits or losses, beyond what could be expected from securities following a martingale. One advantage of this systematization of investment strategies is that those algorithms are amenable to “backtesting.” A backtest is a historical simulation of how an algorithmic strategy would have performed in the past. Backtests are valuable tools because they allow researchers to evaluate the risk/reward profile of an investment strategy before committing funds.

Recent advances in algorithmic research and high-performance computing have made it nearly trivial to test millions and billions of alternative investment strategies on a finite dataset of financial time series. While these advances are undoubtedly useful, they also present a negative and often silenced side-effect: The alarming rise of false positives in related academic publications (The Economist [32]). This paper introduces a computational procedure for detecting false positives in the context of investment strategy research.

To motivate our study, consider a researcher who is investigating an algorithm to profit from momentum. Perhaps the most popular technique among Commodity Trading Advisors (CTAs) is to use so-called crossing-moving averages to detect a change of trend in a security. Even for the simplest case, there are at least five parameters that the researcher can fit: Two sample lengths for the moving averages, entry threshold, exit threshold and stop-loss. The number of combinations that can be tested over thousands of securities is in the billions. For each of those billions of backtests, we could estimate its Sharpe ratio (or any other performance statistic), and determine whether that Sharpe ratio is indeed statistically significant at a confidence level of 95%. Although this approach is consistent with the Neyman-Pearson framework of hypothesis testing, it is highly likely that false positives will emerge with a probability greater than 5%. The reason

---

1Several technical tools are based on this principle, such as the Moving Average Convergence Divergence (MACD) indicator.
is that a 5% false positive probability only holds when we apply the test exactly once. However, we are applying the test on the same data multiple times (indeed, billions of times), making the emergence of false positives almost certain.

The core question we are asking is this: *What constitutes a legitimate empirical finding in the context of investment research?* This may appear to be a rather philosophical question, but it has important practical implications, as we shall see later in our discussion. Financial discoveries typically involve identifying a phenomenon with low signal-to-noise ratio, where that ratio is driven down as a result of competition. Because the signal is weak, a test of hypothesis must be conducted on a large sample as a way of assessing the existence of a phenomenon. This is not the typical case in scientific areas where the signal-to-noise ratio is high. By way of example, consider the apparatus of classical mechanics, which was developed centuries before Neyman and Pearson proposed their theory of hypothesis testing. Newton did not require statistical testing of his gravitation theory, because the signal from that phenomenon dominates the noise.

The question of 'legitimate empirical findings' is particularly troubling when researchers conduct multiple tests. The probability of finding false positives increases with the number of tests conducted on the same data (Miller [25]). As each researcher carries out millions of regressions (Sala-i-Martin [28]) on a finite number of independent datasets without controlling for the increased probability of false positives, some researchers have concluded that 'most published research findings are false' (see Ioannidis [17]).

Furthermore, it is common practice to use this computational power to *calibrate* the parameters of an investment strategy in order to maximize its performance. But because the signal-to-noise ratio is so weak, often the result of such calibration is that parameters are chosen to profit from past noise rather than future signal. The outcome is an overfit backtest [1]. Scientists at Lawrence Berkeley National Laboratory have developed an online tool to demonstrate this phenomenon. This tool generates a time series of pseudorandom returns, and then calibrates the parameters of an optimal monthly strategy (i.e., the sequence of days of the month to be long the security, and the sequence of days of the month to be short). After a few hundred iterations, it is trivial to find highly profitable strategies in-sample, despite the small number of parameters involved. Performance out-of-sample is, of course, utterly disappointing. The tool is available at [http://datagrid.lbl.gov/backtest/ index.php](http://datagrid.lbl.gov/backtest/index.php).

Backtests published in academic or practitioners’ publications almost never declare the number of trials involved in a discovery. Because those
researchers have most likely not controlled for the number of trials, it is highly probable that their findings constitute false positives ([1, 3]). Even though researchers at academic and investment institutions may be aware of these problems, they have little incentive to expose them. Whether their motivations are to receive tenure or raise funds for a new systematic fund, those researchers would rather ignore this problem and make their investors or managers believe that backtest overfitting does not affect their results. Some may even pretend that they are controlling for overfitting using inappropriate techniques, exploiting the ignorance of their sponsors, as we will see later on when discussing the 'hold-out' method.

The goal of our paper is to develop computational techniques to control for the increased probability of false positives as the number of trials increases, applied to the particular field of investment strategy research. For instance, journal editors and investors could demand researchers to estimate that probability when a backtest is submitted to them.

Our approach. First, we introduce a precise characterization of the event of backtest overfitting. The idea is simple and intuitive: For overfitting to occur, the strategy configuration that delivers maximum performance in sample (IS) must systematically underperform the remaining configurations out of sample (OOS). Typically the principal reason for this underperformance is that the IS "optimal" strategy is so closely tied to the noise contained in the training set that further optimization of the strategy becomes pointless or even detrimental for the purpose of extracting the signal.

Second, we establish a general framework for assessing the probability of the event of backtest overfitting. We model this phenomenon of backtest overfitting using an abstract probability space in which the sample space consist of pairs of IS and OOS test results.

Third, we set as null hypothesis that backtest overfitting has indeed taken place, and develop an algorithm that tests for this hypothesis. For a given strategy, the probability of backtest overfitting (PBO) is then evaluated as the conditional probability that this strategy underperforms the median OOS while remaining optimal IS. While the PBO provides a direct way to quantify the likelihood of backtest overfitting, the general framework also affords us information to look into the overfitting issue from different perspectives. For example, besides PBO, this framework can also be used to assess performance decay, probability of loss, and possible stochastic dominance of a strategy.

It is worth clarifying in what sense do we speak of a probability of back-
test overfitting. Backtest overfitting is a deterministic fact (either the model is overfit or it is not), hence it may seem unnatural to associate a probability to a non-random event. Given some empirical evidence and priors, we can infer the posterior probability that overfitting has taken place. Examples of this line of reasoning abound in information theory and machine learning treatises, e.g. [23]. It is in this Bayesian sense that we define and estimate PBO.

A generic, model-free, and nonparametric testing algorithm is desirable, since backtests are applied to trading strategies produced using a great variety of different methods and models. For this reason, we present a specific implementation, which we call a \textit{combinatorially symmetric cross-validation} (CSCV). We show that CSCV produces reasonable estimates of PBO for several useful examples.

Our CSCV implementation draws from elements in experimental mathematics, information theory, Bayesian inference, machine learning and decision theory to address the very particular problem of assessing the representativeness of a backtest. This is not an easy problem, as evidenced by the scarcity of academic papers addressing a dilemma that most investors face. This gap in the literature is disturbing, given the heavy reliance on backtests among practitioners. One advantage of our solution is that it only requires time series of backtested performance. We avoid the credibility issue of preserving a truly out-of-sample test-set by not requiring a fixed "hold-out," and swapping all in-sample (IS) and out-of-sample (OOS) datasets. Our approach is generic in the sense of not requiring knowledge of either the trading rule or forecasting equation. The output is a bootstrapped distribution of OOS performance measure. Although in our examples we measure performance using the Sharpe ratio, our methodology does not rely on this particular performance statistic, and it can be applied to any alternative preferred by the reader.

We emphasize that the CSCV implementation is only one illustrative technique. The general framework is flexible enough to accommodate other task-specific methods for estimating the PBO.

\textbf{Comparisons to other approaches.} Perhaps the most common approach to prevent overfitting among practitioners is to require the researcher to withhold a portion of the available data sample for separate testing and validation as OOS performance (this is known as the "hold-out" or "test set" method). If the IS and OOS performance levels are congruent, the investor might decide to "reject" the hypothesis that the backtest is overfit.
The main advantage of this procedure is its simplicity. This approach is, however, unsatisfactory for multiple reasons.

First, if the data is publicly available, it is quite likely that the researcher has used the "hold-out" as part of the IS dataset. Second, even if no "hold-out" data was used, any seasoned researcher knows well how financial variables performed over the time period covered by the OOS dataset, and that information may well be used in the strategy design, consciously or not (see Schorfheide and Wolpin [29]).

Third, hold-out is clearly inadequate for small samples—the IS dataset will be too short to fit, and the OOS dataset too short to conclude anything with sufficient confidence. Weiss and Kulikowski [34] argue that hold-out should not be applied to an analysis with less than 1,000 observations. For example, if a strategy trades on a weekly basis, hold-out should not be used on backtests of less than 20 years. Along the same lines, Van Belle and Kerr [33] point out the high variance of hold-out estimation errors. If one is unlucky, the chosen hold-out section may be the one that refutes a valid strategy or supports an invalid strategy. Different hold-outs are thus likely to lead to different conclusions.

Fourth, even if the researcher works with a large sample, the OOS analysis will need to consume a large proportion of the sample to be conclusive, which is detrimental to the strategy's design (see Hawkins [15]). If the OOS is taken from the end of a time series, we are losing the most recent observations, which often are the most representative going forward. If the OOS is taken from the beginning of the time series, the testing has been done on arguably the least representative portion of the data.

Fifth, as long as the researcher tries more than one strategy configuration, overfitting is always present (see Bailey et al. [1] for a proof). The hold-out method does not take into account the number of trials attempted before selecting a particular strategy configuration, and consequently hold-out cannot correctly assess a backtest's representativeness.

In short, the hold-out method leaves the investor guessing to what degree the backtest is overfit. The answer to the question "is this backtest overfit?" is not a true-or-false, but a non-null probability that depends on the number of trials involved (input ignored by hold-out). In this paper we will present a way to compute this probability.

Another approach popular among practitioners consists in modeling the underlying financial variable by generating pseudorandom scenarios and measuring the performance of the resulting investment strategy for those scenarios (see Carr and Lopez de Prado [6] for a valid application of this technique). This approach has the advantage of generating a distribution of
outcomes, rather than relying on a single OOS performance estimate, as the "hold-out" method does. The disadvantages are that the model that generates random series of the underlying variable may also be overfit, or may not contain all relevant statistical features, and may need to be customized to every variable (with large development costs). Some retail trading platforms offer backtesting procedures based on this approach, such as by pseudorandom generation of tick data by fractal interpolation.

Several procedures have been proposed to determine whether an econometric model is overfit. See White [35], Romano et al. [27], Harvey et al. [13] for a discussion in the context of Econometric models. Essentially these methods propose a way to adjust the $p$-values of estimated regression coefficients to account for the multiplicity of trials. These are valuable approaches when the trading rule relies on an econometric specification. That is not generally the case, as discussed in Bailey et al. [1]. Investment strategies in general are not amenable to characterization through a system of algebraic equations. Regression-tree decision making, for example, requires a hierarchy that only combinatorial frameworks like graph theory can provide, and which are beyond the geometric arguments used in econometric models (see Calkin and Lopez de Prado [4, 5]). On the other hand, the approach proposed here shares the same philosophy in that both are trying to assess the probability of overfitting.

**Structure of the paper.** The rest of the study is organized as follows: Section 2 sets the foundations of our framework: we describe our general framework for the backtest overfitting probability in Subsection 2.1 and present the CSCV method for estimate this probability in Subsection 2.2. Section 3 discusses other ways that our general framework can be used to assess a backtest. Section 4 further discusses some of the features of the CSCV method, and how it relates to other machine learning methods. Section 5 lists some of the limitations of this method. Section 6 discusses a practical application, and Section 7 summarizes our conclusions. We have carried out several test cases to illustrate how the PBO compares to different scenarios, and to assess the accuracy of our method using two alternative approaches (Monte Carlo Methods and Extreme Value Theory). The interested reader can find the details of those studies following this link: [http://ssrn.com/abstract=2568435](http://ssrn.com/abstract=2568435)
2 THE FRAMEWORK

2.1 DEFINITION OF OVERFITTING IN THE CONTEXT OF STRATEGY SELECTION

We first establish a measure theoretic framework in which the probability of backtest overfitting and other statistics related to the issue of overfitting can be rigorously defined. Consider a probability space \((\Omega, \mathcal{F}, \operatorname{Prob})\) where \(T\) represents a sample space of pairs of IS and OOS samples. We aim at estimating the probability of overfitting for the following backtest strategy selection process: select from \(N\) strategies labeled as \((1, 2, \ldots, N)\) the 'best' one using backtesting according to a given performance measure, say, the Sharpe ratio. Fixing a performance measure, we will use random vectors \(R = (i, i^\prime, \ldots, RN)\) and \(\overline{R} = (\overline{i}, \overline{i}^\prime, \ldots, R\overline{N})\) to represent the IS and OOS performance of the \(N\) strategies, respectively. For a given sample \(c \in T\), that is a concrete pair of IS and OOS samples, we will use \(R^c\) and \(\overline{R}^c\) to signify the performances of the \(N\) strategies on the IS and OOS pair given by \(c\). For most applications \(T\) will be finite and one can choose to use the power set \(\mathcal{T}\) as \(\mathcal{F}\). Moreover, often it makes sense in this case to assume that the \(\operatorname{Prob}\) is uniform on elements in \(\mathcal{T}\). However, we do not make specific assumptions at this stage of general discussion so as to allow flexibility in particular applications.

The key observation here is to compare the ranking of the selected strategies IS and OOS. Therefore we consider the ranking space \(\Omega\) consists of the \(N!\) permutations of \((1, 2, \ldots, N)\) indicating the ranking of the \(N\) strategies. Then we use random vectors \(r, \overline{r}\) to represent the ranking of the components of \(R, \overline{R}\), respectively. For example, if \(N = 3\) and the performance measure is the Sharpe ratio, for a particular sample \(c \in T\), \(R = (0.5, 1.1, 0.7)\) and \(\overline{R} = (0.6, 0.7, 1.3)\), then we have \(r^c = (1, 3, 2)\) and \(\overline{r}^c = (1, 2, 3)\). Thus, both \(r\) and \(\overline{r}\) are random vectors mapping \((\Omega, \mathcal{F}, \operatorname{Prob})\) to \(\Omega\).

Now, we define backtest overfitting, in the context of investment strategy selection alluded to above. We will need to use the following subset of \(\Omega\): \(\Omega_n = \{1 \notin \Omega^c / \Omega, 1/n = N\}\).

**Definition 2.1.** (Backtest Overfitting) We say that the backtest strategy selection process overfits if a strategy with optimal performance IS has an expected ranking below the median OOS. By the Bayesian formula and using the notation above that is

\[
\sum_{n=0}^{N} E[\overline{r}_n] | r \in \Omega_n^* | \operatorname{Prob}[r \in \Omega_n^*] \leq N/2.
\]  

(2.1)
Definition 2.2. (Probability of Backtest Overfitting) A strategy with optimal performance is not necessarily optimal OOS. Moreover, there is a non-null probability that this strategy with optimal performance ranks below the median OOS. This is what we define as the probability of backtest overfit (PBO). More precisely,

\[ PBO = \sum_{n=1}^{N} \text{Prob} \left[ r_{n} < N/2 \mid r \in \Omega_{n}^{*} \right] \text{Prob} \left[ r \in \Omega_{n}^{*} \right]. \]  \hspace{1cm} (2.2)

In other words, we say that a strategy selection process overfits if the expected performance of the strategies selected is less than the median performance rank OOS of all strategies. In that situation, the strategy selection process becomes in fact detrimental. Note that in this context IS corresponds to the subset of observations used to select the optimal strategy among the \( N \) alternatives. With IS we do not mean the period on which the investment model underlying the strategy was estimated (e.g., the period on which crossing moving averages are computed, or a forecasting regression model is estimated). Consequently, in the above definition we refer to overfitting in relation to the strategy selection process, not a strategy's model calibration (e.g., in the context of regressions). That is the reason we were able to define overfitting without knowledge of the strategy's underlying models, i.e., in a model-free and non-parametric manner.

2.2 THE CSCV PROCEDURE

The framework Subsection 2.1 is flexible in dealing with the probability of backtest overfitting and other statistical characterizations related to the issue of overfitting. However, in order to quantify say the PBO for concrete applications we need a method to estimate the probability that was abstractly defined in the previous section. Estimating the probability in a particular application relies on schemes for selecting samples of IS and OOS pairs. This section is devoted to establishing such a procedure, which we name combinatorially symmetric cross-validation, abbreviated as (CSCV) for convenience of reference.

Suppose that a researcher is developing an investment strategy. She considers a family of system specifications and parametric values to be backtested, in an attempt to uncover the most profitable incarnation of that idea. For example, in a trend-following moving average strategy, the researcher might try alternative sample lengths on which the moving averages are computed, entry thresholds, exit thresholds, stop losses, holding periods,
sampling frequencies, and so on. As a result, the researcher ends up running a number \( N \) of alternative model configurations (or trials), out of which one is chosen according to some performance evaluation criterion, such as the Sharpe ratio.

**Algorithm 2.3 (CSCV).** We proceed as follows.

**First,** we form a matrix \( \mathbf{M} \) by collecting the performance series from the \( N \) trials. In particular, each column \( n = 1, \ldots, N \) represents a vector of profits and losses over \( t = 1, \ldots, T \) observations associated with a particular model configuration tried by the researcher. \( \mathbf{M} \) is therefore a real-valued matrix of order \((\Gamma \times N)\). The only conditions we impose are that:

i) \( \mathbf{M} \) is a true matrix, i.e. with the same number of rows for each column, where observations are synchronous for every row across the \( N \) trials, and

ii) the performance evaluation metric used to choose the "optimal" strategy can be estimated on subsamples of each column.

For example, if that metric was the Sharpe ratio, we would expect that the IID Normal distribution assumption could be maintained on various slices of the reported performance. If different model configurations trade with different frequencies, observations should be aggregated to match a common index \( t = 1, \ldots, T \).

**Second,** we partition \( \mathbf{M} \) across rows, into an even number \( S \) of disjoint submatrices of equal dimensions. Each of these submatrices \( \mathbf{M}_s \), with \( s = 1, \ldots, S \), is of order \((T/S \times N)\).

**Third,** we form all combinations \( C_S \) of \( \mathbf{M}_s \), taken in groups of size \( S/2 \). This gives a total number of combinations

\[
\binom{S}{S/2} = \frac{S}{S/2-1} \frac{S}{S/2-2} \cdots \frac{S}{2} = \prod_{i=0}^{S/2-1} \frac{S - i}{S/2 - i}
\]  

(2.3)

For instance, if \( S = 16 \), we will form 12,780 combinations. Each combination \( c \in C_S \) is composed of \( S/2 \) submatrices \( \mathbf{M}_s \).

**Fourth,** for each combination \( c \in C_S \), we:

a) Form the training set \( J \), by joining the \( S/2 \) submatrices \( \mathbf{M}_s \) that constitute \( c \) in their original order. \( J \) is a matrix of order \((T/S) \times (S/2 \times N) = T/2 \times N\).
b) Form the testing set $\bar{J}$, as the complement of $J$ in $M$. In other words, $\bar{J}$ is the $T/2 \times N$ matrix formed by all rows of $M$ that are not part of $J$ also in their original order. (The order in forming $J$ and $\bar{J}$ does not matter for some performance measures such as the Sharpe ratio but does matter for others e.g. return maximum drawdown ratio).

c) Form a vector $\mathbf{R}^c$ of performance statistics of order $N$, where the $n$th component $R_{n}^c$ of $\mathbf{R}^c$ reports the performance associated with the $n$th column of $J$ (the testing set). As before rank of the components of $\mathbf{R}^c$ is denoted by $\rho^c$ the IS ranking of the $N$ strategies.

d) Repeat c) with $J$ replaced by $\bar{J}$ (the test set) to derive $\mathbf{R}^{\bar{c}}$ and $\rho^{\bar{c}}$ the OOS performance statistics and rank of the $N$ strategies, respectively.

e) Determine the element $n^*$ such that $R_{n^*}^c \in \Omega_{n^*}$. In other words, $n^*$ is the best performing strategy IS.

f) Define the relative rank of $R_{n^*}^c$, by $\tilde{\rho}_{n^*} := R_{n^*}^c/(N + 1) \in (0, 1)$. This is the relative rank of the OOS performance associated with the strategy chosen IS. If the strategy optimization procedure is not overfitting, we should observe that $R_{n^*}^c$ systematically outperforms OOS, just as $R_{n^*}^c$ outperformed IS.

g) We define the logit $A_{c} = \ln \frac{\tilde{\rho}_{n^*}}{1 - \tilde{\rho}_{n^*}}$. High logit values imply a consistency between IS and OOS performances, which indicates a low level of backtest overfitting.

