



(51) International Patent Classification:
G06F 21/54 (2013.01)

(21) International Application Number:

PCT/US2016/066233

(22) International Filing Date:

12 December 2016 (12.12.2016)

(25) Filing Language:

English

(26) Publication Language:

English

(30) Priority Data:

62/265,910 10 December 2015 (10.12.2015) US
15/166,207 26 May 2016 (26.05.2016) US

(71) Applicant: **AYASDI, INC.** [US/US]; 4400 Bohannon Drive, Suite 200, Menlo Park, California 94025 (US).

(72) Inventors: **YEUNG, Jean-Ezra**; 737 Post Street, No. 714, San Francisco, California 94109 (US). **BAK, Anthony**; 2062 12th Avenue, San Francisco, California 94116 (US).

(74) Agents: **KLOKE, Daniel C.** et al.; Sheppard, Mullin, Richter & Hampton LLP, 379 Lytton Avenue, Palo Alto, California 94301 (US).

(81) Designated States (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States (unless otherwise indicated, for every kind of regional protection available): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK,

[Continued on next page]

(54) Title: DETECTION OF FRAUD OR ABUSE

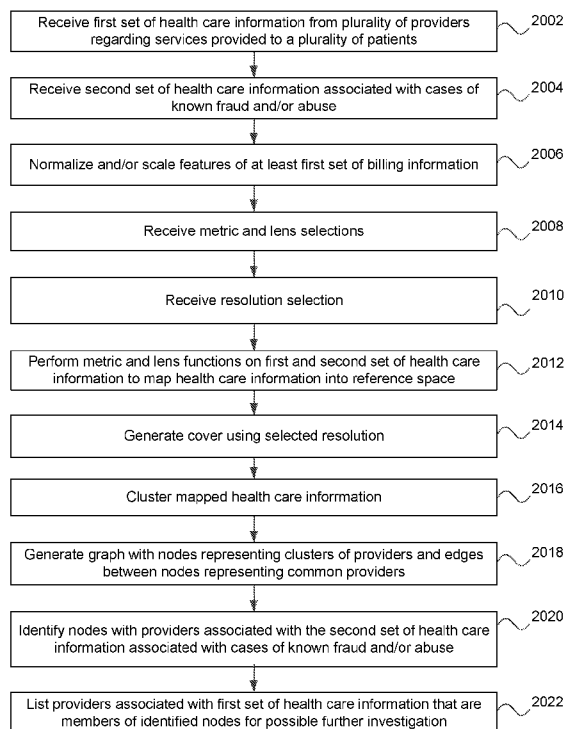


FIG. 20

(57) Abstract: An example method includes receiving a first set of data identifying entities and performance information for analysis, receiving a second set of data identifying entities and performance information associated with known or suspected past fraud or abuse, receiving metric and lens selections, performing metric and lens functions based on the metric and lens selections on first and second set of data, generating cover of reference space and cluster mapped performance information to identify nodes in a graph, each node including one or more entities as members, each node being connected to another node if they share at least one common entity as members, identifying nodes that include at least one member from the second set of data, determining entities that are members of the identified nodes that are from the first set of data, and generating a first report listing the determined entities as possibly involved in fraud or waste.



SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, —
GW, KM, ML, MR, NE, SN, TD, TG).

*before the expiration of the time limit for amending the
claims and to be republished in the event of receipt of
amendments (Rule 48.2(h))*

Published:

— *with international search report (Art. 21(3))*

DETECTION OF FRAUD OR ABUSE

BACKGROUND

1. Field of the Invention

[0001] Embodiments of the present invention(s) are directed to detection of fraud or abuse and more particularly to detecting fraud and abuse among multiple entities of a variety of services being represented as having been provided or utilized by each entity.

2. Related Art

[0002] According to the National Health Care Anti-Fraud Association (NHCAA), “the United States spends more than \$2 trillion on health care every year.” Of that amount, the NHCAA estimates more than \$60 billion each year is lost to fraud or abuse. Fraud is generally defined as knowingly and willfully executing, or attempting to execute, a scheme or artifice to defraud any health care benefit program or to obtain (by means of false or fraudulent pretenses representations, or promises) any of the money or property owned by, or under the custody or control of, any health care benefit program. (18 U.S.C. § 1347). An example of fraud includes a false claim for payment.

[0003] Abuse involves payment for items or services when there is no legal entitlement to that payment and the entity (e.g. health care provider or supplier) has not knowingly and/or intentionally misrepresented facts to obtain payment. Abuse cannot always be easily identified, because what is “abuse” versus “fraud” depends on specific facts and circumstances, intent, and prior knowledge, and available evidence, among other factors. Examples of abuse include:

- Unnecessary costs to the health care system (e.g., Medicare and Medicaid programs in the United States)
- Improper payment for services
- Billing for services or items not furnished
- Payment for services that fail to meet professionally recognized standards of care
- Services that are medically unnecessary
- Overcharging
- Misusing codes on a claim

[0004] Waste is overutilization of services or other practices that, directly or indirectly, result in unnecessary costs to the health care system, including Medicare and Medicaid programs. It is not generally considered to be caused by criminally negligent actions, but by the misuse of resources.