**Fifth**, we compute the distribution of ranks OOS by collecting all the $\chi_c$, for $c \in C_s$. Define the relative frequency at which $A$ occurred across all $C_s$ by

$$f(\lambda) = \sum_{c \in C_s} \frac{\chi(A_c)}{(\#CS)}$$  \hspace{1cm} (2.4)

where $\chi$ is the characterization function and $\#(\emptyset)$ signifies the number of elements in $C_s$. Then $\int_{-\infty}^{\infty} f(A)dA = 1$. This concludes the procedure.

Figure 1 schematically represents how the combinations in $C_s$ are used to produce training and testing sets, where $S = 4$. It shows the six combinations of four subsamples A, B, C, D, grouped in two subsets of size two. The first subset is the training set (or in-sample). This is used to determine the optimal model configuration. The second subset is the testing set (or out-of-sample), on which the in-sample optimal model configuration is
Figure 1: Generating the Cs symmetric combination.

tested. Running the N model configurations over each of these combinations allows us to derive a relative ranking, expressed as a logit. The outcome is a distribution of logits, one per combination. Note that each training subset combination is re-used as a testing subset and vice-versa (as is possible because we split the data in two equal parts).

3 OVERFIT STATISTICS

The framework introduced in Section 2 allows us to characterize the reliability of a strategy’s backtest in terms of four complementary analysis:

1. Probability of Backtest Overfitting (PBO): The probability that the model configuration selected as optimal IS will underperform the median of the N model configurations OOS.

2. Performance degradation: This determines to what extent greater performance IS leads to lower performance OOS, an occurrence associated with the memory effects discussed in Bailey et al. [1].

3. Probability of loss: The probability that the model selected as optimal IS will deliver a loss OOS.

4. Stochastic dominance: This analysis determines whether the procedure used to select a strategy IS is preferable to randomly choosing one model configuration among the N alternatives.

3.1 PROBABILITY OF BACKTEST OVERFITTING (PBO)

The PBO defined in Section 2.1 may now be estimated using the CSCV method with \( \phi = \int_{-\infty}^{\infty} f(X) dX \). This represents the rate at which optimal IS strategies underperform the median of the OOS trials. The analogue of \( \tilde{r} \) in
medical research is the placebo given to a portion of patients in the test set. If the backtest is truly helpful, the optimal strategy selected IS should outperform most of the $N$ trials OOS. That is the case when $\lambda_0 > 0$. For $\phi \approx 0$, a low proportion of the optimal IS strategy outperformed the median of trials in most of the testing sets indicating no significant overfitting. On the flip side, $\phi \approx I$ indicates high likelihood of overfitting. We consider at least three uses for PBO: i) In general the value of $\phi$ provides us a quantitative sense about the likelihood of overfitting. In accordance with standard applications of the Neyman-Pearson framework, a customary approach would be to reject models for which PBO is estimated to be greater than 0.05. ii) PBO could be used as a prior probability in Bayesian applications, where for instance the goal may be to derive the posterior probability of a model's forecast. iii) We could compute the PBO on a large number of investment strategies, and use those PBO estimates to compute a weighted portfolio, where the weights are given by $(1-PBO) / PBO$ or some other scheme.

3.2 PERFORMANCE DEGRADATION AND PROBABILITY OF LOSS

Section 2.2 introduced the procedure to compute, among other results, the pair $(R_{\eta^*}, R_{\eta^-})$ for each combination $\eta \in \Theta$. Note that while we know that $R_{\eta^*}$ is the maximum among the components of $R$, $R_{\eta^-}$ is not necessarily the maximum among the components of $R$. Because we are trying every combination of $M_\eta$ taken in groups of size $S/2$, there is no reason to expect the distribution of $R$ to dominate over $\overline{R}$. The implication is that, generally, $\overline{R}_{\eta^-} < \max\{\overline{R}\} \approx \max\{R\} = R_{\eta^*}$. For a regression $\overline{R}_{\eta^-} = a + \beta \eta^* + \epsilon^\prime$, the $\beta$ will be negative in most practical cases, due to compensation effects described in Bailey et al. [1]. An intuitive explanation for this negative slope is that overfit backtests minimize future performance: The model is so fit to past noise, that it is often rendered unfit for future signal. And the more overfit a backtest is, the more memory is accumulated against its future performance.

It is interesting to plot the pairs $(R_{\eta^*}, R_{\eta^-})$ to visualize how strong is such performance degradation, and to obtain a more realistic range of attainable performance OOS (see Figure 8). A particularly useful statistic is the proportion of combinations with negative performance, $\text{Prob}[\overline{R}_{\eta^-} < 0]$. Note that, even if $\phi \approx 0$, $\text{Prob}[\overline{R}_{\eta^-} < 0]$ could be high, in which case the strategy's performance OOS is probably poor for reasons other than overfitting.

Figure 2 provides a graphical representation of i) Out-Of-Sample Performance Degradation, ii) Out-Of-Sample Probability of Loss, and iii) Proba-
Figure 2: Performance degradation and distribution of logits. Note that even if $\phi \approx 0$, $\text{Prob}[R_{n*} < 0]$ could be high, in which case the strategy's performance OOS is poor for reasons other than overfitting.
bility of Overrating (PBO).

The upper plot of Figure 2 shows that pairs of (SR IS, SR OOS) for the optimal model configurations selected for each subset \( c \in \mathcal{C}^4 \), which corresponds to the performance degradation associated with the backtest of an investment strategy. We can once again appreciate the negative relationship between greater SR IS and SR OOS, indicating that at some point seeking the optimal performance becomes detrimental. Whereas 100% of the SR IS are positive, about 78% of the SR OOS are negative. Also, Sharpe ratios IS range between 1 and 3, indicating that backtests with high Sharpe ratios tell us nothing regarding the representativeness of that result.

We cannot hope escaping the risk of overfitting by exceeding some SR IS threshold. On the contrary, it appears that the higher the SR IS, the lower the SR OOS. In this example we are evaluating performance using the Sharpe ratio, however, we again stress that our procedure is generic and can be applied to any performance evaluation metric \( \mathbf{R} \) (Sortino ratio, Jensen's Alpha, Probabilistic Sharpe Ratio, etc.). The method also allows us to compute the proportion of combinations with negative performance, \( \text{Prob}[R_{n^*,i} < 0] \), which corresponds to analysis ii).

The lower plot of Figure 2 shows the distribution of logits for the same strategy, with a PBO of 74%. It displays the distribution of logits, which allows us to compute the probability of backtest overfitting (PBO). This represents the rate at which optimal IS strategies underperform the median of the OOS trials.

Figure 3 plots the performance degradation and distribution of logits of a real investment strategy. Unlike in the previous example, the OOS probability of loss is very small (about 3%), and the proportion of selected (IS) model configurations that performed OOS below the median of overall model configurations was only 4%.

The upper plot of Figure 3 plots the performance degradation associated with the backtest of a real investment strategy. The regression line that goes through the pairs of (SR IS, SR OOS) is much less steep, and only 3% of the SR OOS are negative. The lower plot of Figure 3 shows the distribution of logits, with a PBO of 0.04%. According to this analysis, it is unlikely that this backtest is significantly overfit. The chances that this strategy performs well OOS are much greater than in the previous example.

3.3 STOCHASTIC DOMINANCE

A further application of the results derived in Section 2.2 is to determine whether the distribution of \( \overline{R_{n^*}} \) across all \( c \in \mathcal{C} \) stochastically dominates
Figure 3: Performance degradation and distribution of logits for a real investment strategy.
over the distribution of all $\overline{R}$. Should that not be the case, it would present strong evidence that strategy selection optimization does not provide consistently better OOS results than a random strategy selection. One reason that the concept of stochastic dominance is useful is that it allows us to rank gambles or lotteries without having to make strong assumptions regarding an individual’s utility function. See Hadar and Russell [11] for an introduction to these matters.

In the context of our framework, first-order stochastic dominance occurs if $\text{Prob}[\overline{R}_n^* \geq x] \geq \text{Prob}[\text{Mean} (\overline{R}) \geq x]$ for all $x$, and for some $x$, $\text{Prob}[\overline{R}_n^* \geq x] > \text{Prob}[\text{Mean} (\overline{R}) \geq x]$. It can be verified visually by checking that the cumulative distribution function of $\overline{R}_n^*$ is not above the cumulative distribution function of $\overline{R}$ for all possible outcomes, and at least for one outcome the former is strictly below the latter. Under such circumstances, the decision maker would prefer the criterion used to produce $\overline{R}_n^*$ over a random sampling of $\overline{R}$, assuming only that her utility function is weakly increasing.

A less demanding criterion is second-order stochastic dominance. This requires that $\text{Prob}[\overline{R}_n^* \geq x] > \text{Prob}[\text{Mean} (\overline{R}) \geq x]$ for all $x$, and that $\text{Prob}[\overline{R}_n^* \geq x] > 0$ at some $x$. When that is the case, the decision maker would prefer the criterion used to produce $\overline{R}_n^*$ over a random sampling of $\overline{R}$, as long as she is risk averse and her utility function is weakly increasing.

Figure 4 complements the analysis presented in Figure 2, with analysis of stochastic dominance. Stochastic dominance allows us to rank gambles or lotteries without having to make strong assumptions regarding an individual’s utility function.

Figure 4 also provides an example of the cumulative distribution function of $\overline{R}_n^*$ across all $c \in C_s$ (red line) and $\overline{R}$ (blue line), as well as the second order stochastic dominance ($SD2[x]$) for every OOS SR. In this example, the distribution of OOS SR of optimized (IS) model configurations does not dominate (to first order) the distribution of OOS SR of overall model configurations.

This can be seen in the fact that for every level of OOS SR, the proportion of optimized model configurations is greater than the proportion of non-optimized, thus the probabilistic mass of the former is shifted to the left of the non-optimized. SD2 plots the second order stochastic dominance, which indicates that the distribution of optimized model configurations does not dominate the non-optimized even according to this less demanding criterion. It has been computed on the same backtest used for Figure 2. Consistent with that result, the overall distribution of OOS performance dominates the
Figure 4: Stochastic dominance (example 1).

OOS performance of the optimal strategy selection procedure, a clear sign of overfitting.

Figure 5 provides a counter-example, based on the same real investment strategy used in Figure 3. It indicates that the strategy selection procedure used in this backtest actually added value, since the distribution of OOS performance for the selected strategies clearly dominates the overall distribution of OOS performance. (First-order stochastic dominance is a sufficient condition for second-order stochastic dominance, and the plot of $SD2[x]$ is consistent with that fact.)

4 FEATURES OF THE CSCV SAMPLING METHOD

Our testing method utilises multiple developments in the fields of machine learning (combinatorial optimization, jackknife, cross-validation) and decision theory (logistic function, stochastic dominance). Standard cross-validation methods include $k$-fold cross-validation (K-FCV) and leave-one-out cross-validation (LOOCV).

Now, K-FCV randomly divides the sample of size $T$ into $k$ subsamples
of size $T/k$. Then it sequentially tests on each of the $k$ samples the model trained on the $T - T/k$ sample. Although a very valid approach in many situations, we believe that our procedure is more satisfactory than K-FCV in the context of strategy selection. In particular, we would like to compute the Sharpe ratio (or any other performance measure) on each of the $k$ testing sets of size $T/k$. This means that $k$ must be sufficiently small, so that the Sharpe ratio estimate is reliable (see Bailey and Lopez de Prado [2] for a discussion of Sharpe ratio confidence bands). But if $k$ is small, K-FCV will essentially reduce to a "hold-out" method, which we have argued is unreliable. Also, LOOCV is a K-FCV where $k = T$. We are not aware of any reliable performance metric computed on a single OOS observation.

The combinatorially symmetric cross-validation (CSCV) method we have proposed in Section 2.2 differs from both K-FCV and LOOCV. The key idea is to generate $\binom{S}{S/2}$ testing sets of size $T/2$ by recombining the $S$ slices of the overall sample of size $T$. This procedure presents a number of advantages. First, CSCV ensures that the training and testing sets are of equal size, thus providing comparable accuracy to the IS and OOS Sharpe ratios (or any other performance metric that is susceptible to sample size).

Figure 5: Stochastic dominance (example 2).
This is important, because making the testing set smaller than the training set (as hold-out does) would mean that we are evaluating with less accuracy OOS than the was used to choose the optimal strategy. Second, CSCV is symmetric, in the sense that all training sets are re-used as testing sets and vice versa. In this way, the decline in performance can only result from overfitting, not arbitrary discrepancies between the training and testing sets.

Third, CSCV respects the time-dependence and other season-dependent features present in the data, because it does not require a random allocation of the observations to the $S$ subsamples. We avoid that requirement by recombining the $S$ subsamples into the $(S/2)$ testing sets. Fourth, CSCV derives a non-random distribution of logits, in the sense that each logit is deterministically derived from one item in the set of combinations $\theta^4$. As with jackknife resampling, running CSCV twice on the same inputs generates identical results. Therefore, for each analysis, CSCV will provide a single result, $\phi$, which can be independently replicated and verified by another user. Fifth, the dispersion of the distribution of logits conveys relevant information regarding the robustness of the strategy selection procedure. A robust strategy selection leads to a consistent OOS performance rankings, which translate into similar logits.

Sixth, our procedure to estimate PBO is model-free, in the sense that it does not require the researcher to specify a forecasting model or the definitions of forecasting errors. It is also non-parametric, as we are not making distributional assumptions on PBO. This is accomplished by using the concept of logit, $\lambda$. A logit is the logarithm of odds. In our problem, the odds are represented by relative ranks (i.e., the odds that the optimal strategy chosen IS happens to underperform OOS). The logit function presents the advantage of being the inverse of the sigmoidal logistic distribution, which resembles the cumulative Normal distribution.

As a consequence, if $\mathcal{Z}_{i}$ are distributed close to uniformly (the case when the backtest appears to be informationless), the distribution of the logits will approximate the standard Normal. This is important, because it gives us a baseline of what to expect in the threshold case where the backtest does not seem to provide any insight into the OOS performance. If good backtesting results are conducive to good OOS performance, the distribution of logits will be centered in a significantly positive value, and its left tail will marginally cover the region of negative logit values, making $\phi \approx 0$.

A key parameter of our procedure is the value of $S$. This regulates the number of submatrices $M$, that will be generated, each of order $(T/S \times N)$, and also the number of logit values that will be computed, $(S/2)^2$. Indeed, $S$ must be large enough so that the number of combinations suffices to draw
inference. If $S$ is too small, the left tail of the distribution of logits will be underrepresented. On the other hand, if we believe that the performance series is time-dependent and incorporates seasonal effects, $S$ cannot be too large, or the relevant time structure may be shuttered across the partitions.

For example, if the backtest includes more than six years of data, $S = 24$ generates partitions spanning over a quarter each, which would preserve daily, weekly and monthly effects, while producing a distribution of 2,704,156 logits. By contrast, if we are interested in quarterly effects, we have two choices: i) Work with $S = 12$ partitions, which will give us 924 logits, and/or ii) double $T$, so that $S$ does not need to be reduced. The accuracy of the procedure relies on computing a large number of logits, where that number is derived in Equation (2.3). Because $f(\lambda)$ is estimated as a proportion of the number of logits, $S$ needs to be large enough to generate sufficient logits. For a proportion $p$ estimated on a sample of size $N$, the standard deviation of its expected value can be computed as $\sigma[p] = \sqrt{p(1-p)/N}$ (see Gelman and Hill [10]). In other words, the standard deviation is highest for $p = \frac{1}{2}$, with $\sigma[p] = \sqrt{\frac{1}{4N}}$. Fortunately, even a small number $S$ generates a large number of logits. For example, $S = 16$ we will obtain 12,780 logits (see Equation (2.3)), and $\sigma[f(\lambda)] < 0.0045$, with less than a 0.01 estimation error at 95% confidence level. Also, if $M$ contains 4 years of daily data, $S = 16$ would equate to quarterly partitions, and the serial correlation structure would be preserved. For these two reasons, we believe that $S = 16$ is a reasonable value to use in most cases.

Another key parameter is the number of trials (i.e., the number of columns in $M_T$). Hold-out's disregard for the number of trials attempted was the reason we concluded it was an inappropriate method to assess a backtest's representativeness (see Bailey et al. [1] for a proof). $N$ must be large enough to provide sufficient granularity to the values of the relative rank, $\hat{\omega}_c$. If $N$ is too small, $\hat{\omega}_c$ will take only a very few values, which will translate into a very discrete number of logits, making $f(\lambda)$ too discontinuous, and adding estimation error to the evaluation of $\phi$. For example, if the investor is sensitive to values of $\phi < 1/10$, it is clear that the range of values that the logits can adopt must be greater than 10, and so $N \gg 10$ is required. Other considerations regarding $N$ will be discussed in the following Section.

Finally, PBO is evaluated by comparing combinations of $\Gamma/2$ observations with their complements. But the backtest works with $T$ observations, rather than only $\Gamma/2$. Therefore, $T$ should be chosen to be double of the number of observations used by the investor to choose a model configuration
or to determine a forecasting specification.

5 LIMITATIONS AND MISUSE

The general framework in Subsection 2.1 can be flexibly used to assess backtest overfitting probability. Quantitative assessment, however, also relies on methods for estimating the probability measure. In this paper, we focus on one of such implementations: the CSCV method. This procedure was designed to evaluate PBO under minimal assumptions and input requirements. In doing so, we have attempted to provide a very general (in fact, model-free and non-parametric) procedure against which IS backtests can be benchmarked. However, any particular implementation has its limitations and the CSCV method is no exception. Below is a discussion of some of the limitations of this method from the perspective of design and application.

5.1 Limitation in design

First, a key feature of the CSCV implementation is symmetry. In dividing the total sample of the testing results into IS and OOS both the size and method of division in CSCV are symmetric. The advantage of such an symmetric division has been elaborated above. However, the complexity of investment strategies and performance measures makes it unlikely that any particular method will be a one size fits all solution. For some backtests other methods, for example K-FCV, may well be better suited.

Moreover, symmetrically dividing the sample performance into \( S \) symmetrically layered sub-samples also may not suitable for certain strategies. For example, if the performance measure as a time series has a strong autocorrelation, then such a division may obscure the characterization especially when \( S \) is large.

Finally, the CSCV estimate of the probability measure assumes all the sample statistics carries the same weight. Without knowing any prior information on the distribution of the backtest performance measure this is, of course, a natural and reasonable choice. If, however, one does have knowledge regarding the distribution of the backtest performance measure, then model-specific methods of dividing the sample performance measure and assigning different weights to different strips of the subdivision are likely to be more accurate. For instance, if a forecasting equation was used to generate the trials, it would be possible to develop a framework that evaluates PBO particular to that forecasting equation.
5.2 Limitation in application

First, the researcher must provide full information regarding the actual trials conducted, to avoid the file drawer problem (the test is only as good as the completeness of the underlying information), and should test as many strategy configurations as is reasonable and feasible. Hiding trials will lead to an underestimation of the overfit, because each logit will be evaluated under a biased relative rank \( \tilde{m_c} \). This would be equivalent to removing subjects from the trials of a new drug, once we have verified that the drug was not effective on them. Likewise, adding trials that are doomed to fail in order to make one particular model configuration succeed biases the result. If a model configuration is obviously flawed, it should have never been tried in the first place. A case in point is guided searches, where an optimization algorithm uses information from prior iterations to decide what direction should be followed next. In this case, the columns of matrix \( M \) should be the final outcome of each guided search (i.e., after it has converged to a solution), and not the intermediate steps.\(^2\) This procedure aims at evaluating how reliable a backtest selection process is when choosing among feasible strategy configurations. As a rule of thumb, the researcher should backtest as many theoretically reasonable strategy configurations as possible.

Second, this procedure does nothing to evaluate the correctness of a backtest. If the backtest is flawed due to bad assumptions, such as incorrect transaction costs or using data not available at the moment of making a decision, our approach will be making an assessment based on flawed information.

Third, this procedure only takes into account structural breaks as long as they are present in the dataset of length \( T \). If a structural break occurs outside the boundaries of the available dataset, the strategy may be overfit to a particular data regime, which our PBO has failed to account for because the entire set belongs to the same regime. This invites the more general warning that the dataset used for any backtest is expected to be representative of future states of the modeled financial variable.

Fourth, although a high PBO indicates overfitting in the group of \( N \) tested strategies, skillful strategies can still exist in these \( N \) strategies. For example, it is entirely possible that all the \( N \) strategies have high but similar Sharpe ratios. Since none of the strategies is clearly better than the rest, PBO will be high. Here overfitting is among many 'skillful' strategies.

Fifth, we must warn the reader against applying CSCV to guide the

\(^2\)We thank David Aronson and Timothy Masters (Baruch College) for asking for this clarification.
search for an optimal strategy. That would constitute a gross misuse of our method. As Strathern [31] eloquently put it, "when a measure becomes a target, it ceases to be a good measure." Any counter-overfitting technique used to select an optimal strategy will result in overfitting. For example, CSCV can be employed to evaluate the quality of a strategy selection process, but PBO should not be the objective function on which such selection relies.

6 A PRACTICAL APPLICATION

Bailey et al. [1] present an example of an investment strategy that attempts to profit from a seasonal effect. For the reader's convenience, we reiterate here how the strategy works. Suppose that we would like to identify the optimal monthly trading rule, given four customary parameters: Entry_day, Holding_period, Stopjoss and Side.

Side defines whether we will hold long or short positions on a monthly basis. Entry_day determines the business day of the month when we enter a position. Holding_period gives the number of days that the position is held. Stopjoss determines the size of the loss as a multiple of the series' volatility that triggers an exit for that month's position. For example, we could explore all nodes that span the interval [1, . . . , 22] for Entry_day, the interval [1, . . . , 20] for Holding_period, the interval [0, . . . , 10] for Stopjoss, and {−1, 1} for Sign. The parameters combinations involved form a four-dimensional mesh of 8,800 elements. The optimal parameter combination can be discovered by computing the performance derived by each node.

First, as discussed in the above cited paper, a time series of 1,000 daily prices (about 4 years) was generated by drawing from a random walk. Parameters were optimized (Entry_day = 11, Holding_period = 4, Stopjoss = -1 and Side = 1), resulting in an annualized Sharpe ratio of 1.27. Given the elevated Sharpe ratio, we may conclude that this strategy's performance is significantly greater than zero for any confidence level. Indeed, the PSR-Stat is 2.83, which implies a less than 1% probability that the true Sharpe ratio is below 0 (see Bailey and Lopez de Prado [2] for details). Figure 6 gives a graphical illustration of this example.

We have estimated the PBO using our CSCV procedure, and obtained the results illustrated below. Figure 7 shows that approx. 53% of the SR OOS are negative, despite all SR IS being positive and ranging between 1 and 2.2. Figure 8 plots the distribution of logits, which implies that, despite the elevated SR IS, the PBO is as high as 55%. Consequently,
Figure 6 shows backtested performance of a seasonal strategy (example 1).

Figure 9 shows that the distribution of optimized OOS SR does not dominate the overall distribution of OOS SR. This is consistent with the fact that the underlying series follows a random walk, thus the serial independence among observations makes any seasonal patterns coincidental. The CSCV framework has succeeded in diagnosing that the backtest was overfit.

Second, we generated a time series of 1,000 daily prices (about 4 years), following a random walk. But unlike the first case, we have shifted the returns of the first 5 random observations of each month to be centered at a quarter of a standard deviation. This simulates a monthly seasonal effect, which the strategy selection procedure should discover. Figure 10 plots the random series, as well as the performance associated with the optimal parameter combination: Entry_day = 1, Holding_period = 4, Stopjoss = -10 and Side = 1. The annualized Sharpe ratio at 1.54 is similar to the previous (overfit) case (1.54 vs. 1.3).

The next three graphs report the results of the CSCV analysis, which confirm the validity of this backtest in the sense that performance inflation from overfitting is minimal. Figure 11 shows only 13% of the OOS SR to be negative. Because there is a real monthly effect in the data, the PBO for
Figure 7: CSCV analysis of the backtest of a seasonal strategy (example 1): Performance degradation.

this second case should be substantially lower than the PBO of the first case. Figure 12 shows a distribution of logits with a PBO of only 13%. Figure 13 evidences that the distribution of OOS SR from IS optimal combinations clearly dominates the overall distribution of OOS SR. The CSCV analysis has this time correctly recognized the validity of this backtest, in the sense that performance inflation from overfitting is small.

In this practical application we have illustrated how simple is to produce overfit backtests when answering common investment questions, such as the presence of seasonal effects. We refer the reader to [1, Appendix 4] for the implementation of this experiment in Python language. Similar experiments can be designed to demonstrate overfitting in the context of other effects, such as trend-following, momentum, mean-reversion, event-driven effects, and the like. Given the facility with which elevated Sharpe ratios can be manufactured IS, the reader would be well advised to remain critical of backtests and researchers that fail to report the PBO results.
7 CONCLUSIONS

In [2] Bailey and Lopez de Prado developed methodologies to evaluate the probability that a Sharpe ratio is inflated (PSR), and to determine the minimum track record length (MinTRL) required for a Sharpe ratio to be statistically significant. These statistics were developed to assess Sharpe ratios based on live investment performance and backtest track records. This paper has extended this approach to present formulas and approximation techniques for finding the probability of backtest overfitting.

To that end, we have proposed a general framework for modeling the IS and OOS performance using probability. We define the probability of backtested overfitting (PBO) as the probability that an optimal strategy IS underperforms the mean OOS. To facilitate the evaluation of PBO for particular applications, we have proposed a combinatorially symmetric cross-validation (CSCV) implementation framework for estimating this probability. This estimate is generic, symmetric, model-free and non-parametric. We have assessed the accuracy of CSCV as an approximation of PBO in
two different ways, on a wide variety of test cases. Monte Carlo simulations show that CSCV applied on a single dataset provides similar results to computing PBO on a large number of independent samples. We have also directly computed PBO by deriving the Extreme Value distributions that model the performance of IS optimal strategies. These results indicate that CSCV provides reasonable estimates of PBO, with relatively small errors.

Besides estimating PBO, our general framework and its CSCV implementation scheme can also be used to deal with other issues related to overfitting, such as performance degeneration, probability of loss and possible stochastic dominance of a strategy. On the other hand, the CSCV implementation also has some limitations. This suggests that other implementation frameworks may well be more suitable, particularly for problems with structure information.

Nevertheless, we believe that CSCV provides both a new and powerful tool in the arsenal of an investment and financial researcher, and that it also
Figure 10: Backtested performance of a seasonal strategy (example 2).

constitutes a nice illustration of our general framework for quantitatively studying issues related to backtest overfitting. We certainly hope that this study will raise greater awareness concerning the futility of computing and reporting backtest results, without first controlling for PBO and MinBTL.

References


Figure 11: CSCV analysis of the backtest of a seasonal strategy (example 2): monthly effect.


Figure 12: CSCV analysis of the backtest of a seasonal strategy (example 2): logit distribution.


Figure 13: CSCV analysis of the backtest of a seasonal strategy (example 2): dominance.


THE SHARPE RATIO EFFICIENT FRONTIER

David H. Bailey
Complex Systems Group Leader - Lawrence Berkeley National Laboratory
dhbailey@lbl.gov

Marcos M. Lopez de Prado
Head of Global Quantitative Research - Tudor Investment Corp.
and
Research Affiliate - Lawrence Berkeley National Laboratory
lopezdeprado@lbl.gov

First version: May 2008
This version: April 2012

We thank the editor of the Journal of Risk and two anonymous referees for helpful comments. We are grateful to Tudor Investment Corporation, Jose Blanco (UBS), Sid Browne (Guggenheim Partners), David Easley (Cornell University), Laurent Favre (Alternative Soft), Matthew Foreman (University of California, Irvine), Ross Garon (S.A.C. Capital Advisors), Robert Jarrow (Cornell University), David Leinweber (Lawrence Berkeley National Laboratory), Elmar Mertens (Federal Reserve Board), Attilio Meucci (Kepos Capital, SUNY), Maureen O’Hara (Cornell University), Eva del Pozo (Complutense University), Riccardo Rebonato (PIMCO, University of Oxford) and Luis Viceira (HBS).

Supported in part by the Director, Office of Computational and Technology Research, Division of Mathematical, Information, and Computational Sciences of the U.S. Department of Energy, under contract number DE-AC02-05CH1231.
ABSTRACT

We evaluate the probability that an estimated Sharpe ratio exceeds a given threshold in presence of non-Normal returns. We show that this new uncertainty-adjusted investment skill metric (called Probabilistic Sharpe ratio, or PSR) has a number of important applications: First, it allows us to establish the track record length needed for rejecting the hypothesis that a measured Sharpe ratio is below a certain threshold with a given confidence level. Second, it models the trade-off between track record length and undesirable statistical features (e.g., negative skewness with positive excess kurtosis). Third, it explains why track records with those undesirable traits would benefit from reporting performance with the highest sampling frequency such that the IID assumption is not violated. Fourth, it permits the computation of what we call the Sharpe ratio Efficient Frontier (SEF), which lets us optimize a portfolio under non-Normal, leveraged returns while incorporating the uncertainty derived from track record length.

Keywords: Sharpe ratio, Efficient Frontier, IID, Normal distribution, Skewness, Excess Kurtosis, track record.

JEL Classifications: C02, G11, G14, D53.
1. INTRODUCTION

Roy (1952) was the first to suggest a risk-reward ratio to evaluate a strategy's performance. Sharpe (1966) applied Roy's ideas to Markowitz's mean-variance framework, in what has become one of the best known performance evaluation metrics. Lopez de Prado and Peijan (2004) showed that the implied assumptions (namely, that returns are independent and identically distributed (IID) Normal) may hide substantial drawdown risks, especially in the case of hedge fund strategies.

Renowned academics (Sharpe among them) have attempted to persuade the investment community against using the Sharpe ratio in breach of its underlying assumptions. Notwithstanding its many deficiencies, Sharpe ratio has become the 'gold standard' of performance evaluation. Sharpe ratios are greatly affected by some of the statistical traits inherent to hedge fund strategies in general (and high frequency strategies in particular), like non-normality and reduced granularity (due to returns aggregation). As a result, Sharpe ratios from these strategies tend to be "inflated". Ingersoll, Spiegel, Goetzmann and Welch (2007) explain that sampling returns more frequently reduces the inflationary effect that some manipulation tactics have on the Sharpe ratio.

We accept the futility of restating Sharpe ratio's deficiencies to investors. Instead, a first goal of this paper is to introduce a new measure called Probabilistic Sharpe Ratio (PSR), which corrects those inflationary effects. This uncertainty-adjusted Sharpe ratio demands a longer track record length and/or sampling frequency when the statistical characteristics of the returns distribution would otherwise inflate the Sharpe ratio. That leads us to our second goal, which is to show that Sharpe ratio can still evidence skill if we learn to require the proper length for a track record. We formally define the concept of Minimum Track Record Length (MinTRL) needed for rejecting the null hypothesis of 'skill beyond a given threshold' with a given degree of confidence. The question of how long should a track record be in order to evidence skill is particularly relevant in the context of alternative investments, due to their characteristic non-Normal returns. Nevertheless, we will discuss the topic of "track record length" from a general perspective, making our results applicable to any kind of strategy or investment.

A third goal of this paper is to introduce the concept of Sharpe ratio Efficient Frontier (SEF), which permits the selection of optimal portfolios under non-Normal, leveraged returns, while taking into account the sample uncertainty associated with track record length. The portfolio optimization approach hereby presented differs from other higher-moment methods in that skewness and kurtosis are incorporated through the standard deviation of the Sharpe ratio estimator. This avoids having to make arbitrary assumptions regarding the relative weightings that higher moments have in the utility function. We feel that practitioners will find this approach useful, because the Sharpe ratio has become -to a certain extent- the default utility function used by investors. SEF can be intuitively explained to investors as the set of portfolios that maximize the expected Sharpe ratio for different degrees of confidence. The maximum Sharpe ratio portfolio is a member of the SEF, but it may differ from the portfolio that maximizes the PSR. While the former portfolio is oblivious to the resulting confidence bands around that maximized

---

1 See Sharpe (1975) and Sharpe (1994). Sharpe suggested the name reward-to-variability ratio, another matter on which that author's plea has been dismissed.
Sharpe, the latter is the portfolio that maximizes the probability of skill, taking into account the impact that non-Normality and track record length have on the Sharpe ratio’s confidence band.

We do not explicitly address the case of serially-conditioned processes. Instead, we rely on Mertens (2002), who originally assumed IID non-Normal returns. That framework is consistent with the scenario that the skill and style of the portfolio manager does not change during the observation period. Fortunately, Opdyke (2007) has shown that Mertens’ equation has a limiting distribution that is valid under the more general assumption of stationary and ergodic returns, and not only IID. Thus, our results are valid under such conditions, beyond the narrower IID assumption.

The rest of the paper is organized as follows: Section 2 presents the theoretical framework that will allow us to achieve the three stated goals. Section 3 introduces the concept of Probabilistic Sharpe Ratio (PSR). Section 4 relates applies this concept to answer the question of what is an acceptable track record length for a given confidence level. Section 5 presents numerical examples that illuminate how these concepts are interrelated and can be used in practice. Section 6 applies our methodology to Hedge Fund Research data. Section 7 takes this argument further by introducing the concept of Sharpe Ratio Efficient Frontier (SEF). Section 8 outlines the conclusions. Two mathematical appendices proof statements made in the body of the paper.

2. THE FRAMEWORK
We have argued that the Sharpe ratio is a deficient measure of investment skill. In order to understand why, we need to review its theoretical foundations, and the implications of its assumption of Normal returns. In particular, we will see that non-Normality may increase the variance of the Sharpe ratio estimator, therefore reducing our confidence in its point estimate. When unaddressed, this means that investors may be comparing Sharpe ratio estimates with widely different confidence bands.

2.1. SHARPE RATIO’S POINT ESTIMATE
Suppose that a strategy’s excess returns (or risk premiums), \( r_t \), are IID\(^2\)

\[
    r \sim N(\mu, \sigma^2)
\]  

(1)

where \( N \) represents a Normal distribution with mean \( \mu \) and variance \( \sigma^2 \). The purpose of the Sharpe ratio (SR) is to evaluate the skills of a particular strategy or investor.

\[
    SR = \frac{\mu}{\sigma}
\]  

(2)

Since \( \mu, \sigma \) are usually unknown, the true value \( SR \) cannot be known for certain. The inevitable consequence is that Sharpe ratio calculations may be the subject of substantial estimation errors. We will discuss next how to determine them under different sets of assumptions.

\(^2\) Even if returns are serially correlated, there may be a sampling frequency for which their autocorrelation becomes insignificant. We leave for a future paper the analysis of returns’ serial conditionality under different sampling frequencies, and their joint impact on Sharpe ratio estimates.
2.2. ASSUMING IID NORMAL RETURNS

Like any estimator, SR has a probability distribution. Following Lo (2002), in this section we will derive what this distribution is in the case of IID Normal returns. The Central Limit Theorem states that $\sqrt{n}(\hat{\mu} - \mu) \overset{a}{\rightarrow} N(0, \sigma^2)$ and $\sqrt{n}(\hat{\sigma}^2 - \sigma^2) \overset{a}{\rightarrow} N(0, 2\sigma^4)$, where $\overset{a}{\rightarrow}$ denotes asymptotic convergence. Let $\theta = (\mu, \sigma)$ be the column-vector of the Normal distribution’s parameters, with an estimate in $\hat{\theta} = (\hat{\mu}, \hat{\sigma})$. For IID returns, $\sqrt{n}(\hat{\theta} - \theta) \overset{a}{\rightarrow} N(0, V_{\theta})$, where $V_{\theta} = \begin{pmatrix} \sigma^2 & 0 \\ 0 & 2\sigma^4 \end{pmatrix}$ is the variance of the estimation error on $\theta$.

Let’s denote $\hat{SR} = g(\hat{\theta})$, where $g(.)$ is the function that estimates $SR$, and apply the delta method (see White (1984)),

$$\sqrt{n}\{g(\hat{\theta}) - g(\theta)\} \overset{a}{\rightarrow} N(0, V_g)$$  \hspace{1cm} (3)

$v_g$ is the variance of the $g(.)$ function. Because $\frac{\partial g}{\partial \theta} = \begin{pmatrix} \frac{1}{\sigma} \\ -\frac{\hat{\mu}}{2\sigma^2} \end{pmatrix}$, we obtain that $V_g = \left(\frac{\partial g}{\partial \mu}\right)^2 \sigma^2 + \left(\frac{\partial g}{\partial \sigma^2}\right)^2 2\sigma^4$. This means that the asymptotic distribution of $\hat{SR}$ reduces to

$$\left(\hat{SR} - SR\right) \overset{a}{\rightarrow} N\left(0, \frac{1 + \frac{1}{2}SR^2}{n}\right)$$  \hspace{1cm} (4)

If $q$ is the number of observations per year, the point estimate of the annualized Sharpe ratio is

$$SR_q \overset{a}{\rightarrow} N^{\lambda}SR, \quad qV_g$$  \hspace{1cm} (5)

Under the assumption of Normal IID returns, the $SR$ estimator follows a Normal distribution with mean $SR$ and a standard deviation that depends on the very value of $SR$ and the number of observations. This is an interesting result, because it tells us that, ceteris paribus, in general we would prefer investments with a longer track record. That is hardly surprising, and is common practice in the hedge fund industry to ask for track records greater than 3 or more years of monthly returns. Furthermore, Eq. (4) tells us how a greater $n$ exactly impacts the variance of the $SR$ estimate, which is an idea we will expand in later sections.

2.3. SHARPE RATIO AND NON-NORMALITY

The $SR$ does not characterize a distribution of returns, in the sense that there are infinite Normal distributions that deliver any given $SR$. This is easy to see in Eq. (2), as merely re-scaling the returns series will yield the same $SR$, even though the returns come from Normal distributions.
with different parameters. This argument can be generalized to the case of non-Normal distributions, with the aggravation that, in the non-Normal case, the number of degrees of freedom is even greater (distributions with entirely different first four moments may still yield the same $SR$).

Appendix 2 demonstrates that a simple mixture of two Normal distributions produces infinite combinations of skewness and kurtosis with equal $SR$. More precisely, the proof states that, in the most general cases, there exists a $p$ value able to mix any two given Normal distributions and deliver a targeted $SR$. The conclusion is that, however high a $SR$ might be, it does not preclude the risk of severe losses. To understand this fact, consider the following combinations of parameters:

$$
\begin{align*}
\mu_1 &= \mu_1^* \left(1 - \frac{j-1}{k-1}\right) + SR^* \frac{j-1}{k} ; \mu_2 = SR^* \left(1 + \frac{1}{k}\right) \left(1 - \frac{j-1}{k-1}\right) + \mu_2^* \frac{j-1}{k-1} \\
\sigma_1 &= \frac{\sigma_1^*}{k} j; \sigma_2 = \frac{\sigma_2^*}{k} j \\
& j = 1, ..., k
\end{align*}
$$

For $\mu_1^* \leq \mu_j \leq SR^* \left(1 - \frac{j-1}{k}\right)$ and $SR^* \left(1 + \frac{1}{k}\right) \leq \mu_2 \leq \mu_2^*$, each combination implies a non-Normal mixture. For $k=20$ and $(\mu_1^*, \mu_2^*, \sigma_1^*, \sigma_2^*) = (-5,5,5,5)$, there are 160,000 combinations of $(\mu_1, \mu_2, \sigma_1, \sigma_2)$, but as determined in Appendix 2, only for 96,551 of them there exists a $p$ such that $SR^* = 1$. Figure 1 plots the resulting combinations of skewness and kurtosis for mixtures of Normal distributions with the same Sharpe ratio ($SR^* = 1$). An interesting feature of modeling non-Normality through a mixture of Normal distributions is the trade-off that exists between skewness and kurtosis. In this analytical framework, the greater the absolute value of skewness is, the greater the kurtosis tends to be. Lopez de Prado and Peijan (2004) find empirical evidence of this trade-off in their study of returns distributions of hedge fund styles. A mixture of Normal distributions seems to accurately capture this feature in the data.