[0005] As the number of providers and consumers of services and resources grow and the number and kind of services and resources provided increase, there is an opportunity to collect a considerable amount of data regarding the providers and their actions (e.g., billing practices, type and number of services provided, charges, and the like). Data collected over time by a large number of such providers or consumers does not always directly indicate when the provider or consumer is involved in fraud or abuse.

[0006] As the collection and storage data has increased, there is an increased need to analyze and make sense of large amounts of data. Unfortunately, previous methods of analysis of large multidimensional datasets tend to be insufficient (if possible at all) to identify important relationships and may be computationally inefficient.

[0007] In one example, previous methods of analysis often use clustering. Clustering is often too blunt an instrument to identify important relationships in the data. Similarly, previous methods of linear regression, projection pursuit, principal component analysis, and multidimensional scaling often do not reveal important relationships. Existing linear algebraic and analytic methods are too sensitive to large scale distances and, as a result, lose detail.

[0008] Although some previous methods allow graphs depicting some relationships in the data, the graphs are not interactive and require considerable time for a team of such experts to understand the relationships. Further, the output of previous methods does not allow for exploratory data analysis where the analysis can be quickly modified to discover new relationships.

SUMMARY OF THE INVENTION(S)

[0009] An example non-transitory computer readable medium including executable instructions, the instructions being executable by a processor to perform a method. The method may comprise receiving a first set of data identifying entities and performance information for analysis, receiving a second set of data identifying entities and performance information associated with known or suspected past fraud or abuse, receiving metric and lens selections, performing metric and lens functions, based on the metric and lens selections, on

first and second set of data to map performance information from the first and second sets to a reference space, generating cover of reference space and cluster mapped performance information to identify nodes in a graph, each node including one or more entities as members, each node being connected to another node if they share at least one common entity as members, identifying nodes that include at least one member from the second set of data, determining entities that are members of the identified nodes that are from the first set of data, and generating a first report listing the determined entities that are members of the identified nodes that are from the first set of data as possibly involved in fraud or waste.

[0010] In some embodiments, the method further comprises determining other entities that are member of other nodes that are connected to the identified nodes by an edge, the other entities being from the first set of data, and adding one or more of the other entities to the first report. The entities of the first set of data may be providers of health services and the performance information is health care information including charges for treatment and treatments provided to patients. In some embodiments, the entities of the first set of data are consumers of services and the performance information is access information and resource utilization of networks and network resources.

[0011] The method may further comprise applying one or more functions on at least some performance data of the first set of data and including those determined entities that are members of the identified nodes that are from the first set of data in the first report based, in part, on at least one value calculated as a result of the one or more functions. The one or more functions may be or include an L1 function or L infinity function.

[0012] In various embodiments, the method further comprises receiving a new entity with new performance information associated with that new entity, determine distances between new performance information of new entity and performance information of entities of first and second sets of data, comparing distances between new performance information for new entity and the distances between entities of each node, determine location of new entity in the TDA graph based on the comparison, and generate a second report if the new entity is determined to be in a node that has at least one member from the second set of data.

[0013] In some embodiments, the method further comprises receiving a new entity with new performance information associated with that new entity, determining distances between new performance information of new entity and performance information of entities of first and second sets of data, comparing distances between new performance information for new

entity and the distances between entities of each node, determining location of new entity in the TDA graph based on the comparison, and generating a second report if the new entity is determined to be in a node that is linked by an edge to a node that has at least one member from the second set of data.

[0014] The method may further comprise removing performance information from the first set of data if the performance information is older than a predetermined date leaving remaining performance information, performing the metric and lens functions on first and second set of data to map remaining performance information from the first set of data and the performance information from the second set of data to the reference space, and generating cover of the reference space and cluster mapped performance information to identify nodes in the graph.

[0015] As discussed herein, another method may comprise receiving a first set of data identifying entities and performance information for analysis, receiving a second set of data identifying entities and performance information associated with known or suspected past fraud or abuse, receiving metric and lens selections, performing metric and lens functions, based on the metric and lens selections, on first and second set of data to map performance information from the first and second sets to a reference space, generating cover of reference space and cluster mapped performance information to identify nodes in a graph, each node including one or more entities as members, each node being connected to another node if they share at least one common entity as members, identifying nodes that include at least one member from the second set of data, determining entities that are members of the identified nodes that are from the first set of data, and generating a first report listing the determined entities that are members of the identified nodes that are from the first set of data as possibly involved in fraud or waste.

[0016] An example system may comprise a processor and a memory. The memory may include instructions to configure the processor to receive a first set of data identifying entities and performance information for analysis, receive a second set of data identifying entities and performance information associated with known or suspected past fraud or abuse, receive metric and lens selections, perform metric and lens functions, based on the metric and lens selections, on first and second set of data to map performance information from the first and second sets to a reference space, generate cover of reference space and cluster mapped performance information to identify nodes in a graph, each node including one or more entities

as members, each node being connected to another node if they share at least one common entity as members, identify nodes that include at least one member from the second set of data, determine entities that are members of the identified nodes that are from the first set of data, and generate a first report listing the determined entities that are members of the identified nodes that are from the first set of data as possibly involved in fraud or waste.

BRIEF DESCRIPTION OF THE DRAWINGS

[0017] FIG. 1A is an example graph representing data that appears to be divided into three disconnected groups.

[0018] FIG. 1B is an example graph representing data set obtained from a Lotka-Volterra equation modeling the populations of predators and prey over time.

[0019] FIG. 1C is an example graph of data sets whereby the data does not break up into disconnected groups, but instead has a structure in which there are lines (or flares) emanating from a central group.

[0020] FIG. 2 is an exemplary environment in which embodiments may be practiced.

[0021] FIG. 3 is a block diagram of an exemplary analysis server.

[0022] FIG. 4 is a flow chart depicting an exemplary method of dataset analysis and visualization in some embodiments.

[0023] FIG. 5 is an exemplary ID field selection interface window in some embodiments.

[0024] FIG. 6A is an exemplary data field selection interface window in some embodiments.

[0025] FIG. 6B is an exemplary metric and filter selection interface window in some embodiments.

[0026] FIG. 7 is an exemplary filter parameter interface window in some embodiments.

[0027] FIG. 8 is a flowchart for data analysis and generating a visualization in some embodiments.

[0028] FIG. 9 is an exemplary interactive visualization in some embodiments.

[0029] FIG. 10 is an exemplary interactive visualization displaying an explain information window in some embodiments.

[0030] FIG. 11 is a flowchart of functionality of the interactive visualization in some embodiments.

[0031] FIG. 12 is a flowchart of for generating a cancer map visualization utilizing biological data of a plurality of patients in some embodiments.

[0032] FIG. 13 is an exemplary data structure including biological data for a number of patients that may be used to generate the cancer map visualization in some embodiments.

[0033] FIG. 14 is an exemplary visualization displaying the cancer map in some embodiments.

[0034] FIG. 15 is a flowchart of for positioning new patient data relative to the cancer map visualization in some embodiments.

[0035] FIG. 16 is an exemplary visualization displaying the cancer map including positions for three new cancer patients in some embodiments.

[0036] FIG. 17 is a flowchart of utilization the visualization and positioning of new patient data in some embodiments.

[0037] FIG. 18 is an exemplary digital device in some embodiments.

[0038] FIG. 19 is a block diagram of an exemplary analysis server including a normalization module, a fraud or abuse grouping module, a reporting module, and a ranking module in some embodiments.

[0039] FIG. 20 is a flowchart for identifying health care providers to investigate for possible fraud or abuse in some embodiments.

[0040] FIG. 21 is an example TDA visualization that may be generated from the steps of FIG. 20 in some embodiments.

[0041] FIG. 22 is an example TDA visualization that is a portion of the TDA visualization of FIG. 21.

[0042] FIG. 23 is a flowchart of for positioning new health care information of a particular provider in a TDA graph which may be visualized or not visualized in some embodiments.

[0043] FIG. 24 is a flowchart of for positioning new health care information of a new provider in a TDA graph which may be visualized or not visualized in some embodiments.

[0044] FIG. 25 is a flowchart for identifying health care providers to investigate for possible fraud or abuse in various embodiments.

[0045] FIG. 26 is a sample report generated by the ranking module.

[0046] FIG. 27 depicts a visualization of a graph that illustrates outcomes that are not significantly localized.

[0047] FIG. 28 depicts a visualization of a graph that illustrates outcomes that are more localized than FIG. 27.

[0048] FIG. 29 is a block diagram of an exemplary analysis server including an outcome analysis module.

[0049] FIG. 30 depicts an example outcome analysis module in some embodiments.

[0050] FIG. 31 is a flowchart for outcome auto analysis in some embodiments.

[0051] FIG. 32 is a flowchart for selection of a subset of metrics in some embodiments.

[0052] FIG. 33A depicts groupings of data points with fairly consistent outcomes.

[0053] FIG. 33B depicts an example graph using a Manhattan metric. Like FIG. 33A, FIG. 33B depicts groupings of data points with fairly consistent outcomes.