[EXHIBIT 1 HERE]

The above set includes combinations as different as $(\mu_1, \mu_2, \sigma_1, \sigma_2, \rho) = (-5,1.05,5,0.05,0.015)$ and $(\mu_1, \mu_2, \sigma_1, \sigma_2, \rho) = (0.3237,1.8816,0.05,0.05,0.8706)$. Figure 2 displays the probability density functions of these two distributions, which have the same Sharpe ratio ($SR^* = 1$). The continuous line represents the mixture of two Normal distributions, and the dashed line the Normal distribution with the same mean and standard deviation as the mixture. The mixture on the right side incorporates a 1.5% probability that a return is drawn from a distribution with mean -5 and a standard deviation of 5 (a catastrophic outcome).

[FIGURE 2 HERE]

Consequently, for a risk averse investor, $SR$ does not provide a complete ranking of preferences, unless non-Normality is taken into account. But, how accurately can skewness and kurtosis be

---

3 Readers interested in the estimation of the parameters that characterize a mixture of 2 Gaussians will find an efficient algorithm in Lopez de Prado and Foreman (2011).
estimated from this set of mixtures? In order to answer that question, for each of the 96,551 mixtures included in the above set we have generated a random sample of 1,000 observations (roughly 4 years of daily observations), estimated the first 4 moments on each random sample and compared those estimates with the true mixture’s moments (see Eqs. (26)-(35)). Figures 3 (a-d) show that the estimation error is relatively small when moments adopt values within reasonable ranges, particularly for the first 3 moments.

[FIGURE 3 HERE]

Figure 4 reports the results of fitting the two specifications in Eq. (7) on the estimation errors $i e r$ and their squares ($e r^2$) for moments $m=\ldots, 4$.

$$
e r_1 = \delta_{0,m} + \delta_{1,m} y_m + S_{2,m} y_m^2 + \varepsilon_1$$  \hspace{1cm} (7) 
$$
e r_1^2 = \theta_{0,m} + \theta_{1,m} y_m + S_{2,m} y_m^2 + \xi_1$$

where $y_1 = \mu$, $\gamma_2 = \sigma$, $\gamma_3 = \frac{e^{(x-\mu)^3}}{\sigma^3}$ is skewness, and $\gamma_4 = \frac{e^{(x-\mu)^4}}{\sigma^4}$ is kurtosis.

[FIGURE 4 HERE]

Consistent with the visual evidence in Figure 3, Figure 4 shows that the estimation error of the mean is not a function of the mean’s value (see $e r$ _Prob column with prob values at levels usually rejected). The standard deviation’s estimator is biased towards underestimating risks (the intercept’s $e r$ _Prob is at levels at which we would typically reject the null hypothesis of unbiasedness), but at least the estimation error does not seem affected by the scale of the true standard deviation. In the case of the third and fourth moments’ estimation errors, we find bias and scale effects of first and second degree. This is evidence that estimating moments beyond the third, and particularly the fourth moment, requires longer sample lengths than estimating only the first two moments. We will restate this point in Section 4.

2.4. INCORPORATING NON-NORMALITY

The previous section argued that non-Normal distributions with very diverse risk profiles can all have the same $SR$. In this section we will discuss the key fact that, although skewness and kurtosis does not affect the point estimate of $SR$, it greatly impacts its confidence bands, and consequently its statistical significance. This fact of course has dreadful implications when, as it is customary, point estimates of $SR$ are used to rank investments.

Mertens (2002) concludes that the Normality assumption on returns could be dropped, and still the estimated Sharpe ratio would follow a Normal distribution with parameters

$$(\overline{SR} - SR)^a \sim N \left(0, \frac{1 + \frac{1}{2} SR^2 - \gamma_3 SR + \gamma_4 - 3 SR^2}{n} \right)$$  \hspace{1cm} (8)$$

The good news is, $SR$ follows a Normal distribution even if the returns do not. The bad news is, although most investors prefer to work in a mean-variance framework, they need to take non-
Normality into account (in addition, of course, to sample length). Figure 5 illustrates how combinations of skewness and kurtosis impact the standard deviation of the SR estimator. This has the serious implication that non-Normal distributions may severely inflate the SR estimate, to the point that having a high SR may not be sufficient warranty of its statistical significance.

[FiguRE 5 HERE]

Christie (2005) uses a GMM approach to derive a limiting distribution that only assumes stationary and ergodic returns, thus allowing for time-varying conditional volatilities, serial correlation and even non-IID returns. Surprisingly, Opdyke (2007) proved that the expressions in Mertens (2002) and Christie (2005) are in fact identical. To Dr. Mertens’ credit, his result appears to be valid under the more general assumption of stationary and ergodic returns, and not only IID.

2.5. CONFIDENCE BAND
We have mentioned that skewness and kurtosis will affect the confidence band around our estimate of SR, but we did not explicitly derive its expression. After some algebra, Eq.(8) gives the estimated standard deviation of $\tilde{S}R$ as $\tilde{\sigma}_{SR} = \sqrt{1 - \gamma_3 SR + \frac{\gamma_4 - 1}{4} SR^2 / n - 1}$, where $n - 1$ is due to Bessel’s correction. The true value $SR$ is bounded by our $\tilde{S}R$ estimate with a significance level $\alpha$

$$Prob[SR \in (SR - Z_{\alpha/2} d_{SR}, SR + Z_{\alpha/2} \tilde{\sigma}_{SR})] = 1 - \alpha$$

(9)

In general it is misleading to judge strategies’ performance by merely comparing their respective point estimates of $\tilde{S}R$, without considering the estimation errors involved in each calculation. Instead, we could compare $\tilde{S}R$’s translation in probabilistic terms, which we will define next.

3. PROBABILISTIC SHARPE RATIO (PSR)
Now that we have derived an expression for the confidence bands of SR, we are ready to aim for the first goal stated in the Introduction: Provide a de-inflated estimate of SR. Given a predefined benchmark $^4$ Sharpe ratio ($SR^\ast$), the observed Sharpe ratio $\tilde{S}R$ can be expressed in probabilistic terms as

$$PSRiSR^\ast = Prob[SR > SR^\ast] = 1 - \int_{-\infty}^{SR^\ast} Prob(SR) \cdot dSR$$

(10)

We ask the question, what is the probability that $\tilde{S}R$ is greater than a hypothetical $SR^\ast$? Applying what we have learnt in the previous sections, we propose

---

$^4$ This could be set to a default value of zero (i.e., comparing against no investment skill).
\[ P_{SR}(SR^*) = Z \left[ \frac{(SR - SR^*)\sqrt{n - 1}}{\sqrt{1 - \hat{\gamma}_3^2 SR + \hat{\gamma}_4 - 1/4}} \right] \]  

(11)

where \( Z \) is the \( cdf \) of the Standard Normal distribution. For a given \( SR^* \), \( P_{SR} \) increases with greater \( SR \) (in the original sampling frequency, i.e. non-annualized), or longer track records (\( \leq \)), or positively skewed returns (\( f_3 \)), but it decreases with fatter tails (\( f_4 \)). Because hedge fund strategies are usually characterized by negative skewness and fat tails (Brooks and Kat (2002), Lopez de Prado and Rodrigo (2004)), Sharpe ratios tend to be "inflated". \( P_{SR}(SR^*) \) takes those characteristics into account and delivers a corrected, atemporal measure of performance expressed in terms of probability of skill. It is not unusual to find strategies with irregular trading frequencies, such as weekly strategies that may not trade for a month. This poses a problem when computing an annualized Sharpe ratio, and there is no consensus as how skill should be measured in the context of irregular bets. Because \( PSR \) measures skill in probabilistic terms, it is invariant to calendar conventions. All calculations are done in the original frequency of the data, and there is no annualization. This is another argument for preferring \( PSR \) to traditional annualized \( SR \) readings in the context of strategies with irregular frequencies.

Section 2.3 made the point that estimates of skewness and kurtosis may incorporate significant errors. If the researcher believes that this is the case with their estimated \( \hat{\gamma}_3 \) and \( \hat{\gamma}_4 \), we recommend that a lower bound is inputted in place of \( f_3 \) and an upper bound in place of \( f_4 \) in Eq. (8), for a certain confidence level. However, if these estimates are deemed to be reasonably accurate, this 'worst case scenario analysis' is not needed.

An example will clarify how \( PSR \) reveals information otherwise dismissed by \( SR \). Suppose that a hedge fund offers you the statistics displayed in Figure 6, based on a monthly track record over the last two years.

[FIGURE 6 HERE]

[FIGURE 7 HERE]

At first sight, an annualized Sharpe ratio of 1.59 over the last two years seems high enough to reject the hypothesis that it has been achieved by sheer luck. The question is, "how inflated is this annualized Sharpe ratio due to the track record's non-normality, length and sampling frequency?" Let's start by comparing this performance with the skill-less benchmark (\( SR^* = 0 \)) while assuming Normality (\( f_3 = 0, f_4 = 3 \)). The original sampling frequency is monthly, and so the estimate that goes into Eq. (11) is \( \hat{SR} = 0.458 \). This yields a reassuring \( P_{SR}(0) = 0.982 \). However, when we incorporate the skewness (\( f_3 = -2.448 \)) and kurtosis information (\( f_4 = 10.164 \)), then \( P_{SR}(0) = 0.913 \) at a 95% confidence level, we would accept this track record in

\(^5 \text{SR} \) and \( SR^* \) are expressed in the same frequency as the returns time series.

\(^6 \) After applying \( PSR \) on his track record, a hedge fund manager suggested this measure to be named “The Sharpe razor” [sic].
the first instance, but could not reject the hypothesis that this Sharpe ratio is skill-less in the second instance.

Figure 7 illustrates what is going on. The dashed black line is the Normal pdf that matches the Mean and StDev values in Figure 6. The black line represents the mixture of two Normal distributions that matches all four moments in Table 1 \((\mu_1 = -0.1, \mu_2 = 0.06, \sigma_1 = 0.12, \sigma_2 = 0.03, p=0.5)\). Clearly, it is a mistake to assume normality, as that would ignore critical information regarding the hedge fund's loss potential.

What the annualized Sharpe ratio of 1.59 was hiding was a relatively small probability (15\%) of a return drawn from an adverse distribution (a negative multiple of the mixed distribution's mode). This is generally the case in track records with negative skewness and positive excess kurtosis, and it is consistent with the signs of \(\hat{\gamma}_3\) and \(\hat{\gamma}_4\) in Eq. (11).

This is not to say that a track record of 1.59 Sharpe ratio is worthless. As a matter of fact, should we have 3 years instead of 2, \(PSR(0) = 0.953\), enough to reject the hypothesis of skill-less performance even after considering the first four moments. In other words, a longer track record may be able to compensate for the uncertainty introduced by non-Normal returns. The next Section quantifies that "compensation effect" between non-Normality and the track record's length.

PSR takes into account the statistical accuracy of the point estimate of \(SR\) for different levels of skewness and kurtosis (and length of track record). In this sense, it incorporates information regarding the non-Normality of the returns. However, we caution the reader that PSR does not, and does not attempt to, incorporate the effect of higher moments on preferences. The investor still only cares about mean and variance, but she is rightly worried that in the presence of skewness and kurtosis -about which she does not care per se- her estimates may be inaccurate and 'flattering'.

4. TRACK RECORD LENGTH

Understanding that Sharpe ratio estimations are subject to significant errors begs the question: "How long should a track record be in order to have statistical confidence that its Sharpe ratio is above a given threshold? " In mathematical terms, for \(\hat{SR} > SR^\ast\), this is equivalent to asking

\[
\{n | \hat{PSR}(SR^\ast) > 1 - \alpha\}
\]

(12)

with minimum track record length (MinTRL) in

\[
MinTRL = n^\ast = 1 + \left[1 - \hat{\gamma}_3 SR + \frac{\hat{\gamma}_4 - 1}{4} SR^2\left(\frac{Z_{\alpha}}{SR - SR^\ast}\right)^2\right] (13)
\]

And again we observe that a longer track record will be required the smaller \(SR\) is, or the more negatively skewed returns are, or the greater the fat tails, or the greater our required level of confidence. A first practical implication is that, if a track record is shorter than MinTRL, we do
not have enough confidence that the observed $\tilde{S}R$ is above the designated threshold $SR^\ast$. A second practical implication is that a portfolio manager will be penalized because of her non-Normal returns, however she can regain the investor’s confidence over time (by extending the length of her track record).

It is important to note that $MinTRL$ is expressed in terms of number of observations, not annual or calendar terms. A note of caution is appropriate at this point: Eqs. (11) and (13) are built upon Eq. (8), which applies to an asymptotic distribution. CLT is typically assumed to hold for samples in excess of 30 observations (Hogg and Tanis (1996)). So even though a $MinTRL$ may demand less than 2.5 years of monthly data, or 0.5769 years of weekly data, or 0.119 years of daily data, etc. the moments inputted in Eq. (13) must be computed on longer series for CLT to hold. This is consistent with practitioners’ standard practice of requiring similar lengths during the due diligence process.

5. NUMERICAL EXAMPLES

Everything we have learnt in the previous sections can be illustrated in a few practical examples. Figure 8 displays the minimum track record lengths ($MinTRL$) in years required for various combinations of measured $\tilde{S}R$ (rows) and benchmarked $SR^\ast$ (columns) at a 95% confidence level, based upon daily IID Normal returns. For example, the fifth column informs us that a 2.73 years track record is required for an annualized Sharpe of 2 to be considered greater than 1 at a 95% confidence level.

[FIGURE 8 HERE]

We ask, what would the $MinTRL$ be for a weekly strategy with also an observed annualized Sharpe of 2? Figure 9 shows that, if we move to weekly IID Normal returns, the requirement is 2.83 years of track record length, a 3.7% increase.

[FIGURE 9 HERE]

Figure 10 indicates that the track record length needed increases to 3.24 years if instead we work with monthly IID Normal returns, an 18.7% increase compared to daily IID Normal returns. This increase in $MinTRL$ occurs despite the fact that both strategies have the same observed annualized Sharpe ratio of 2, and it is purely caused by a decrease in frequency.

[FIGURE 10 HERE]

Let’s stay with monthly returns. Brooks and Kat (2002) report that the HFR Aggregate Hedge Fund returns index exhibits $\hat{p}_3 = -0.72$ and $\hat{p}_4 = 5.78$. In these circumstances, Figure 11 tells us that the track record should now be 4.99 years long. This is 54% longer than what we required with Normal monthly returns, and 82.8% longer than what was needed with Normal daily returns.

[FIGURE 11 HERE]
6. SKILLFUL HEDGE FUND STYLES
We are now ready to run our model on real data. Figure 12 applies our methodology on HFR Monthly indices from January 1st 2000 to May 1st 2011 (134 monthly observations, or 11.167 years). MinTRL is expressed in years, subject to a confidence level of 95%.

A PSR(0) > 0.95 indicates that a SR is greater than 0 with a confidence level of 0.95. Similarly, a PSR(0.5) > 0.95 means that a SR is greater than 0.5 (annualized) with a confidence level of 0.95. The Probabilistic Sharpe ratio has taken into account multiple statistical features present in the track record, such as its length, frequency and deviations from Normality (skewness, kurtosis).

Because our sample consists of 11.167 years of monthly observations, a PSR(0) > 0.95 is consistent with a MinTRL(0) < 11.167 at 95% confidence, and a PSR(0.5) > 0.95 is consistent with a MinTRL(0.5) < 11.167 at 95% confidence. Our calculations show that most hedge fund styles evidence some level of skill, i.e. their SR are above the zero benchmark. However, looking at PSR(0.5), we observe that only 9 style indices substantiate investment skill over an annualized Sharpe ratio of 0.5 at a 95% confidence level:

- Distressed Securities
- Equity Market Neutral
- Event Driven
- Fixed Asset-Backed
- Macro
- Market Defensive
- Mortgage Arbitrage
- Relative Value
- Systematic Diversified

[FIGURE 12 HERE]

This is not to say that only hedge funds practicing the 9 styles listed above should be considered. Our analysis has been performed on indices, not specific track records. However, it could be argued that special care should be taken when analyzing performance from styles other than the 9 mentioned. We would have liked to complete this analysis with a test of structural breaks, however the amount and quality of data does not allow for meaningful estimates.

7. THE SHARPE RATIO EFFICIENT FRONTIER
PSR evaluates the performance of an individual investment in terms of an uncertainty-adjusted SR. It seems natural to extend this argument to a portfolio optimization or capital allocation context. Rather than a mean-variance frontier of portfolio returns on capital, we will build a mean-variance frontier of portfolio returns on risk.

Following Markowitz (1952), a portfolio \( w \) belongs to the Efficient Frontier if it delivers maximum expected excess return on capital \( E[r_{w}] \) subject to the level of uncertainty surrounding those portfolios’ excess returns \( \tilde{\sigma}_{r_{w}} \).
\[
\max_w E[r_w | \sigma_{r_w}] = \sigma^* \tag{14}
\]

Similarly, we define what we denote the Sharpe ratio Efficient Frontier (SEF) as the set of portfolios \( \{w\} \) that deliver the highest expected excess return on risk (as expressed by their Sharpe ratios) subject to the level of uncertainty surrounding those portfolios’ excess returns on risk (the standard deviation of the Sharpe ratio).

\[
\max_w \hat{SR}(r_w) | \hat{\sigma}_{\hat{SR}(r_w)} = \sigma^* \tag{15}
\]

But why would we compute an efficient frontier of Sharpe ratios while accepting that returns \( (r) \) are non-Normal? Because a great majority of investors use the \( SR \) as a proxy for their utility function. Even though they do not care about higher moments per se, they must de-inflate their estimates of \( SR \) (a mean-variance metric) using the third and fourth moments. A number of additional reasons make this analysis interesting:

1. **SEF** deals with efficiency within the return on risk (or Sharpe ratio) space rather than return on capital. Unlike returns on capital, Sharpe ratios are invariant to leverage.
2. Even if returns are non-Normally distributed,
   a. the distribution of Sharpe ratios follows a Normal, therefore an efficient frontier-style of analysis still makes sense.
   b. as long as the process is IID, the cumulative returns distribution asymptotically converges to Normal, due to the Central Limit Theorem.
3. Performance manipulation methods like those discussed by Ingersoll, Spiegel, Goetzmann and Welch (2007) generally attempt to inflate the Sharpe ratio by distorting the returns distribution. As **SEF** considers higher moments, it adjusts for such manipulation.
4. It is a second degree of uncertainty analysis. The standard (Markowitz) portfolio selection framework measures uncertainty in terms of standard deviation on returns. In the case of **SEF**, uncertainty is measured on a function \( (\hat{SR}(r_w)) \) that already incorporates an uncertainty estimate \( (\hat{\sigma}_{r_w}) \). Like in Black-Litterman (1992), this approach does not assume perfect knowledge of the mean-variance estimates, and deals with uncertainty in the model's input variables. This in turn increases the robustness of the solution, which contrasts with the instability of mean-variance optimization (see Best and Grauer (1991)).
5. Computing the **SEF** will allow us to identify the portfolio that delivers the highest PSR for any given \( SR^* \) threshold, thus dealing with non-Normality and sample uncertainty due to track record length in the context of portfolio selection. From Eq. (11), the highest PSR portfolio is the one such that

\[
\arg \max_w \frac{\hat{SR}(r_w)}{\hat{\sigma}_{\hat{SR}(r_w)}} = \arg \max_w \frac{\hat{SR}(r_w) \sqrt{n - 1}}{\sqrt{1 - \hat{\gamma}_3(r_w) \hat{SR}(r_w) + \frac{\hat{\gamma}_4(r_w) - 1}{4} \hat{SR}(r_w)^2}} \tag{16}
\]
A numerical example will clarify this new analytical framework. There exist 43,758 fully invested long portfolios that are linear combinations of the 9 HFR indices identified in the previous section, with weightings

\[ w_i = \frac{j^\gamma}{\sum_{j=1}^{\gamma}} \quad f = 0, \ldots, 10, \quad i = 1, \ldots, 9 \]  

Because non-Normality and sample length impact our confidence on each portfolio's risk-adjusted return, selecting the highest Sharpe ratio portfolio is suboptimal. This is illustrated in Figure 13, where the highest SR portfolio (right end of the SEF) comes at the expense of substantial uncertainty with regards that estimate, since \((\tilde{\sigma}_{\text{SR}(r_w)}, \tilde{R}(r_w)) = (0.155, 0.818)\). The portfolio that delivers the highest PSR is indeed quite different, as marked by the encircled cross \((\tilde{\sigma}_{\text{PSR}(r_w)}, \tilde{R}(r_w)) = (0.103, 0.708)\). Recall that the x-axis in this figure does not represent the risk associated with an investment, but the statistical uncertainty surrounding our estimation of SR.