[0054] FIG. 33C depicts an example graph using an Absolute Correlation metric. FIG. 33C depicts one large group with different outcomes that are intermixed.

[0055] FIG. 34A depicts groupings of data points with fairly consistent outcomes although there are some relatively minor intermixed data.

[0056] FIG. 34B depicts an example graph using a Euclidean metric.

[0057] FIG. 34C depicts an example graph using a Norm. Correlation metric.

[0058] FIG. 34D depicts an example graph using an Chebyshev metric.

[0059] FIG. 35 is a flowchart of selection of a subset of metric-lens combinations in some embodiments.

[0060] FIG. 36A depicts an example graph using a Euclidean (L2) metric with a neighborhood lens.

[0061] FIG. 36B depicts an example graph using an angle metric and MDS Coord. 2 lens.

[0062] FIG. 36C depicts an example graph using a Euclidean (L1) using a neighborhood lens.

[0063] FIG. 37A depicts groupings of data points with some mixed outcomes.

[0064] FIG. 37B depicts an example graph using a Euclidean metric and neighborhood lens.

[0065] FIG. 37C depicts an example graph using a Cosine using a PCA lens.

[0066] FIG. 38 is a flowchart for identifying one or more graphs based on a graph score using the metric-lens combinations of the subset of metric-lens combinations.

[0067] FIG. 39A depicts a visualization of a graph using a Chebyshev (L-Infinity) metric and a neighborhood lens 2 (resolution 61, gain of 4.0).

[0068] FIG. 39B depicts a visualization of a graph using a variance normalized Euclidean metric and a neighborhood lens 1(resolution 57, gain of 3.0).

DETAILED DESCRIPTION OF THE DRAWINGS

[0069] Systems and methods described herein enable detection of fraud and/or abuse in systems that report a variety of different services ostensibly performed, received, or consumed by a variety of different entities. For example, in healthcare, multiple providers (e.g., laboratories, individual doctors, groups of doctors, and clinics) provide a variety of health care services (e.g., treatments, tests, procedures, provisioning of medicines, health equipment, personal care, long term care, residence monitoring, or the like) for a large number of different patients. Many providers will provide bills for compensation for the different services to private or public health care insurers (e.g., Blue Cross, Medicaid, and Medicare).

[0070] Each health care insurer may receive an incredible number of requests for reimbursements, invoices, billing reports, services reports, or the like from a large number of providers over time. Due to the number of charges, services provided, equipment used, pharmaceuticals prescribed, and patients, providers may intentionally or unintentionally seek to obtain payment beyond what they are due (e.g., as a result of fraud or abuse). Detecting fraud or abuse can be very difficult given the volume and structure of information.

[0071] Various embodiments described herein utilize Topological Data Analysis (TDA) to assist in identifying entities that are potentially demanding or requesting payment, services, or equipment where there is no legal entitlement due to fraud, abuse, and/or waste. The shape of data revealed by a graph generated by TDA may provide queryless insights of relationships between cases of known fraud and/or abuse with other entities and their practices. “Queryless” may refer to the initial generation of a TDA graph or visualization that depicts shape of data after TDA analysis without first receiving a specific query for the TDA analysis.

[0072] Utilizing TDA, entities (e.g. providers) and/or information of entities can be grouped with cases of known fraud and/or abuse. Alternately, the shape of data revealed by TDA may provide insights indicating outliers or information that is at the extremes when compared to other entities and/or other provided information depicted in the graph. Entities and/or information of entities that are at the extremes or are outliers may be identified as possible cases of fraud and/or abuse.

[0073] In the health care example, health care information data indicating services, charges, visits, equipment utilization, facility utilization, treatments, charges or the like may be collected from a variety of providers. Another dataset may indicate cases of known fraud and/or abuse (e.g., as confirmed or determined by an investigator or analyst). The health care

information and the subset of the data associated with fraud and/or abuse may be analyzed using TDA (further discussed herein). A graph (either visualized or not visualized) may indicate similarity of behaviors, charges, or results of some providers with other providers associated with fraud and/or abuse. Systems and methods described herein, may provide or enable providing a list of providers by identifying those providers and/or information associated with the providers (e.g., features of the data including, for example, billing and services) that can be grouped (e.g., in the same node or proximate to the same node in the graph generated by TDA) with information associated with fraud or abuse.

[0074] Some embodiments described herein may be a part of the subject of Topological Data Analysis (TDA). TDA is an area of research which has produced methods for studying point cloud data sets from a geometric point of view. Other data analysis techniques use “approximation by models” of various types. For example, regression methods model the data as the graph of a function in one or more variables. Unfortunately, certain qualitative properties (which one can readily observe when the data is two-dimensional) may be of a great deal of importance for understanding, and these features may not be readily represented within such models.

[0075] FIG. 1A is an example graph representing data that appears to be divided into three disconnected groups. In this example, the data for this graph may be associated with various physical characteristics related to different population groups or biomedical data related to different forms of a disease. Seeing that the data breaks into groups in this fashion can give insight into the data, once one understands what characterizes the groups.

[0076] FIG. 1B is an example graph representing data set obtained from a Lotka-Volterra equation modeling the populations of predators and prey over time. From FIG. 1B, one observation about this data is that it is arranged in a loop. The loop is not exactly circular, but it is topologically a circle. The exact form of the equations, while interesting, may not be of as much importance as this qualitative observation which reflects the fact that the underlying phenomenon is recurrent or periodic. When looking for periodic or recurrent phenomena, methods may be developed which can detect the presence of loops without defining explicit models. For example, periodicity may be detectable without having to first develop a fully accurate model of the dynamics.

[0077] FIG. 1C is an example graph of data sets whereby the data does not break up into disconnected groups, but instead has a structure in which there are lines (or flares) emanating

from a central group. In this case, the data also suggests the presence of three distinct groups, but the connectedness of the data does not reflect this. This particular data that is the basis for the example graph in FIG. 1C arises from a study of single nucleotide polymorphisms (SNPs).

[0078] In each of the examples above, aspects of the shape of the data are relevant in reflecting information about the data. Connectedness (the simplest property of shape) reflects the presence of a discrete classification of the data into disparate groups. The presence of loops, another simple aspect of shape, often reflect periodic or recurrent behavior. Finally, in the third example, the shape containing flares suggests a classification of the data descriptive of ways in which phenomena can deviate from the norm, which would typically be represented by the central core. These examples support the idea that the shape of data (suitably defined) is an important aspect of its structure, and that it is therefore important to develop methods for analyzing and understanding its shape. The part of mathematics which concerns itself with the study of shape is called topology, and topological data analysis attempts to adapt methods for studying shape which have been developed in pure mathematics to the study of the shape of data, suitably defined.

[0079] One question is how notions of geometry or shape are translated into information about point clouds, which are, after all, finite sets? What we mean by shape or geometry can come from a dissimilarity function or metric (e.g., a non-negative, symmetric, real-valued function d on the set of pairs of points in the data set which may also satisfy the triangle inequality, and $d(x; y) = 0$ if and only if $x = y$). Such functions exist in profusion for many data sets. For example, when the data comes in the form of a numerical matrix, where the rows correspond to the data points and the columns are the fields describing the data, the n -dimensional Euclidean distance function is natural when there are n fields. Similarly, in this example, there are Pearson correlation distances, cosine distances, and other choices.

[0080] When the data is not Euclidean, for example if one is considering genomic sequences, various notions of distance may be defined using measures of similarity based on Basic Local Alignment Search Tool (BLAST) type similarity scores. Further, a measure of similarity can come in non-numeric forms, such as social networks of friends or similarities of hobbies, buying patterns, tweeting, and/or professional interests. In any of these ways the notion of shape may be formulated via the establishment of a useful notion of similarity of data points.

[0081] One of the advantages of TDA is that it may depend on nothing more than such a notion, which is a very primitive or low-level model. It may rely on many fewer assumptions than standard linear or algebraic models, for example. Further, the methodology may provide new ways of visualizing and compressing data sets, which facilitate understanding and monitoring data. The methodology may enable study of interrelationships among disparate data sets and/or multiscale/multiresolution study of data sets. Moreover, the methodology may enable interactivity in the analysis of data, using point and click methods.

[0082] TDA may be a very useful complement to more traditional methods, such as Principal Component Analysis (PCA), multidimensional scaling, and hierarchical clustering. These existing methods are often quite useful, but suffer from significant limitations. PCA, for example, is an essentially linear procedure and there are therefore limits to its utility in highly non-linear situations. Multidimensional scaling is a method which is not intrinsically linear, but can in many situations wash out detail, since it may overweight large distances. In addition, when metrics do not satisfy an intrinsic flatness condition, it may have difficulty in faithfully representing the data. Hierarchical clustering does exhibit multiscale behavior, but represents data only as disjoint clusters, rather than retaining any of the geometry of the data set. In all four cases, these limitations matter for many varied kinds of data.

[0083] We now summarize example properties of an example construction, in some embodiments, which may be used for representing the shape of data sets in a useful, understandable fashion as a finite graph:

- The input may be a collection of data points equipped in some way with a distance or dissimilarity function, or other description. This can be given implicitly when the data is in the form of a matrix, or explicitly as a matrix of distances or even the generating edges of a mathematical network.
- One construction may also use one or more lens functions (i.e. real valued functions on the data). Lens function(s) may depend directly on the metric. For example, lens function(s) might be the result of a density estimator or a measure of centrality or data depth. Lens function(s) may, in some embodiments, depend on a particular representation of the data, as when one uses the first one or two coordinates of a principal component or multidimensional scaling analysis. In some embodiments, the lens function(s) may be columns which expert knowledge identifies as being intrinsically interesting, as in cholesterol levels and BMI in a study of heart disease.