[FIGURE 13 HERE]

Figure 14 illustrates how the composition of the SEF evolves as \(\tilde{\sigma}_{\text{SR}(r_w)}\) increases. The vertical line at \(\tilde{\sigma}_{\text{PSR}(r_w)} = 0.103\) indicates the composition of the highest PSR portfolio, while the vertical line at \(\tilde{\sigma}_{\text{SR}(r_w)} = 0.155\) gives the composition of the highest SR portfolio. The transition across different regions of the SEF is very gradual, as a consequence of the robustness of this approach.

[FIGURE 14 HERE]

Figure 15 shows why the Max PSR solution is preferable: Although it delivers a lower Sharpe ratio than the Max SR portfolio (0.708 vs. 0.818 in monthly terms), its better diversified allocations allow for a much greater confidence (0.103 vs. 0.155 standard deviations). Max PSR invests in 5 styles, and the largest holding is 30%, compared to the 4 styles and 50% maximum holding of the Max SR portfolio.

[FIGURE 15 HERE]

The Max PSR portfolio displays better statistical properties than the Max SR portfolio, as presented in Figure 16: Max PSR is very close to Normal (almost null skewness and kurtosis close to 3, \(JB \text{ prob} \approx 0.6\)), while the Max SR portfolio features a left fat-tail \(JB \text{ prob} \approx 0\). A risk averse investor should not accept a 17.4% probability of returns being drawn from an adverse distribution in exchange for aiming at a slightly higher Sharpe ratio (Figures 17-18).

[FIGURE 16 HERE]

[FIGURE 17 HERE]
In other words, taking into account higher moments has allowed us to naturally find a better balanced portfolio that is optimal in terms of uncertainty-adjusted Sharpe ratio. We say "naturally" because this result is achieved without requiring constraints on the maximum allocation permitted per holding. The reason is, PSR recognizes that concentrating risk increases the probability of catastrophic outcomes, thus it penalizes such concentration.

8. CONCLUSIONS
A probabilistic translation of Sharpe ratio, called PSR, is proposed to account for estimation errors in an IID non-Normal framework. When assessing Sharpe ratio's ability to evaluate skill, we find that a longer track record may be able to compensate for certain statistical shortcomings of the returns probability distribution. Stated differently, despite Sharpe ratio's well-documented deficiencies, it can still provide evidence of investment skill, as long as the user learns to require the proper track record length.

Even under the assumption of IID returns, the track record length required to exhibit skill is greatly affected by the asymmetry and kurtosis of the returns distribution. A typical hedge fund's track record exhibits negative skewness and positive excess kurtosis, which has the effect of "inflating" its Sharpe ratio. One solution is to compensate for such deficiencies with a longer track record. When that is not possible, a viable option may be to provide returns with the highest sampling frequency such that the IID assumption is not violated. The reason is, for negatively skewed and fat-tailed returns distributions, the number of years required may in fact be lowered as the sampling frequency increases. This has led us to affirm that "badly behaved" returns distributions have the most to gain from offering the greatest transparency possible, in the form of higher data granularity.

We present empirical evidence that, despite the high Sharpe ratios publicized for several hedge fund styles, in many cases they may not be high enough to indicate statistically significant investment skill beyond a moderate annual Sharpe ratio of 0.5 for the analyzed period, confidence level and track record length.

Finally, we discuss the implications that this analysis has in the context of capital allocation. Because non-Normality, leverage and track record length impact our confidence on each portfolio's risk-adjusted return, selecting the highest Sharpe ratio portfolio is suboptimal. We develop a new analytical framework, called the Sharpe ratio Efficient Frontier (SEF), and find that the portfolio of hedge fund indices that maximizes Sharpe ratio can be very different from the portfolio that delivers the highest PSR. Maximizing for PSR leads to better diversified and more balanced hedge fund allocations compared to the concentrated outcomes of Sharpe ratio maximization.
APPENDICES

A.1. HIGHER MOMENTS OF A MIXTURE OF m NORMAL DISTRIBUTIONS

Let $z$ be a random variable distributed as a standard normal, $z \sim N(0, 1)$. Then, $\eta = \mu + \sigma \zeta \sim N(\mu, \sigma^2)$, with characteristic function:

$$0_{\eta}(s) = E[e^{is\eta}] = E[e^{is\mu}]E[e^{is\eta}] = e^{is\mu} \frac{1}{\sqrt{2\pi \sigma^2}} e^{-\frac{s^2\zeta^2}{2\sigma^2}} = e^{is\mu} \frac{1}{\sqrt{2\pi \sigma^2}}$$

Let $r$ be a random variable distributed as a mixture of $m$ normal distributions, $r \sim \sum_{j=1}^{m} p_j (\mu_j, \sigma_j^2)$, with $\sum_{j=1}^{m} p_j = 1$. Then:

$$\vartheta_r(s) = E[e^{isr}] = e^{is\mu} \sum_{j=1}^{m} p_j (e^{is\mu_j} - 1)$$

The $k^{th}$ moment centered about zero of any random variable $x$ can be computed as:

$$E[x^k] = \left. \frac{d^k}{ds^k} \left( \vartheta_x(s) \right) \right|_{s=0}$$

In the case of $r$, the first 5 moments centered about zero can be computed as indicated above, leading to the following results:

$$E[r] = \sum_{j=1}^{m} p_j \mu_j$$
$$E[r^2] = \sum_{j=1}^{m} p_j (\sigma_j^2 + \mu_j^2)$$
$$E[r^3] = \sum_{j=1}^{m} p_j (3\sigma_j^2 \mu_j + \mu_j^3)$$
$$E[r^4] = \sum_{j=1}^{m} p_j (3\sigma_j^4 + 6\sigma_j^2 \mu_j^2 + \mu_j^4)$$
$$E[r^5] = \sum_{j=1}^{m} p_j (15\sigma_j^4 \mu_j + 10\sigma_j^2 \mu_j^3 + \mu_j^5)$$

The first 5 central moments about the mean are computed by applying Newton’s binomium:

$$E[r - E[r]]^j = \sum_{j=0}^{k} (-1)^j \binom{k}{j} (E[r])^j E[r^{k-j}]$$
\[ E[r - E[r]] = 0 \] (27)
\[ E[r - E[r]]^2 = E[r^2] - (\E[r])^2 \] (28)
\[ E[r - E[r]]^2 = E[r^3] - 3E[r^2]E[r] + 2(\E[r])^3 \] (29)
\[ E[r - E[r]]^4 = E[r^4] - 4E[r^3]E[r] + 6E[r^2](E[r])^2 - 3(E[r])^4 \] (30)
\[ E[r - E[r]]^4 = E[r^4] - SE[r^3]E[r] + 10E[r^3](E[r])^2 - 10E[r^2](E[r])^3 + 4(\E[r])^5 \] (31)

### A.2. TARGETING SHARPE RATIO THROUGH A MIXTURE OF TWO NORMAL DISTRIBUTIONS

Suppose that \( r \sim \mathcal{N}(\mu_1, \mu_2, \sigma_1, \sigma_2, p, 1 - p) \). We ask for what value \( p \) the Mixture of two Normal distributions is such that

\[ \frac{E[r]}{\sqrt{E[r - E[r]]^2}} = SR^* \] (32)

where \( SR^* \) is a targeted Sharpe ratio. Setting \( SR^* \) implies that \( p \) will now be a function of the other parameters, \( p = p(\mu_1, \sigma_1, \mu_2, \sigma_2, SR^*) \). In this section we will derive that function \( f \).

From Eq. (32), \( (\E[r])^2 = SR^2E[r - E[r]]^2 \). Applying Eq. (28), this expression simplifies into

\[ (\E[r])^2(1 + SR^*^2) = SR^2E[r^2] \] (33)

From Eq. (21) and Eq. (22),

- \( (E[r])^2 = (\mu_1 + \mu_2(1 - p))^2 = \rho^2(\mu_1^2 + \mu_2^2 - 2\mu_1\mu_2) + 2\rho(\mu_1\mu_2 - \mu_1^2) + \mu_2^2 \)
- \( E[r^2] = (\sigma_1^2 + \mu_1^2)\rho + (\sigma_2^2 + \mu_2^2)(1 - \rho) = \rho(\sigma_1^2 + \mu_1^2 - \sigma_2^2 - \mu_2^2) + \sigma_2^2 + \mu_2^2 \)

Let \( \alpha = \mu_1 + \mu_2 - 2\mu_1\mu_2 \) and \( \beta = 1 + \frac{1}{SR^*^2} \). Then, Eq. (33) can be rewritten as

\[ (p^2\alpha + 2p(\mu_1\mu_2 - \mu_2^2) + \mu_2^2)\beta = \rho(\sigma_1^2 + \mu_1^2 - \sigma_2^2 - \mu_2^2) + \sigma_2^2 + \mu_2^2 \] (34)

which can be reduced into

\[ p^2\alpha\beta + \rho(2\beta(\mu_1\mu_2 - \mu_2^2) - \mu_1^2 - \sigma_1^2 + \mu_2^2 + \sigma_2^2) + \mu_2^2(\beta - 1) - \sigma_2^2 = 0 \] (35)
For \( a = \alpha \beta, b = 2\beta (\mu_1 \mu_2 - \mu_2^2) - \mu_1 - \sigma^2 + \mu_1 + \sigma |, c = \mu_2^2 \beta - 1) - \sigma | \), Eq. (35) leads to the monic quadratic equation

\[
p^2 + p \frac{b}{a} + \frac{c}{a} = 0
\]

with solution in

\[
p^* = \frac{1}{2} \left( -\frac{b}{a} \pm \sqrt{\left(\frac{b}{a}\right)^2 - 4 \frac{c}{a}} \right)
\]

where

- \( a = \left(1 + \frac{1}{SR^2}\right)(\mu_1^2 + \mu_2^2 - 2\mu_1\mu_2)\)
- \( b = 2 \left(1 + \frac{1}{SR^2}\right)(\mu_1\mu_2 - \mu_2^2) - \mu_1^2 - \sigma^2 + \mu_2^2 + \sigma_2^2\)
- \( c = \mu_2^2 - \sigma_2^2\)

Let’s discuss now the condition of existence of the solution: In order to be a probability, solutions with an imaginary part must be discarded, which leads to the condition that

\[b^2 \geq 4ac\]

Furthermore, because in Eq. (33) we squared both sides of the equality, \( p^* \) could deliver

\[
\frac{E[r]}{\sqrt{E[r]-E[r^2]}} = -SR^*.
\]

So a second condition comes with selecting the root \( p^* \) such that

\[
Sgn(SR^*) = S_g n(\rho^* \mu_1 + (1 - \rho^*) \mu_2)
\]

Finally, in order to have \( 0 \leq p^* \leq 1 \), it is necessary that either

\[
\frac{\mu_2}{\sigma_2} \geq SR^* \geq \frac{\mu_1}{\sigma_1} \quad \text{or} \quad \frac{\mu_1}{\sigma_1} \geq SR^* \geq \frac{\mu_2}{\sigma_2}
\]

This result allows us to simulate a wide variety of non-Normal distributions delivering the same targeted Sharpe ratio (\( SR^* \)).
The implementation in Python and MinTTR calls are implemented in the following code...

```python
#!/usr/bin/env python
# PSR class for computing the Probabilistic Sharpe Ratio
# On 20120502 by MJJP <lopezdeprado@lbl.gov>

from scipy.stats import norm

#--------------------------------------------------------
# PSR class
class PSR:
    def __init__(self, stats, sr_ref, obs, prob):
        self.PSR = 0
        self.minTRL = 0
        self.stats = stats
        self.sr_ref = sr_ref
        self.obs = obs
        self.prob = prob

    def set_PSR(self, moments):
        stats = self.stats['moments'] + [0 for i in range(len(self.stats['moments']))]
        sr = self.stats[0]/self.stats[1]
        self.PSR = norm.cdf((sr-self.sr_ref)*self.obs**0.5/\n                             (1-sr*stats[2]+sr**2*(stats[3]-1)/4.)*0.5)

    def set_TRL(self, moments):
        stats = self.stats['moments'] + [0 for i in range(len(self.stats['moments']))]
        sr = self.stats[0]/self.stats[1]
        self.minTRL = 1+(1-stats[2]*sr+(stats[3]-1)/4.*sr**2)*\n                        (norm.ppf(self.prob)/(sr-self.sr_ref))**2

    def get_PSR(self, moments):
        self.set_PSR(moments)
        return self.PSR

    def get_TRL(self, moments):
        self.set_TRL(moments)
        return self.minTRL

#--------------------------------------------------------
# Main function
def main():
    # Inputs (stats on excess returns)
    stats=[2,12**0.5,-0.72,5.78] #non-annualized stats
    sr_ref=1/12**0.5 #reference Sharpe ratio (non-annualized)
    obs=59.895
```

59.895, which gives value of $\text{PSR} \left( \frac{1}{\sqrt{12}} \right) = 0.95$. 

prob=0.95

#2) Create class
psr=PSR(stats,sr_ref,obs,prob)

#3) Compute and report values
print 'PSR(2m,3m,4m):',[psr.get_PSR(i) for i in \range(2,5,1)]
print 'minTRI(2m,3m,4m):',[psr.get_TRI(i) for i in \range(2,5,1)]

# Boilerplate
if __name__ == '__main__': main()
REFERENCES


An infinite number of mixtures of two Gaussians can deliver any given SR, despite of having widely different levels of skewness and kurtosis. This is problematic, because high readings of SR may come from extremely risky distributions, like combinations on the left side of this figure (negative skewness and positive kurtosis).
Figure 2(a) - Probability density function for a Mixture of two Gaussians with parameters \((\mu_1, \mu_2, \sigma_1, \sigma_2, \rho) = (-5, 1.05, 5, 0.05, 0.015)\)
These two distributions were drawn from the combinations plotted in Figure 1. Both have a Sharpe ratio of 1, despite of their evidently different risk profile. The dashed black line represents the probability distribution function of a Normal distribution fitted of each of these mixtures. The variance not only underestimates non-Normal risks, but its own estimator is greatly affected by non-Normality. A minimal change in the mixture's parameters could have a great impact on the estimated value of the mixture's variance.

Figure 2(b) - Probability density function for a Mixture of two Gaussians with parameters $(\mu_1, \mu_2, \sigma_1, \sigma_2, \rho) = (0.3237, 1.8816, 0.05, 0.05, 0.8706)$
Figure 3(a) - True vs. estimated mean

Figure 3(b) - True vs. estimated standard deviation
Estimations errors increase with higher moments, requiring longer sample sizes.
If we draw samples from random mixtures of two Gaussians, we can study how the estimation errors on their moments are affected by the moment's values.

The standard deviation of the SR estimator is sensitive to skewness and kurtosis. For $SR=1$, we see that $\sigma^{3/4}$ is particularly sensitive to skewness, as we could expect from inspecting Eq. (8).
This mixture of two Gaussians exactly matches the moments reported in Figure 6. The dash line shows that a Normal fit severely underestimates the downside risks for this portfolio manager. Moreover, there is a significant probability that this portfolio manager may have no investment skill, despite of having produced an annualized Sharpe ratio close to 1.6.
<table>
<thead>
<tr>
<th>Observed Sharpe</th>
<th>0</th>
<th>0.5</th>
<th>1</th>
<th>1.5</th>
<th>2</th>
<th>2.5</th>
<th>3</th>
<th>3.5</th>
<th>4</th>
<th>4.5</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.5</td>
<td>1</td>
<td>1.83</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2.71</td>
<td>1.85</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.5</td>
<td>1.21</td>
<td>2.72</td>
<td>1.87</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>.69</td>
<td>1.22</td>
<td>2.73</td>
<td>1.91</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.5</td>
<td>.44</td>
<td>.69</td>
<td>1.22</td>
<td>2.74</td>
<td>1.96</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>.31</td>
<td>.44</td>
<td>.69</td>
<td>1.23</td>
<td>2.76</td>
<td>11.2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3.5</td>
<td>.23</td>
<td>.31</td>
<td>.45</td>
<td>.7</td>
<td>1.24</td>
<td>2.78</td>
<td>11.9</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>.18</td>
<td>.23</td>
<td>.31</td>
<td>.45</td>
<td>.7</td>
<td>1.24</td>
<td>2.8</td>
<td>11.7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4.5</td>
<td>.14</td>
<td>.18</td>
<td>.23</td>
<td>.32</td>
<td>.45</td>
<td>.71</td>
<td>1.25</td>
<td>2.82</td>
<td>11.26</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>.12</td>
<td>.14</td>
<td>.18</td>
<td>.24</td>
<td>.32</td>
<td>.46</td>
<td>.71</td>
<td>1.27</td>
<td>2.84</td>
<td>11.36</td>
</tr>
</tbody>
</table>

Figure 8 - Minimum track record in years, under daily IID Normal returns

<table>
<thead>
<tr>
<th>Observed Sharpe</th>
<th>0</th>
<th>0.5</th>
<th>1</th>
<th>1.5</th>
<th>2</th>
<th>2.5</th>
<th>3</th>
<th>3.5</th>
<th>4</th>
<th>4.5</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.5</td>
<td>1</td>
<td>1.87</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2.75</td>
<td>1.95</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.5</td>
<td>1.25</td>
<td>2.78</td>
<td>11.8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>.72</td>
<td>1.27</td>
<td>2.83</td>
<td>11.26</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.5</td>
<td>.48</td>
<td>.74</td>
<td>1.29</td>
<td>2.89</td>
<td>11.49</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>.35</td>
<td>.49</td>
<td>.75</td>
<td>1.33</td>
<td>2.96</td>
<td>11.78</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3.5</td>
<td>.27</td>
<td>.36</td>
<td>.5</td>
<td>.78</td>
<td>1.38</td>
<td>3.4</td>
<td>12.12</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>.21</td>
<td>.27</td>
<td>.37</td>
<td>.52</td>
<td>.8</td>
<td>1.41</td>
<td>3.14</td>
<td>12.51</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4.5</td>
<td>.18</td>
<td>.22</td>
<td>.28</td>
<td>.38</td>
<td>.54</td>
<td>.83</td>
<td>1.46</td>
<td>3.25</td>
<td>12.95</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>.15</td>
<td>.18</td>
<td>.23</td>
<td>.29</td>
<td>.39</td>
<td>.56</td>
<td>.86</td>
<td>1.51</td>
<td>3.38</td>
<td>13.44</td>
</tr>
</tbody>
</table>