- In some embodiments, the construction may depend on a choice of two or more processing parameters, resolution, and gain. Increase in resolution typically results in more nodes and an increase in the gain increases the number of edges in a visualization and/or graph in a reference space as further described herein.
- The output may be, for example, a visualization (e.g., a display of connected nodes or “network”) or simplicial complex. One specific combinatorial formulation in one embodiment may be that the vertices form a finite set, and then the additional structure may be a collection of edges (unordered pairs of vertices) which are pictured as connections in this network.

[0084] In various embodiments, a system for handling, analyzing, and visualizing data using drag and drop methods as opposed to text based methods is described herein.

Philosophically, data analytic tools are not necessarily regarded as “solvers,” but rather as tools for interacting with data. For example, data analysis may consist of several iterations of a process in which computational tools point to regions of interest in a data set. The data set may then be examined by people with domain expertise concerning the data, and the data set may then be subjected to further computational analysis. In some embodiments, methods described herein provide for going back and forth between mathematical constructs, including interactive visualizations (e.g., graphs), on the one hand and data on the other.

[0085] In one example of data analysis in some embodiments described herein, an exemplary clustering tool is discussed which may be more powerful than existing technology, in that one can find structure within clusters and study how clusters change over a period of time or over a change of scale or resolution.

[0086] An exemplary interactive visualization tool (e.g., a visualization module which is further described herein) may produce combinatorial output in the form of a graph which can be readily visualized. In some embodiments, the exemplary interactive visualization tool may be less sensitive to changes in notions of distance than current methods, such as multidimensional scaling.

[0087] Some embodiments described herein permit manipulation of the data from a visualization. For example, portions of the data which are deemed to be interesting from the visualization can be selected and converted into database objects, which can then be further analyzed. Some embodiments described herein permit the location of data points of interest

within the visualization, so that the connection between a given visualization and the information the visualization represents may be readily understood.

[0088] FIG. 2 is an exemplary environment 200 in which embodiments may be practiced. In various embodiments, data analysis and interactive visualization may be performed locally (e.g., with software and/or hardware on a local digital device), across a network (e.g., via cloud computing), or a combination of both. In many of these embodiments, a data structure is accessed to obtain the data for the analysis, the analysis is performed based on properties and parameters selected by a user, and an interactive visualization is generated and displayed. There are many advantages between performing all or some activities locally and many advantages of performing all or some activities over a network.

[0089] Environment 200 comprises user devices 202a-202n, a communication network 204, data storage server 206, and analysis server 208. Environment 200 depicts an embodiment wherein functions are performed across a network. In this example, the user(s) may take advantage of cloud computing by storing data in a data storage server 206 over a communication network 204. The analysis server 208 may perform analysis and generation of an interactive visualization.

[0090] User devices 202a-202n may be any digital devices. A digital device is any device that comprises memory and a processor. Digital devices are further described in FIG. 2. The user devices 202a-202n may be any kind of digital device that may be used to access, analyze and/or view data including, but not limited to a desktop computer, laptop, notebook, or other computing device.

[0091] In various embodiments, a user, such as a data analyst, may generate a database or other data structure with the user device 202a to be saved to the data storage server 206. The user device 202a may communicate with the analysis server 208 via the communication network 204 to perform analysis, examination, and visualization of data within the database.

[0092] The user device 202a may comprise a client program for interacting with one or more applications on the analysis server 208. In other embodiments, the user device 202a may communicate with the analysis server 208 using a browser or other standard program. In various embodiments, the user device 202a communicates with the analysis server 208 via a virtual private network. It will be appreciated that that communication between the user device 202a, the data storage server 206, and/or the analysis server 208 may be encrypted or otherwise secured.

[0093] The communication network 204 may be any network that allows digital devices to communicate. The communication network 204 may be the Internet and/or include LAN and WANs. The communication network 204 may support wireless and/or wired communication.

[0094] The data storage server 206 is a digital device that is configured to store data. In various embodiments, the data storage server 206 stores databases and/or other data structures. The data storage server 206 may be a single server or a combination of servers. In one example the data storage server 206 may be a secure server wherein a user may store data over a secured connection (e.g., via https). The data may be encrypted and backed-up. In some embodiments, the data storage server 206 is operated by a third-party such as Amazon's S3 service.

[0095] The database or other data structure may comprise large high-dimensional datasets. These datasets are traditionally very difficult to analyze and, as a result, relationships within the data may not be identifiable using previous methods. Further, previous methods may be computationally inefficient.