Figure 9 - Minimum track record in years, under weekly IID Normal returns

<table>
<thead>
<tr>
<th>Observed Sharpe</th>
<th>0</th>
<th>0.5</th>
<th>1</th>
<th>1.5</th>
<th>2</th>
<th>2.5</th>
<th>3</th>
<th>3.5</th>
<th>4</th>
<th>4.5</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.5</td>
<td>1</td>
<td>11.2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2.9</td>
<td>11.38</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.5</td>
<td>1.4</td>
<td>3.4</td>
<td>11.92</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>.87</td>
<td>1.49</td>
<td>3.24</td>
<td>12.71</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.5</td>
<td>.63</td>
<td>.94</td>
<td>1.6</td>
<td>3.49</td>
<td>13.72</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>.5</td>
<td>.68</td>
<td>1.1</td>
<td>1.74</td>
<td>3.8</td>
<td>14.96</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3.5</td>
<td>.42</td>
<td>.54</td>
<td>.74</td>
<td>1.1</td>
<td>1.9</td>
<td>4.17</td>
<td>16.43</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>.37</td>
<td>.45</td>
<td>.58</td>
<td>.8</td>
<td>1.21</td>
<td>2.9</td>
<td>4.59</td>
<td>18.12</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4.5</td>
<td>.33</td>
<td>.4</td>
<td>.49</td>
<td>.64</td>
<td>.88</td>
<td>1.33</td>
<td>2.3</td>
<td>5.7</td>
<td>2.4</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>.3</td>
<td>.36</td>
<td>.43</td>
<td>.53</td>
<td>.7</td>
<td>.97</td>
<td>1.46</td>
<td>2.54</td>
<td>5.61</td>
<td>22.18</td>
</tr>
</tbody>
</table>

Figure 10 - Minimum track record in years, under monthly IID Normal returns
Figure 11 - Minimum track record in years, under monthly IID returns
with $f_3 = -0.72$ and $f_4 = 5.78$

<table>
<thead>
<tr>
<th>Observed Sharpe</th>
<th>0</th>
<th>0.5</th>
<th>1</th>
<th>1.5</th>
<th>2</th>
<th>2.5</th>
<th>3</th>
<th>3.5</th>
<th>4</th>
<th>4.5</th>
</tr>
</thead>
<tbody>
<tr>
<td>True Sharpe Ratio</td>
<td>0.5</td>
<td>12.3</td>
<td>3.62</td>
<td>14.23</td>
<td>1.93</td>
<td>4.24</td>
<td>16.7</td>
<td>1.31</td>
<td>2.26</td>
<td><strong>4.99</strong></td>
</tr>
<tr>
<td>2.5</td>
<td>1.1</td>
<td>1.53</td>
<td>2.88</td>
<td>5.88</td>
<td>23.26</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>0.84</td>
<td>1.17</td>
<td>1.79</td>
<td>3.11</td>
<td>6.9</td>
<td>27.35</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3.5</td>
<td>0.73</td>
<td>0.97</td>
<td>1.36</td>
<td>2.8</td>
<td>3.63</td>
<td>8.6</td>
<td>31.98</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>0.66</td>
<td>0.84</td>
<td>1.11</td>
<td>1.57</td>
<td>2.4</td>
<td>4.2</td>
<td>9.35</td>
<td>37.15</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4.5</td>
<td>0.61</td>
<td>0.75</td>
<td>0.96</td>
<td>1.27</td>
<td>1.79</td>
<td>2.76</td>
<td>4.84</td>
<td>1.78</td>
<td>42.85</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>0.57</td>
<td>0.69</td>
<td>0.85</td>
<td>1.18</td>
<td>1.44</td>
<td>2.4</td>
<td>3.15</td>
<td>5.53</td>
<td>12.34</td>
<td>49.9</td>
</tr>
</tbody>
</table>

Figure 12 - Performance analysis on HFR Monthly indices

Only a few hedge fund investment styles evidence skill beyond a Sharpe ratio of 0.5 with a confidence level of 95%.
A Sharpe ratio Efficient Frontier can be derived in terms of optimal mean-variance combinations of risk-adjusted returns.
We can compute the capital allocations that deliver maximum Sharpe ratios for each confidence level. The difference with Markowitz's Efficient Frontier is that SEF is computed on risk-adjusted returns, rather than returns on capital.

**Figure 14 - Composition of the SEF for different $\tilde{\delta}_{SR(r_w)}$ values**

<table>
<thead>
<tr>
<th>HFR Index</th>
<th>Code</th>
<th>Max PSR</th>
<th>Max SR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dist Secur</td>
<td>HFRIDSI Index</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Equity Neutral</td>
<td>HFRIEMNI Index</td>
<td>0</td>
<td>0.2</td>
</tr>
<tr>
<td>Event Driven</td>
<td>HFRIEDI Index</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Fixed Asset-Back</td>
<td>HFRIFIMB Index</td>
<td>0.3</td>
<td>0.5</td>
</tr>
<tr>
<td>Macro</td>
<td>HFRIMI Index</td>
<td>0.1</td>
<td>0</td>
</tr>
<tr>
<td>Mkt Defens</td>
<td>HFRIFOFM Index</td>
<td>0.2</td>
<td>0</td>
</tr>
<tr>
<td>Mrg Arbit</td>
<td>HFRIMAI Index</td>
<td>0.3</td>
<td>0.2</td>
</tr>
<tr>
<td>Relative Value</td>
<td>HFRIRVA Index</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Sys Diversified</td>
<td>HFRIMTI Index</td>
<td>0.1</td>
<td>0.1</td>
</tr>
</tbody>
</table>

**Figure 15 - Composition of the Max PSR and Max SR portfolios**
Maximum PSR portfolios are risk-adjusted optimal, while maximum SR portfolios are risk-adjusted suboptimal. The reason is, although a maximum SR portfolio may be associated with a high expected Sharpe ratio (point estimate), the confidence bands around that expectation may be rather wide. Consequently, maximum PSR portfolios are distributed closer to a Normal, and demand a lower MinTRL than maximum SR portfolios.
Figure 18(a) - Mixture of Normal distributions that recover the first four moments for the Max PSR
Figure 18(b) - Mixture of Normal distributions that recover the first four moments for the Max SR.
DISCLAIMER

The views expressed in this paper are those of the authors and not necessarily reflect those of Tudor Investment Corporation. No investment decision or particular course of action is recommended by this paper.
The strategy approval decision: A Sharpe ratio indifference curve approach

David H. Bailey; Marcos Lopez de Prado; Eva del Pozo

DOI: 10.3233/AF-13018

Abstract, HTML, and PDF:
http://algorithmicfinance.org/2-1/pp99-109

Aims and Scope  Algorithmic Finance is a high-quality academic research journal that seeks to bridge computer science and finance, including high frequency and algorithmic trading, statistical arbitrage, momentum and other algorithmic portfolio management strategies, machine learning and computational financial intelligence, agent-based finance, complexity and market efficiency, algorithmic analysis on derivatives, behavioral finance and investor heuristics, and news analytics.

Managing Editor
Philip Maymin, NYU-Polytechnic Institute

Deputy Managing Editor
Jayaram Muthuswamy, Kent State University

Advisory Board
Kenneth J. Arrow, Stanford University
Herman Chernoff, Harvard University
David S. Johnson, AT&T Labs Research
Leonid Levin, Boston University
Myron Scholes, Stanford University
Michael Sipser, Massachusetts Institute of Technology
Richard Thaler, University of Chicago
Stephen Wolfram, Wolfram Research

Editorial Board
Associate Editors
Peter Bossaerts, California Institute of Technology
Emanuel Derman, Columbia University
Ming-Yang Kao, Northwestern University
Pete Kyle, University of Maryland
David Leinweber, Lawrence Berkeley National Laboratory
Richard J. Lipton, Georgia Tech
Avi Silberschatz, Yale University
Robert Webb, University of Virginia

Affiliate Editors
Giovanni Barone-Adesi, University of Lugano
Bruce Lehmann, University of California, San Diego

Subscription, submission, and other info:
www.iospress.nl/journal/algorithmic-finance

Unique Features of the Journal  Open access: Online articles are freely available to all. No submission fees: There is no cost to submit articles for review. There will also be no publication or author fee for at least the first two volumes. Authors retain copyright: Authors may repot their versions of the papers on preprint archives, or anywhere else, at any time. Enhanced content: Enhanced, interactive, computable content will accompany papers whenever possible. Possibilities include code, datasets, videos, and live calculations. Comments: Algorithmic Finance is the first journal in the Financial Economics Network of SSRN to allow comments. Archives: The journal is published by IOS Press. In addition, the journal maintains anarchy o.g SSRN.c.gra. Legal: While the journal does reserve the right to change these features at any time without notice, the intent will always be to provide the world's most freely and quickly available research on algorithmic finance. ISSN: Online ISSN: 2157-6203. Print ISSN: 2158-5571.

379
The strategy approval decision: A Sharpe ratio indifference curve approach

David H. Bailey\textsuperscript{a}, Marcos Lopez de Prado\textsuperscript{b}, and Eva del Pozo\textsuperscript{c}

\textsuperscript{a}Complex Systems, Lawrence Berkeley National Laboratory, Berkeley, CA, USA
E-mail: dhailey@lbl.gov

\textsuperscript{b}Global Quantitative Research - Tudor Investment Corporation; Lawrence Berkeley National Laboratory, Berkeley, CA, USA
E-mail: lopezdeprado@lbl.gov

\textsuperscript{c}Mathematical Finance, Universidad Complutense de Madrid, Madrid, Spain
E-mail: epozo@ccce.ucm.es

Abstract. The problem of capital allocation to a set of strategies could be partially avoided or at least greatly simplified with an appropriate strategy approval decision process. This paper proposes such a procedure. We begin by splitting the capital allocation problem into two sequential stages: strategy approval and portfolio optimization. Then we argue that the goal of the second stage is to beat a naive benchmark, and the goal of the first stage is to identify which strategies improve the performance of such a naive benchmark. We believe that this is a sensible approach, as it does not leave all the work to the optimizer, thus adding robustness to the final outcome.

We introduce the concept of the Sharpe ratio indifference curve, which represents the space of pairs (candidate strategy's Sharpe ratio, candidate strategy's correlation to the approved set) for which the Sharpe ratio of the expanded approved set remains constant. We show that selecting strategies (or portfolio managers) solely based on past Sharpe ratio will lead to suboptimal outcomes, particularly when we ignore the impact that these decisions will have on the average correlation of the portfolio. Our strategy approval theorem proves that, under certain circumstances, it is entirely possible for firms to improve their overall Sharpe ratio by hiring portfolio managers with negative expected performance. Finally, we show that these results have important practical business implications with respect to the way investment firms hire, layoff and structure payouts.

JEL classifications: C02, Gil, G14, D53.

Keywords: Portfolio theory, Sharpe ratio, pairwise correlation, indifference curve, diversification, free call option.

1. Introduction

The problem of allocating capital to Portfolio Managers (PMs) or strategies is typically addressed using a variation of Markowitz's (1952) approach.

The views expressed in this publication are the authors' and do not necessarily reflect the opinion of Tudor Investment Corporation. We would like to thank the Managing Editor of Algorithmic Finance, Philip Maymin (New York University-Polytechnic Institute), as well as two anonymous referees, for their insightful comments during the peer-review process. We are grateful to Tudor Investment Corporation, Marco Avellaneda (Courant Institute of Mathematical Sciences, New York University), Jose A. Blanco (UBS), Peter Carr (Morgan Stanley, New York University), Jose A. Gil Fana (Universidad Complutense de Madrid), David Leinweber (Lawrence Berkeley National Laboratory), Attilio Meucci (Kepos Capital, New York University), Riccardo Rebonato (PIMCO, University of Oxford), Jose M. Roboos (Universidad de Santiago de Compostela), Piedad Tolmos (Universidad Juan Carlos I), Luis Viceira (Harvard Business School) and Jose L. Vilar Zanon (Universidad Complutense de Madrid).

Supported in part by the Director, Office of Computational and Technology Research, Division of Mathematical, Information, and Computational Sciences of the U.S. Department of Energy, under contract number DE-AC02-05CH11231.

2158-5571/13/$27.50 © 2013 - IOS Press and the authors. All rights reserved
This method is agnostic as to the criterion employed to pre-select those PMs. In this paper we will show that the standard procedure used by the investment management industry to hire and layoff PMs may indeed lead to suboptimal capital allocations.

In a series of papers, Sharpe (1966, 1975, 1994) introduced a risk-adjusted measure of investment's performance. This measure, universally known as the Sharpe ratio (SR), has become the gold-standard to evaluate PMs in the investment management industry. It is well known that the Sharpe ratio is the right selection criterion if the investor is restricted to picking only one investment, i.e. when maximum concentration is mandated (Bodie et al., 1995). However, the Sharpe ratio is not necessarily a good criterion when a sequence of individual decisions must be made, such as hiring an additional PM or adding an investment strategy to an existing fund. Most firms address this sequential decision making problem by requiring any candidate manager or strategy to pass several fixed thresholds, including SR (De Souza & Gokcan, 2004) and track record length (Bailey & Lopez de Prado, 2012). Among the PMs or strategies that have passed those thresholds, capital is then allocated following an optimization process.

This implies that the capital allocation process is in practice composed of two distinct stages: approval and optimization. The problem is, these two stages are carried out independently, potentially leading to incoherent outcomes. Selecting a strategy because it passes a certain SR threshold ignores the candidate strategy's correlation to the set of already existing strategies. The consequence of this incoherence between the approval and optimization stages is that the overall outcome of the capital allocation process may be suboptimal, regardless of the optimization applied in the second stage. For example, approving a strategy with lower SR may introduce more diversification than another strategy with higher SR but also higher correlation to the approved set.

Before considering a candidate strategy for approval, it is critical to determine not only its expected SR, but also its average correlation against the approved set of strategies. A first goal of this paper is to demonstrate that there is no fixed SR threshold which we should demand for strategy approval. We must define an Approval benchmark which jointly looks at the candidate's SR and how it fits in the existing menu of strategies. This benchmark must be naive, in the sense that it is pre-optimization. Equal Volatility Weighting is a procedure that has been used in past to benchmark portfolio optimization results (for example, DeMiguel et al., 2009), and this is the one we adopt in our framework.

A second goal of this paper is to formalize the trade-off between a candidate's SR and its correlation to the existing set of strategies, a concept we call the Sharpe ratio indifference curve. We will often find situations in which a highly performing candidate strategy should be declined due to its high average correlation with the existing set. Conversely, a low performing candidate strategy may be approved because its diversification potential offsets the negative impact on the average SR. Looking at the combined effect that a candidate's SR and correlation will have on the approved set also addresses a fundamental critique to the "fixed SR threshold" approach currently applied by most investment firms: Such fixed threshold tends to favor higher over lower frequency strategies. But considering the (low) correlation that the latter strategies have with respect to the former, lower frequency strategies will have a fairer chance of being approved under the new approach hereby presented. Our strategy approval theorem proves that, under certain circumstances, it is entirely possible for firms to improve their overall SR by hiring PMs with negative expected performance.

A third goal of this paper is to explain how this new strategy approval decision process could lead to new business arrangements in the investment management industry. Emulating the performance of "star-PMs" through a large number of uncorrelated low-SR PMs creates the opportunity for investment firms to internalize features that cannot be appropriated by the individual PMs.

It is worth noting that our use of the term benchmark should not be interpreted in the sense of Jensen (1968), Sortino and van der Meer (1991), Treynor (1966) or Treynor and Black (1973). Our motivation is to define a high performance level which, being the result of a naive procedure, must be subsequently improved by any optimization. Tobin (1958) proposed separating the portfolio construction problem into two steps: optimization of risky assets and the amount of leverage. Like Tobin's, our approach also separates the allocation problem into two stages, but as we will see, both methods are substantively different.

Our results are applicable to a wide range of firms faced with the problem of hiring PMs and allocating them capital, including hedge funds, funds of hedge funds, proprietary trading firms, mutual funds, etc. Although some features specific to these investment
vehicles could be integrated in our analysis (e.g., non-Normality of returns, lock-up periods, rebalance frequency, liquidity constraints, etc.), we have not done so in order to keep the framework as generally applicable as possible.

The rest of this paper is structured as follows: Section 2 presents a few propositions on the naive benchmark’s performance. Section 3 introduces the concept of the SR indifference curve (strategy approval theorem). Section 4 makes a specific proposal for the process of approving strategies. Section 5 suggests a new business arrangement based on strategy approval process. Section 6 summarizes our conclusions. The appendices present the mathematical proofs to these propositions.

2. Propositions

Capital allocation to PMs or strategies is typically done without consideration of the strategy approval process or hiring criteria (L’Habitant, 2004). This means that the portfolio optimization step may have to deal with PMs or strategies pre-selected according to a set of rules that lead to suboptimal capital allocations, like hiring PMs with investment strategies similar to those already in the platform. This paper is dedicated to show how to approve strategies (or hiring) in a manner that is consistent with the goal of portfolio optimization. We do so by dividing the capital allocation problem into two sequential stages. First, we define a naive benchmark that must be raised at each hiring. Second, we optimize a portfolio composed of strategies that have passed that naive benchmark. The output of the second step must beat the output of the first (naive) step. In this way, we avoid leaving the entire decision to optimization techniques that have been criticized for their lack of robustness Best and Grauer (1991).

The following propositions differ from standard portfolio theory (including Tobin (1958)) in a number of ways:

1. They discuss the allocation of capital across strategies or PMs, rather than assets.
2. They are based on the establishment of an Equal Volatility Weightings (or naive) benchmark.
3. This benchmark allows us to split the capital allocation problem into two sequential sub-problems:

   a. **Strategy Approval**: The process by which a candidate strategy is approved to be part of a portfolio.
   b. **Portfolio Optimization**: The process that determines the optimal amount of capital to be allocated to each strategy within a portfolio.

4. Key principles of the approach discussed here are:
   a. The goal of the Portfolio Optimization process is to beat the performance of a naive benchmark.
   b. The goal of the Strategy Approval process is to raise the performance of the naive benchmark as high as possible (ideally, to the point that no portfolio optimization is required at all!).

2.1. Benchmark portfolio

2.1.1. Statement

The performance of an Equal Volatility Weights benchmark (SRB) is fully characterized in terms of:

1. Number of approved strategies ($S$).
2. Average SR among strategies (SR).
3. Average off-diagonal correlations among strategies ($\rho$).

In particular, adding strategies ($S$) with the same $SR$ and $\rho$ does improve $SRB$.

Sections A.1 and A.2. prove this statement.

2.1.2. Example

Following Eq. (7) in Appendix 2, it will take 16 strategies with $\overline{SR} = 0.75$ and $\rho = 0.2$ to obtain a benchmark with SR of 1.5. Should the average individual risk-adjusted performance decay to $\overline{SR} = 0.5$, the benchmark’s SR will drop to 1. Figure 1 illustrates the point that, if on top of that performance degradation the individual pairwise correlation raises to $\rho = 0.3$, the benchmark’s SR will be only 0.85.

2.1.3. Practical implications

This proposition allows us to estimate the benchmark’s SR without requiring knowledge of the individual strategy’s SRs or their pairwise correlations. Average volatility is not a necessary input. All that is needed is $S$, $\overline{SR}$ and $\rho$. This makes possible the simulation of performance degradation or correlation stress-test scenarios, as illustrated in the previous epigraph.
2.2. On performance degradation

2.2.1. Statement

The benchmark SR is a linear function of the average SR of the individual strategies, and a decreasing convex function of the number of strategies and the average pairwise correlation. This means that, as the number of strategies (S) increases, favoring low \( \rho \) offers a convex payoff which SR does not. In the presence of performance degradation, low correlated strategies may be preferable to (supposedly) highly performing ones.

Section A.3 proves this statement.

2.2.2. Example

Figure 2(a) shows the benchmark’s SR as a linear function of SR for 5 and 25 strategies. Figure 2(b) shows the benchmark’s SR as a convex function of \( \rho \) for 5 and 25 strategies (see Eqs. (8)-(9) in Appendix 3).

2.2.3. Practical implications

This is a critical result. It implies that we may prefer low correlated strategies, even if underperforming, to outperforming but highly correlated strategies. The exact trade-off between these two characteristics will become clearer in Section 3.

2.3. On the maximum achievable benchmark SR

2.3.1. Statement

There is a limit to how much the benchmark SR can be improved by adding strategies. In particular, that limit is fully determined by:

1. Average SR among strategies (\( \overline{SR} \)),
2. Average off-diagonal correlations among strategies (\( \rho \)).