[0096] The analysis server 208 is a digital device that may be configured to analyze data. In various embodiments, the analysis server may perform many functions to interpret, examine, analyze, and display data and/or relationships within data. In some embodiments, the analysis server 208 performs, at least in part, topological analysis of large datasets applying metrics, filters, and resolution parameters chosen by the user. The analysis is further discussed in FIG. 8 herein.

[0097] The analysis server 208 may generate an interactive visualization of the output of the analysis. The interactive visualization allows the user to observe and explore relationships in the data. In various embodiments, the interactive visualization allows the user to select nodes comprising data that has been clustered. The user may then access the underlying data, perform further analysis (e.g., statistical analysis) on the underlying data, and manually reorient the graph(s) (e.g., structures of nodes and edges described herein) within the interactive visualization. The analysis server 208 may also allow for the user to interact with the data, see the graphic result. The interactive visualization is further discussed in FIGS. 9-11.

[0098] In some embodiments, the analysis server 208 interacts with the user device(s) 202a-202n over a private and/or secure communication network. The user device 202a may comprise a client program that allows the user to interact with the data storage server 206, the

analysis server 208, another user device (e.g., user device 202n), a database, and/or an analysis application executed on the analysis server 208.

[0099] Those skilled in the art will appreciate that all or part of the data analysis may occur at the user device 202a. Further, all or part of the interaction with the visualization (e.g., graphic) may be performed on the user device 202a.

[0100] Although two user devices 202a and 202n are depicted, those skilled in the art will appreciate that there may be any number of user devices in any location (e.g., remote from each other). Similarly, there may be any number of communication networks, data storage servers, and analysis servers.

[0101] Cloud computing may allow for greater access to large datasets (e.g., via a commercial storage service) over a faster connection. Further, it will be appreciated that services and computing resources offered to the user(s) may be scalable.

[0102] FIG. 3 is a block diagram of an exemplary analysis server 208. In exemplary embodiments, the analysis server 208 comprises a processor 302, input/output (I/O) interface 304, a communication network interface 306, a memory system 308, a storage system 310, and a processing module 312. The processor 302 may comprise any processor or combination of processors with one or more cores.

[0103] The input / output (I/O) interface 304 may comprise interfaces for various I/O devices such as, for example, a keyboard, mouse, and display device. The exemplary communication network interface 306 is configured to allow the analysis server 208 to communicate with the communication network 204 (see FIG. 2). The communication network interface 306 may support communication over an Ethernet connection, a serial connection, a parallel connection, and/or an ATA connection. The communication network interface 306 may also support wireless communication (e.g., 802.11 a/b/g/n, WiMax, LTE, WiFi). It will be apparent to those skilled in the art that the communication network interface 306 can support many wired and wireless standards.

[0104] The memory system 308 may be any kind of memory including RAM, ROM, or flash, cache, virtual memory, etc. In various embodiments, working data is stored within the memory system 308. The data within the memory system 308 may be cleared or ultimately transferred to the storage system 310.

[0105] The storage system 310 includes any storage configured to retrieve and store data. Some examples of the storage system 310 include flash drives, hard drives, optical drives, and/or magnetic tape. Each of the memory system 308 and the storage system 310 comprises a computer-readable medium, which stores instructions (e.g., software programs) executable by processor 302.

[0106] The storage system 310 comprises a plurality of modules utilized by embodiments of discussed herein. A module may be hardware, software (e.g., including instructions executable by a processor), or a combination of both. In one embodiment, the storage system 310 comprises a processing module 312 which comprises an input module 314, a filter module 316, a resolution module 318, an analysis module 320, a visualization engine 322, and database storage 324. Alternative embodiments of the analysis server 208 and/or the storage system 310 may comprise more, less, or functionally equivalent components and modules.

[0107] The input module 314 may be configured to receive commands and preferences from the user device 202a. In various examples, the input module 314 receives selections from the user which will be used to perform the analysis. The output of the analysis may be an interactive visualization.

[0108] The input module 314 may provide the user a variety of interface windows allowing the user to select and access a database, choose fields associated with the database, choose a metric, choose one or more filters, and identify resolution parameters for the analysis. In one example, the input module 314 receives a database identifier and accesses a large multi-dimensional database. The input module 314 may scan the database and provide the user with an interface window allowing the user to identify an ID field. An ID field is an identifier for each data point. In one example, the identifier is unique. The same column name may be present in the table from which filters are selected. After the ID field is selected, the input module 314 may then provide the user with another interface window to allow the user to choose one or more data fields from a table of the database.

[0109] Although interactive windows may be described herein, it will be appreciated that any window, graphical user interface, and/or command line may be used to receive or prompt a user or user device 202a for information.