Section A.4 proves this statement.

2.3.2. Example

Suppose that \( \overline{SR} = 0.75 \) and \( \rho = 0.2 \). Regardless of how many equivalent strategies are added, the benchmark’s SR will not exceed 1.68 (Fig. 3). Higher SRs could still be obtained with a skillful (non-naive) portfolio optimization process, but are beyond the benchmark’s reach (see Eqs. (11)-(12) in Appendix 4).

2.3.3. Practical implications

In the absence of \( \overline{SR} \) degradation, it would make little sense increasing the number of strategies (S) beyond a certain number. But since \( \overline{SR} \) degradation is expected, there is a permanent need for building an inventory of replacement strategies (to offset for those decommissioned due to performance degradation or approval error (false positive)). This is consistent with Proposition 2, which offered a theoretical justification for researching as many (low correlated) strategies as possible (the convex payoff due to correlation).

2.4. On the impact of a candidate strategy on the benchmark’s SR

2.4.1. Statement

A strategy being considered for approval would have an impact on the benchmark’s SR (and thus its naive targeted performance) that exclusively depends on:

1. Number of approved strategies (\( S \)),
2. Average SR among strategies (\( \overline{SR} \)).
Fig. 2(a). Benchmark SR as a function of $\bar{S}R$. Figure 2(a) demonstrates the linear impact that $\bar{S}R$ has on the benchmark SR.

Fig. 2(b). Benchmark SR as a function of $\bar{p}$. Figure 2(b) demonstrates the convex impact that $\bar{p}$ has on the benchmark SR.

3. Average off-diagonal correlations among strategies ($\bar{p}$).
4. Average correlation of the candidate strategies against the approved set ($p_{s+i}$).
5. The candidate strategy’s SR ($SR_{s+i}$).

Section A.5 proves this statement.

2.4.2. Example

Suppose it is the case that $S = 2$, $\bar{S}R = 1$, $p = 0.1$, thus $SR_{\bar{p}} = 1.35$. Consider a third strategy with $S_{\bar{p}} = 1$ and $p = 0.1$. Then, applying Eq. (13) in Appendix 5, $SR_{\bar{p}} = 1.58$. Adding the third strategy positively impacted the benchmark’s SR, even though there was no improvement on $\bar{S}R = 1$, $p = 0.1$. We knew this from Proposition 1.

Let’s turn now to the case where $S_{\bar{p}} = 0.7$ and $\bar{p} = 0.1$. Then, $S_{\bar{p}} = 1.42$. If $SR_{\bar{p}} = 0.7$ but $\bar{p} = 0.2$, then $SR_{\bar{p}} = 1.35$. Note how adding a “worsening” strategy (which lowers $\bar{S}R$ and increases $p$) nevertheless did not reduce $SR_{\bar{p}}$, thanks to the diversification gains. We were able to calculate these
scenarios without requiring additional knowledge regarding the strategy's risk or pairwise correlations.

2.4.3. Practical implications

Proposition 4 shows that $p_{s+i}$ and $SR_{s+i}$ suffice to determine the new benchmark's SR. In particular, we do not need to know each pairwise correlation, individual SRs or strategies' volatilities, which greatly simplifies simulation exercises.

3. The SR indifference curve (strategy approval theorem)

The previous propositions converge into the following fundamental result.

3.1. Statement

There exists a trade-off such that we would be willing to accept a strategy with below average SR if its average correlation to the approved set is below a certain level. This determines an indifference curve as a function of:

1. Number of approved strategies ($S$).
2. Average SR ($SR$).
3. Average off-diagonal correlations ($p$).
4. The candidate strategy's SR ($SR_{s+i}$).
5. The SR of the benchmark portfolio ($SR_B$).

where the acceptance threshold in terms of correlation is

$$P_{s+i} = \frac{1}{2} \left[ \left( \frac{SR_B \cdot S + SR_{s+i}}{SR_B} \right)^2 - \frac{5 + 1}{5} - p(S - 1) \right]$$  \hspace{1cm} (1)

Section A.6 proves this statement.

3.2. Example

Suppose the same case as in Proposition 4, namely that $S = 2$, $SR_B = 1$, $p = 0.1$. Thus $SR_B = 1.35$. A third strategy with $5 \frac{3}{4} = 1$ and $p = 0.1$ would lead to $SR_B = 1.58$. The theorem says that, should an alternative third strategy deliver $5 \frac{3}{4} = 1.5$ instead, we would be indifferent with $p = 0.425$ (see Eq. (1)). Beyond that correlation threshold, the alternative with higher SR (i.e., $5 \frac{3}{4} = 1.5$) should be declined. Figure 4 shows the entire indifference curve.

More interestingly, we would also be indifferent to a second alternative whereby $5 \frac{3}{4} = -0.1$ and $p = -0.439$. But why would we ever approve a strategy that very likely will not make any money? Why would a firm hire a PM that loses money? This probably sounds counter-intuitive, but that's where the previous math becomes helpful. The reason is, we are investing in 3 strategies. Overall, we will still have a quite positive

![Figure 3. SR of a portfolio of approved strategies with $\overline{SR} = 0.75$ and $\rho = 0.2$. Figure 3 shows how the benchmark Sharpe ratio increases as a result of new strategies being added (keeping constant $\overline{SR} = 0.75$ and $\rho = 0.2$).](image-url)
return. True that this overall return would be *slightly* greater without the third strategy, however without it the standard deviation would also be *much* larger. All things considered, if that third strategy incorporates an average correlation below \( \hat{p}_3 = -0.439 \), it improves the overall SR beyond \( SR_B = 1.58 \). In this particular example, the third strategy would behave like a call option at a premium equivalent to the \( \sigma_3SR \) it costs (in terms of returns) to "buy" it. Naturally, strategies with a \( \hat{p}_3 = -0.439 \) average correlation may be hard to find, but if they presented themselves, we should consider them.

Finally, suppose that the first of the three alternatives is added (\( SR_A = 1, \hat{p}_3 = 0.1 \)). This will in turn shift the indifference curve to the left and up (see Fig. 5). It means that pairs of \( (SR_A+i, P_s+i) \) that fall between the red and the blue curve have now become acceptable. As the set of approved strategies pools more risk, it is able to clear room for previously rejected strategies without reducing the benchmark's overall SR.

### 3.3. Practical implications

For every candidate strategy, there exists an infinite group of alternative theoretical candidates whereby all deliver the same benchmark SR. The indifference curve represents the exact trade-off between a candidate strategy's SR and its average correlation against the approved set such that the benchmark's SR is preserved.

This theorem addresses a problem faced by most investment firms: The "fixed SR threshold" strategy selection criterion represents a considerable hurdle for the lower frequency strategies. These strategies tend to have a lower annualized SR, but they also bring lower average correlations, with an overall improvement in diversification. This approach finds the balance between both components, allowing for low frequency strategies to be acceptable under an objective set of requirements.

### 4. Proposal for a coherent strategy approval process

Before considering a candidate strategy for approval, it is critical to determine not only its expected SR, but also its average correlation against the approved set of strategies. The above results imply that there is *no fixed SR threshold which we should demand*.
for strategy approval. We must jointly look at the candidate’s SR and how it fits in the existing menu of strategies by considering:

1. Number of approved strategies (S).
2. Average SR among strategies (SR).
3. Average off-diagonal correlations among strategies (β).
4. The candidate strategy’s SR (SR+i).
5. Average correlation of the candidate strategies against the approved set (ps+i).

A realistic backtest would reflect transaction costs and market impact when estimating SR+i, thus incorporating a capacity penalty in this analysis. With these inputs we can then compute SRB (without the candidate strategy), SRB+i (including the candidate strategy), and given SR+i for what ps+i it occurs that SRB+i = SRB+i (indifference point).

We will often find situations in which a highly performing candidate strategy should be declined due to its high average correlation with the existing set. Conversely, a low performing candidate strategy may be approved because its diversification potential offsets the negative impact on the average SR.

It is important to note that the input variables do not need to be restricted to historical estimates, but can reflect forward looking scenarios. This makes it possible to reset approval thresholds under alternative assumptions on capacity, future correlation, performance degradation, etc.

5. Business implications

Far from being solely a theoretical result, these propositions have a number of very practical business implications. Funds typically pay PMs a percentage of the net profits generated by their strategies. PMs do not share a percentage of the losses, which gives them exposure to the upside only. Funds are therefore writing what is called a "free call option" (Gregorion et al., 2011; Ineichen, 2003; L’habitant, 2004). The true value of the option is proportional to the risks associated with a PM’s strategy. The better the PM’s strategy, the lower the probability of losses, therefore the cheaper the option given away by the Fund. Conversely, the option offered to an unskilled PM is extremely expensive. For this reason, funds do not evaluate a PM’s performance in terms of average annual return, as that would not take into account the risks involved and would lead to offering the option to the wrong PMs.

The core argument presented in this paper - that SR is a misleading index of whom a fund should hire or
fire - seems at odds with standard business practices. The SR indifference curve shows that even PMs with a negative individual SR should be hired if they contribute enough diversification. Why is that not the case? Because of a netting problem: a typical business agreement is that PMs are entitled to a percentage of their individual performance, not a percentage of the fund's performance. Legal clauses may release the fund from having to pay a profitable PM if the overall fund has lost money, however that PM is unlikely to remain at the firm after a number of such events. This is a very unsatisfactory situation, for a number of reasons: First, funds are giving up the extra-performance predicted by the SR indifference curve. Second, funds are compelled to hire 'star-PMs', who may require a high portion of the performance fee. Third, funds are always under threat of losing to competitors their 'star-PMs', who may leave the firm with their trade secrets for a slightly better deal. In some firms, PMs' turnover is extremely high, with an average tenure of only one or two years.

A way to avoid this suboptimal outcome is to offer a business deal that pays the PM a percentage of the fund's overall performance. This would again create some tensions, as some 'star-PMs' could do better with their individual deals. However, Section 2.3 tells us that we can emulate the performance of a 'star-PM' by hiring a sufficient number of 'average-PMs' with low correlation to the fund's performance. A first advantage of doing so is that 'average-PMs' have no bargaining power, thus we can pay them a lower proportion of the performance fee. A second advantage is that, because of the relatively low SR, they are unlikely to be poached. A third advantage is that, if we hire 'average-PMs' whose performances have low correlation to the fund's, we can internalize a private value to which the PMs have no access (they cannot leave and "take" the low correlation to the firm with them). The average-PM's performance may exhibit a low correlation to a limited number of funds, but not to all. In other words, the fund can capture the extra-performance postulated by the SR indifference curve without having to pay for it.

A future can therefore be envisioned in which investment firms structure payments with the following features:

- Payout is arranged in terms of funds' overall performance, which may be superior to that of 'star-driven' funds.
- The hiring process targets PMs with relatively low SRs (in some cases even below zero, if their correlation is sufficiently negative), and therefore cheaper to find, keep and replace.
- There is a very low turnover of PMs, as they cannot take the 'low correlation to the fund' with them, and their low SR does not get them an individual deal.

This kind of business arrangement is particularly suitable to firms that engage in algorithmic strategies, because the prerequisite of 'sufficient number of average-PMs' can be easily fulfilled with average-performing trading systems. Since the SR required to put each system in production will be relatively low, they can be developed in large numbers. As long as each quant developer is involved in a limited number of those systems, their bargaining power will still be limited.

6. Conclusions

Ideally, if an investment firm counted with virtually uncorrelated strategies, no optimization would be required at all. Although an unrealistic scenario, it is nonetheless true that many of the problems associated with portfolio optimization could be avoided, to a great extent, with a proper procedure of strategy approval. The procedure discussed in this paper goes in that direction.

We have divided the capital allocation problem in two sequential phases: strategy approval and portfolio optimization. The goal of the strategy approval phase is to raise the naive benchmark's performance, reducing the burden typically placed on the portfolio optimization phase. We have demonstrated that there is no fixed SR threshold that we should demand for strategy approval. Instead, there is an indifference curve of pairs (candidate strategy's SR, candidate's correlation to approved set) that keep the benchmark's SR constant. At the extreme, it may be preferable to approve a candidate's strategy with negative Sharpe if its correlation to the approved set is sufficiently negative.

These results are particularly relevant in the context of performance degradation, as they demonstrate that selecting strategies (or PMs) solely based on past SR may lead to suboptimal results, especially when we ignore the impact that these decisions will have on the average correlation of the portfolio. The practical implication is that firms could emulate the performance of 'star-PMs' through uncorrelated low-SR PMs, who will not have individual bargaining
power. This theoretical framework justifies setting up a legal compensation structure based on overall fund performance, which internalizes a private value to which the PMs have no access, namely their low correlation to the platform.

Appendix

A.1. Definitions and standing hypothesis

Suppose a collection of \( S \) strategies, jointly-distributed as a multivariate Normal. The marginal distribution of excess returns is

\[
 r_s \sim N(\mu_s, \sigma_s^2), \text{ for } s = 1, \ldots, S \tag{2}
\]

The portfolio composed of these \( S \) strategies, characterized by a vector of weightings \( \{\omega_s\} \), has excess returns \( r \), which follow a distribution

\[
 r - N \left( \sum_{s=1}^{S} \omega_s \mu_s, \sum_{x=1}^{S} \omega_x^2 \sigma_x^2 + 2 \sum_{l=s+1}^{S} \omega_l \omega_l \rho_{s,l} \right) \tag{3}
\]

The SR for such portfolio can then be computed as

\[
 SR = \frac{\sum_{s=1}^{S} \omega_s \mu_s}{\sqrt{\sum_{s=1}^{S} \left( \omega_s^2 \sigma_s^2 + 2 \sum_{l=s+1}^{S} \omega_l \omega_l \rho_{s,l} \right)}} \tag{4}
\]

We would like to investigate the variables that affect the risk-adjusted performance of such portfolio of strategies.

A.2. Benchmark portfolio

Let’s set the benchmark portfolio to be the result of a naive equal volatility weighting allocation,

\[
 \omega_s = \frac{1}{\sigma_s}, \text{ for } s = 1, \ldots, S \tag{5}
\]

Then, it is immediate to show that the SR of this benchmark portfolio is

\[
 SR_B = \frac{S}{\sqrt{1 + \frac{2}{S} \sum_{s=1}^{S} \sum_{t=s+1}^{S} \rho_{s,t}}} \tag{6}
\]

where \( SR = \frac{\sum_{s=1}^{S} \mu_s}{\sqrt{\sum_{s=1}^{S} \sigma_s^2 + \sum_{i=2}^{\infty} \frac{\sigma_{si}^2}{i!} \rho_{si}^i}} \), the average SR across the strategies. The average correlation across off-diagonal elements is \( \rho = \frac{\sum_{i=2}^{\infty} \frac{\sigma_{si}^2}{i!} \rho_{si}^i}{\sum_{i=2}^{\infty} \frac{\sigma_{si}^2}{i!}} \). We can compute the SR of the benchmark portfolio as

\[
 SR_{B} = \frac{S}{\sqrt{1 + (S - 1) \rho}} \tag{7}
\]

A.3. Sensitivity to performance degradation

Let’s compute the partial derivative of Eq. (7) with respect to \( SR \) and \( \rho \)

\[
 \frac{\partial SR_E}{\partial SR} = \frac{S}{1 + (S - 1) \rho} \tag{8}
\]

\[
 \frac{\partial SR_B}{\partial \rho} = -\frac{S - 1}{2} \frac{\sqrt{S(1 + (S - 1) \rho)}} {1 + (S - 1) \rho} \tag{9}
\]

\[
 \frac{\partial SR_B}{\partial S} = -\frac{S - 1}{2} \frac{\sqrt{S(1 + (S - 1) \rho)}} {1 + (S - 1) \rho} \tag{10}
\]

Therefore, \( SR_B \) is a linear function of the average performance degradation, but a decreasing convex function of the average correlation increase.

A.4. Diversification

It is interesting to discuss diversification in the context of this benchmark portfolio because we do not assume a skillful capital allocation process. If the capital allocation process is skillful and \( \rho < 1 \), then the portfolio’s Sharpe ratio (SR) will surely beat the benchmark’s (\( SR_B \)). However, if \( \rho = 1 \), then \( SR = SR_B = SR_R \) and the capital allocation process cannot benefit from diversification.

We apply Taylor’s expansion on Eq. (7) with respect to \( S \), to the first order.

\[
 ASR_B = \frac{\partial SR_B}{\partial S} \Delta S + \sum_{i=2}^{\infty} \frac{\partial^i SR_B}{\partial S^i} \frac{\Delta S^i}{i!} \tag{11}
\]

Only when \( \rho = 0 \), the SR can be expanded without limit by increasing \( S \). But otherwise, SR gains become gradually smaller until eventually \( SR_B \) converges to
the asymptotic limit

$$\lim_{S \to \infty} SR_B = \frac{\overline{SR}}{\sqrt{p}}$$  \hspace{1cm} (12)$$

A.5. Impact of candidate strategies on the benchmark

Equation (12) tells us that the maximum SR for the benchmark portfolio is a function of two variables: The average Sharpe ratio among strategies ($\overline{SR}$) and the average off-diagonal correlation ($p$). It also shows that we should accept a below-average SR strategy if it adds diversification. Going back to Eq. (7), the value of $SR_B$ after adding a new strategy can be updated as

$$SR^*_B = \frac{(\overline{SR}, S + SR_{S+1})}{\sqrt{S(p(S - 1) + 2p_{S+1} + 1) + 1}}$$  \hspace{1cm} (13)$$

where

- $SR_{S+1}$ is the SR associated with the candidate strategy.
- $p_{S+1} = \frac{1}{S} \sum_{s=1}^{S} p_{s,S+1}$.
- $P_{s,S+1}$ are the pairwise correlations between the candidate strategy and the set of $S$ approved strategies.

A.6. Indifference curve and strategy approval

From Eq. (13), we can isolate an indifference curve for preserving the benchmark’s SR, i.e. that imposes the condition $SR^*_B = SR_B$.

$$P_{s+1} = \frac{1}{2} \left[ \frac{(\overline{SR}, S + SR_{S+1})^2}{S \cdot SR^*_B} \right]$$

$$- \frac{5}{S} + \frac{1}{S} - p(S - 1)$$  \hspace{1cm} (14)$$

This in turn leads to

$$\frac{dP_{s+1}}{dSR^*_B} = - \frac{\overline{SR} \cdot S + SR_{S+1}}{S \cdot SR^*_B}$$  \hspace{1cm} (15)$$

And inserting Eq. (7) we derive the equilibrium condition,

$$\frac{dSR_{s+1}}{dSR^*_B} = \frac{(\overline{SR} \cdot S + SR_{S+1}) (1 + (S - 1)p_{s,S+1})}{S^2 \cdot SR^*_B^2}$$  \hspace{1cm} (16)$$

References


THE CLAIMS

What is claimed is:

1. A computer-implemented system for automatically generating financial investment portfolios, comprising:
   - an online crowdsourcing site comprising one or more servers and associated software that configures the servers to provide the crowdsourcing site and further comprising a database of open challenges and historic data, wherein on the servers, the site:
     - registers experts, accessing the site from their computers, to use the site over a public computer network,
     - publishes challenges on the public computer network wherein the challenges include challenges that define needed individual scientific forecasts for which forecasting algorithms are sought,
     - implements an algorithmic developer's sandbox that comprises:
       - individual private online workspaces that are available remotely accessible for use to each registered expert and which include a partitioned integrated development environment comprising online access to:
         - algorithm development software,
         - historic data,
         - forecasting algorithm evaluation tools including one or more tools for performing test trials using the historic data, and
         - a process for submitting one of the expert's forecasting algorithms authored in their private online workspace to the system as a contributed forecasting algorithm for inclusion in a forecasting algorithm portfolio;
     - an algorithm selection system comprising one or more servers and associated software that configures the servers to provide the algorithm selection system, wherein on the servers, the algorithm selection system:
       - receives the contributed forecast algorithms from the algorithmic developer's sandbox,
       - monitors user activity inside the private online workspaces including user activity related to the test trials performed within the private online
workspaces on the contributed forecasting algorithms before the contributed forecasting algorithms were submitted to the system,

determines, from the monitored activity, test related data about the test trials performed in the private online workspaces on the contributed forecasting algorithms including identifying a specific total number of times a trial was actually performed in the private online workspace on the contributed forecasting algorithm by the registered user,

determines accuracy and performance of the contributed forecasting algorithms using historical data and analytics software tools including determining, from the test related data, a corresponding probability of backtest overfitting associated with individual ones of the contributed forecasting algorithms, and

based on determining accuracy and performance, identifying a subset of the contributed forecasting algorithms to be candidate forecasting algorithms;

an incubation system comprising one or more servers and associated software that configures the servers to provide the incubation system, wherein on the servers, the incubation system:

receives the candidate forecasting algorithms from the algorithm selection system,

determines an incubation time period for each of the candidate forecasting algorithms by receiving the particular probability of backtest overfitting for the candidate forecasting algorithms and receiving minimum and maximum ranges for the incubation time period,

in response, determining a particular incubation period that varies between the maximum and minimum period based primarily on the probability of backtest overfitting associated with that candidate forecasting algorithm, whereby certain candidate forecasting algorithms will have a much shorter incubation period than others;

includes one or more sources of live data that are received into the incubation system,

applies the live data to the candidate forecasting algorithms for a period of time specified by corresponding incubation time periods,
determines accuracy and performance of the candidate forecasting algorithms in response to the application of the live data including by determining accuracy of output values of the candidate forecast algorithms when compared to actual values that were sought to be forecasted by the candidate forecasting algorithms, and

in response to determining accuracy and performance of the candidate forecasting algorithms, identifies and stores a subset of the candidate forecasting algorithms as graduate forecasting algorithms as a part of a portfolio of operational forecasting algorithms that are used to forecast values in operational systems.

2. The system of claim 1, wherein the system implements a source control system that tracks iterative versions of individual forecast algorithms while the forecast algorithms are authored and modified by users in their private workspace.

3. The system of claim 2, wherein the system determines test related data about test trials performed in the private workspace in specific association with corresponding versions of an individual forecasting algorithm, whereby the algorithm selection system determines the specific total number of times each version of the forecasting algorithm was tested by the user who authored the forecasting algorithm.

4. The system of claim 2, wherein the system determines the probability of backtest overfitting using information about version history of an individual forecast algorithm as determined from the source control system.

5. The system of claim 2, wherein the system associates a total number of test trials performed by users in their private workspace in association with a corresponding version of the authored forecasting algorithm by that user.

6. The system of claim 5, wherein the system determines, from the test data about test trials including a number of test trials and the association of some of the test trials with different versions of forecast algorithms, the corresponding probability of backtest overfitting.
7. The system of claim 1, wherein the system includes a fraud detection system that receives and analyzes contributed forecasting algorithms and determines whether some of the contributed forecasting algorithms demonstrate fraudulent behavior.

8. The system of claim 1, wherein the online crowdsourcing site applies an authorship tag to contributed forecasting algorithm and the system maintains the authorship tag in connection with the contributed forecasting algorithm including as part of a use of the contributed forecasting algorithm as a graduate forecasting algorithm in operation use.

9. The system of claim 8, wherein the system determines corresponding performance of graduate algorithms and generates an output in response to the corresponding performance that is communicated to the author identified by the authorship tag.

10. The system of claim 9, wherein the output communicates a reward.

11. The system of claim 1, wherein the system further comprises a ranking system that ranks challenges based on corresponding difficulty.

12. The system of claim 1, wherein the algorithm selection system includes a financial translator that comprises different sets of financial characteristics that are associated with specific open challenges, wherein the algorithm selection system determines a financial outcome from at least one of the contributed forecasting algorithms by applying the set of financial characteristics to the at least one of the contributed forecast algorithms.

13. The system of claim 1 further comprising a portfolio management system comprising one or more servers, associated software, and data that configure the servers to implement the portfolio management system, wherein on the servers, the portfolio management system:
   - receives graduate forecasting algorithms from the incubation system,
   - stores graduate forecasting algorithms in a portfolio of graduate forecasting algorithms,
   - applies live data to the graduate forecasting algorithms and in response receives output values from the graduate forecasting algorithms,
determines directly or indirectly, from individual forecasting algorithms and their corresponding output values, specific financial transaction orders, and transmits the specific financial transaction orders over a network to execute the order.

14. The system of claim 13 wherein the portfolio management system comprises at least two operational modes, wherein in a first mode, the portfolio management system processes and applies graduate forecasting algorithms that are defined to have an output that is a financial output and the portfolio management system determines from the financial output the specific financial order.

15. The system of claim 14 wherein the portfolio management system comprises a second mode, and in the second mode, the portfolio management system processes and applies graduate forecasting algorithm that are defined to have an output that is a scientific output, applies a financial translator to the scientific output, and the portfolio management system determines from the output of the financial translator a plurality of specific financial orders that when executed generate or modify a portfolio of investments that are based on the scientific output.

16. The system of claim 13 wherein the portfolio management system is further configured to:

   evaluate actual performance outcomes for graduate forecasting algorithms against expected or predetermined threshold performance outcomes for corresponding graduate forecast algorithm,

   based on the evaluation, determine underperforming graduate forecasting algorithms,

   remove underperforming graduate forecasting algorithms from the portfolio, and

   communicate actual performance outcomes, the removal of graduate algorithms, or a status of graduate forecasting algorithms to other components in the computer-implemented system.
17. The system of claim 13 wherein the portfolio management system:
evaluates performance of graduate forecasting algorithms by performing a
simulation after live trading is performed that varies input values and determines
variation in performance of the graduate forecasting algorithm portfolio in response to
the varied input values, and
determines from the variations in performance to which ones of the graduate
forecasting algorithms in the portfolio the variations should be attributed.

18. The system of claim 1 wherein the algorithm selection system is further
configured to include a marginal contribution component that:
determines a marginal forecasting power of a contributed forecasting
algorithm,
by comparing the contributed forecasting algorithm to a portfolio of graduate
forecasting algorithm operating in production in live trading,
determines based on the comparison a marginal value of the contributed
forecasting algorithm with respect to accuracy, performance, or output diversity when
compared to the graduate forecasting algorithms, and
in response the algorithm selection system (in response to itself?) determines
which contributed forecasting algorithm should be candidate forecasting algorithm
based at least partly on the marginal value.

19. The system of claim 1 wherein the algorithm selection system is further
configured to include a scanning component that scans contributed forecasting algorithms
and in scanning searches for different contributed forecasting algorithms that are mutually
complementary.

20. The system of claim 19 wherein the scanning component determines a subset
of the contributed forecasting algorithms that have defined forecast outputs that do not
overlap.

21. The system of claim 1 wherein the incubation system further comprises a
divergence component that:
receives and evaluates performance information related to candidate
forecasting
algorithm,
over time, determines whether the performance information indicates that individual candidate forecasting algorithm systems have diverged from in sample performance values determined prior to the incubation system, and terminates the incubation period for candidate forecasting algorithm that have diverged from their in-sample performance value by a certain threshold.

22. A computer-implemented system for automatically generating financial investment portfolios, comprising:
an online crowdsourcing site comprising one or more servers and associated software that configures the servers to provide the crowdsourcing site and further comprising a database of challenges and historic data, wherein on the servers, the site:
publishes challenges to be solved by users,
implements a development system that comprises:
individual private online workspaces to be used by the users comprising online access to:
algorithm development software for solving the published challenges to create forecasting algorithms,
historic data,
forecasting algorithm evaluation tools for performing test trials using the historic data, and
a process for submitting the forecasting algorithms to the computer-implemented system as contributed forecasting algorithms;
an algorithm selection system comprising one or more servers and associated software that configures the servers to provide the algorithm selection system, wherein on the servers, the algorithm selection system:
receives the contributed forecast algorithms from the development system,
determines a corresponding probability of backtest overfitting associated with individual ones of the received contributed forecasting algorithms, and
based on the determined corresponding probability of backtest overfitting, identifies a subset of the contributed forecasting algorithms to be candidate forecasting algorithms;

an incubation system comprising one or more servers and associated software that configures the servers to provide the incubation system, wherein on the servers, the incubation system:

receives the candidate forecasting algorithms from the algorithm selection system,

determines an incubation time period for each of the candidate forecasting algorithms,

applies live data to the candidate forecasting algorithms for a period of time specified by corresponding incubation time periods,

determines accuracy and performance of the candidate forecasting algorithms in response to the application of the live data, and

in response to determining accuracy and performance of the candidate forecasting algorithms, identifies and stores a subset of the candidate forecasting algorithms as graduate forecasting algorithms as a part of a portfolio of operational forecasting algorithms that are used to forecast values in operational systems.

23. A computer-implemented system for automatically generating financial investment portfolios, comprising:

a site comprising one or more servers and associated software that configures the servers to provide the site and further comprising a database of challenges, wherein on the servers, the site:

publishes challenges to be solved by users,

implements a first system that comprises:

individual workspaces to be used by the users comprising access to:

algorithm development software for solving the published challenges to create forecasting algorithms, and
a process for submitting the forecasting algorithms to the computer-implemented system as contributed forecasting algorithms;

a second system comprising one or more servers and associated software that configures the servers to provide the second system, wherein on the servers, the second system:

- evaluates the contributed forecast algorithms, and
- based on the evaluation, identifies a subset of the contributed forecasting algorithms to be candidate forecasting algorithms;

a third system comprising one or more servers and associated software that configures the servers to provide the third system, wherein on the servers, the third system:

- determines a time period for each of the candidate forecasting algorithms,
- applies live data to the candidate forecasting algorithms for corresponding time periods determined,
- determines accuracy and performance of the candidate forecasting algorithms in response to the application of the live data, and
- based on the determination of accuracy and performance, identifies a subset of the candidate forecasting algorithms as graduate forecasting algorithms, the graduate forecasting algorithms are a part of a portfolio of operational forecasting algorithms that are used to forecast values in operational systems.

24. A computer implemented system for developing forecasting algorithms, comprising:

- a crowdsourcing site which is open to the public and publishes open challenges for solving forecasting problems; wherein the site includes individual private online workspace including development and testing tools used to develop and test algorithms in the individual workspace and for users to submit their chosen forecasting algorithm to the system for evaluation;
- a monitoring system that monitors and records information from each private workspace that encompasses how many times a particular algorithm or its different versions were tested by the expert and maintains a record of algorithm development,
wherein the monitoring and recording is configured to operate independent of control or modification by the experts;

  a selection system that evaluates the performance of submitted forecasting algorithms by performing backtesting using historic data that is not available to the private workspaces, wherein the selection system selects certain algorithms that meet required performance levels and for those algorithms, determines a probability of backtest overfitting and determines from the probability, a corresponding incubation period for those algorithm that varies based on the probability of backtest overfitting.

25. The system of claim 1 further comprising a portfolio management system that comprises a quantum computer configured with software that together processes graduate forecasting algorithms and indirect cost of associated financial activity and in response determines modifications to financial transaction orders before being transmitted, wherein the portfolio management system modifies financial transaction orders to account for overall profit and loss evaluations over a period of time.

26. The system of claim 13 wherein the portfolio management system comprises a quantum computer that is configured with software that together processes graduate forecasting algorithms by generating a range of parameter values for corresponding financial transaction orders, partitioning the range, associating each partition with a corresponding state of a qubit, evaluating expected combinatorial performance of multiple algorithms overtime using the states of associated qubits, and determining as a result of the evaluating, the parameter value in the partitioned range to be used in the corresponding financial transaction order before the corresponding financial transaction order is transmitted for execution.
FIG. 1
FIG. 11

- Candidate algorithms
- Paper Trading
- Graduate algorithms
- Backtests + due diligence
- Algo Traders
- Portfolio Opt. Overlay
- Continued algorithm
- Performance attribution
Quality Assessment Info.

- Challenge Solved
- Data used for test or assessment
- Result (very good, good, poor, etc.)
Hello Steve Allerd,

You are assigned 2 triggered incidents in PagerDuty. Please visit the following URL to manage these incidents.
https://liquid-strategies.pagerduty.com/dashboard

1) Incident #588
   Opened on: Mar 5 at 8:40am EST
   Service: Prod
   Description: FW: NJ1TSTLQSA02 20150305 08:40:47 file check on E:/apps/Strategies/Strategy1/Strategy1EU/bin/20150305/Failed
   Link: http://liquid-strategies.pagerduty.com/i/508
   Escalation Policy: round robin
   Details:

From: GISAlerts@acmepartners.com
Sent: Thursday, March 05, 2015 8:40:47 AM (UTC-05:00) Eastern Time (US & Canada)
To: Allerd, Steve; Blue, Jared; Perry, Anna; Viva, George; Line, Mark
Subject: NJ1TSTLQSA02 20150305 08:40:47 file check on E:/apps/Strategies/Strategy1/Strategy1EU/bin/20150305/TICKS/20150
NJ1TSTLQSA02 20150305 08:40:47 file check on E:/apps/Strategies/Strategy1/Strategy1EU...

2) Incident #509
   Opened on: Mar 5 at 8:45am EST
   Service: Prod
   Description: FW: NJ1TSTLQSA02 20150305 08:45:20 Healthcheck on Failed

FIG. 20
## Incidents

<table>
<thead>
<tr>
<th>#</th>
<th>Created On</th>
<th>Details</th>
<th>Service</th>
<th>Assigned To</th>
<th>Status</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>510</td>
<td>Mar 6, 2015 at 5:37 PM</td>
<td>Description: F/W: NJ1TSTLQSAAP02 20150306 17:37:12 process check on Pricing exe ... /config/afgEU/bx Failed</td>
<td>Prod</td>
<td>--</td>
<td>Resolved</td>
<td>Details</td>
</tr>
<tr>
<td>509</td>
<td>Mar 5, 2015 at 8:45 AM</td>
<td>Description: F/W: NJ1TSTLQSAAP02 20150305 08:45:20 Healthcheck on Failed</td>
<td>Prod</td>
<td>--</td>
<td>Resolved</td>
<td>Details</td>
</tr>
</tbody>
</table>

**Assigned to me**

**All open incidents**

0 triggered 0 acknowledged

**Your open incidents**

0 triggered 0 acknowledged

**Level 2:**
- Mar 16, 2015 at 12:00 AM - Mar 23, 2015 at 12:00 AM
- Round Robin

**Level 1:**
- Mar 16, 2015 at 12:00 AM - Mar 23, 2015 at 12:00 AM
- on-call all the time

**Who is on-call now?**
- Anna

**Level 1:** Anna
- on-call all the time
**Level 2:** Jared Blue
- Mar 5, 2015 at 12:00 AM - Mar 9,
<table>
<thead>
<tr>
<th>File</th>
<th>File Name</th>
<th>Folder Path</th>
<th>Frequency</th>
<th>Start Time</th>
<th>End Time</th>
<th>Status</th>
<th>Timestamp</th>
</tr>
</thead>
<tbody>
<tr>
<td>14</td>
<td>USDNOK Prices</td>
<td>E:/data pricing/us/2015036_USDNOK_Currency_TAS.csv</td>
<td>frequency</td>
<td>08:30:30</td>
<td>17:00:00</td>
<td>OK</td>
<td>2015-03-06</td>
</tr>
<tr>
<td>15</td>
<td>AUDUSD Prices</td>
<td>E:/data pricing/us/2015036_AUDUSD_Currency_TAS.csv</td>
<td>frequency</td>
<td>08:30:30</td>
<td>17:00:00</td>
<td>OK</td>
<td>2015-03-06</td>
</tr>
<tr>
<td>16</td>
<td>GDX Prices</td>
<td>E:/data pricing/us/2015036_GDX_US_Equity_TAS.csv</td>
<td>frequency</td>
<td>09:30:30</td>
<td>16:00:00</td>
<td>Out-Of-Monitor-Time</td>
<td>2015-03-06</td>
</tr>
<tr>
<td>17</td>
<td>XAU Currency</td>
<td>E:/data pricing/us/2015036_XAU_Currency_TAS.csv</td>
<td>frequency</td>
<td>08:30:30</td>
<td>17:00:00</td>
<td>OK</td>
<td>2015-03-06</td>
</tr>
<tr>
<td>18</td>
<td>XOP Prices</td>
<td>E:/data pricing/us/2015036_XOP_Equity_TAS.csv</td>
<td>frequency</td>
<td>09:30:30</td>
<td>16:00:00</td>
<td>Out-Of-Monitor-Time</td>
<td>2015-03-06</td>
</tr>
</tbody>
</table>

FIG. 22
System - Deployment

<table>
<thead>
<tr>
<th>Component</th>
<th>Date</th>
<th>Version</th>
<th>Deploy</th>
</tr>
</thead>
<tbody>
<tr>
<td>analytics</td>
<td>150102 16:21:19</td>
<td>2015.01.02.f</td>
<td></td>
</tr>
<tr>
<td>bf</td>
<td>150303 16:46:12</td>
<td>2015.03.03.b</td>
<td></td>
</tr>
<tr>
<td>coint</td>
<td>150220 06:55:23</td>
<td>2015.02.11.b</td>
<td></td>
</tr>
<tr>
<td>config</td>
<td>141018 21:34:05</td>
<td>2014.10.18.b</td>
<td></td>
</tr>
</tbody>
</table>
Welcome to the Intraweb of ACME INSURANCE SERVICES Liquid Strategies

<table>
<thead>
<tr>
<th>Team Members</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>John Long</td>
<td>200-555-8040</td>
<td></td>
</tr>
<tr>
<td>Mark Line</td>
<td>200-555-7509</td>
<td>600-555-5643</td>
</tr>
<tr>
<td>Steve Allerd</td>
<td>200-555-5498</td>
<td>600-555-5674</td>
</tr>
<tr>
<td>Jared Blue</td>
<td>200-555-9543</td>
<td>600-555-5674</td>
</tr>
<tr>
<td>Anna Perry</td>
<td>200-555-7874</td>
<td>600-555-5644</td>
</tr>
<tr>
<td>Maria Quent</td>
<td>200-555-9654</td>
<td>600-555-3548</td>
</tr>
<tr>
<td>George Verma</td>
<td>200-555-6684</td>
<td>600-555-5645</td>
</tr>
</tbody>
</table>
A. CLASSIFICATION OF SUBJECT MATTER
IPC(8) - G06Q 10/06 (2015.01)
CPC - G06Q 10/06 (2015.04)
According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED
Minimum documentation searched (classification system followed by classification symbols)
IPC(8) - G06Q 10/06; 40/04; 40/06 (2015.01)
USPC - 705/37; 706/13; 706/45

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched
CPC - G06Q 10/06; 40/04; 40/06 (2015.04) (keyword delimited)

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
Orbit, Google Patents, Google Scholar, Google.
Search terms used; crowdsourcing, financial, algorithm

C. DOCUMENTS CONSIDERED TO BE RELEVANT

<table>
<thead>
<tr>
<th>Category</th>
<th>Citation of document, with indication, where appropriate, of the relevant passages</th>
<th>Relevant to claim No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>US 2012/0226629 A1 (PURI) 06 September 2012 (06.09.2012) entire document</td>
<td>1-26</td>
</tr>
</tbody>
</table>

Further documents are listed in the continuation of Box C.

* Special categories of cited documents:
  "A" document defining the general state of the art which is not considered to be of particular relevance
  "E" earlier application or patent but published on or after the international filing date
  "L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
  "O" document referring to an oral disclosure, use, exhibition or other means
  "P" document published prior to the international filing date but later than the priority date claimed

T later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"&" document member of the same patent family

Date of the actual completion of the international search: 14 June 2015
Date of mailing of the international search report: 06 JUL 2015

Name and mailing address of the ISA/US
Mail Stop PCT, Attn: ISA/US, Commissioner for Patents
P.O. Box 1450, Alexandria, Virginia 22313-1450
Facsimile No. 571-273-8300

Authorized officer: Blaine R. Copenheaver
PCT Helpdesk: 571-272-4200
PCT OSP: 571-272-7774

Form PCT/ISA/2.10 (second sheet) (July 2009)