[0110] The filter module 316 may subsequently provide the user with an interface window to allow the user to select a metric to be used in analysis of the data within the chosen data fields. The filter module 316 may also allow the user to select and/or define one or more filters.

[0111] The resolution module 218 may allow the user to select a resolution, including filter parameters. In one example, the user enters a number of intervals and a percentage overlap for a filter.

[0112] The analysis module 320 may perform data analysis based on the database and the information provided by the user. In various embodiments, the analysis module 320 performs an algebraic topological analysis to identify structures and relationships within data and clusters of data. It will be appreciated that the analysis module 320 may use parallel algorithms or use generalizations of various statistical techniques (e.g., generalizing the bootstrap to zig-zag methods) to increase the size of data sets that can be processed. The analysis is further discussed in FIG. 8. It will be appreciated that the analysis module 320 is not limited to algebraic topological analysis but may perform any analysis.

[0113] The visualization engine 322 generates an interactive visualization including the output from the analysis module 320. The interactive visualization allows the user to see all or part of the analysis graphically. The interactive visualization also allows the user to interact with the visualization. For example, the user may select portions of a graph from within the visualization to see and/or interact with the underlying data and/or underlying analysis. The user may then change the parameters of the analysis (e.g., change the metric, filter(s), or resolution(s)) which allows the user to visually identify relationships in the data that may be otherwise undetectable using prior means. The interactive visualization is further described in FIGS. 9-11.

[0114] The database storage 324 is configured to store all or part of the database that is being accessed. In some embodiments, the database storage 324 may store saved portions of the database. Further, the database storage 324 may be used to store user preferences, parameters, and analysis output thereby allowing the user to perform many different functions on the database without losing previous work.

[0115] It will be appreciated that that all or part of the processing module 312 may be at the user device 202a or the database storage server 206. In some embodiments, all or some of the functionality of the processing module 312 may be performed by the user device 202a.

[0116] In various embodiments, systems and methods discussed herein may be implemented with one or more digital devices. In some examples, some embodiments discussed herein may be implemented by a computer program (instructions) executed by a processor. The computer program may provide a graphical user interface. Although such a

computer program is discussed, it will be appreciated that embodiments may be performed using any of the following, either alone or in combination, including, but not limited to, a computer program, multiple computer programs, firmware, and/or hardware.

[0117] A module and/or engine may include any processor or combination of processors. In some examples, a module and/or engine may include or be a part of a processor, digital signal processor (DSP), application specific integrated circuit (ASIC), an integrated circuit, and/or the like. In various embodiments, the module and/or engine may be software or firmware.

[0118] FIG. 4 is a flow chart 400 depicting an exemplary method of dataset analysis and visualization in some embodiments. In step 402, the input module 314 accesses a database. The database may be any data structure containing data (e.g., a very large dataset of multidimensional data). In some embodiments, the database may be a relational database. In some examples, the relational database may be used with MySQL, Oracle, Microsoft SQL Server, Aster nCluster, Teradata, and/or Vertica. It will be appreciated that the database may not be a relational database.

[0119] In some embodiments, the input module 314 receives a database identifier and a location of the database (e.g., the data storage server 206) from the user device 202a (see FIG. 2). The input module 314 may then access the identified database. In various embodiments, the input module 314 may read data from many different sources, including, but not limited to MS Excel files, text files (e.g., delimited or CSV), Matlab .mat format, or any other file.

[0120] In some embodiments, the input module 314 receives an IP address or hostname of a server hosting the database, a username, password, and the database identifier. This information (herein referred to as “connection information”) may be cached for later use. It will be appreciated that the database may be locally accessed and that all, some, or none of the connection information may be required. In one example, the user device 202a may have full access to the database stored locally on the user device 202a so the IP address is unnecessary. In another example, the user device 202a may already have loaded the database and the input module 314 merely begins by accessing the loaded database.

[0121] In various embodiments, the identified database stores data within tables. A table may have a “column specification” which stores the names of the columns and their data types. A “row” in a table, may be a tuple with one entry for each column of the correct type. In one example, a table to store employee records might have a column specification such as:

- employee_id primary key int (this may store the employee's ID as an integer, and uniquely identifies a row)
- age int
- gender char(1) (gender of the employee may be a single character either M or F)
- salary double (salary of an employee may be a floating point number)
- name varchar (name of the employee may be a variable-length string)

In this example, each employee corresponds to a row in this table. Further, the tables in this exemplary relational database are organized into logical units called databases. An analogy to file systems is that databases can be thought of as folders and files as tables. Access to databases may be controlled by the database administrator by assigning a username/password pair to authenticate users.

[0122] Once the database is accessed, the input module 314 may allow the user to access a previously stored analysis or to begin a new analysis. If the user begins a new analysis, the input module 314 may provide the user device 202a with an interface window allowing the user to identify a table from within the database. In one example, the input module 314 provides a list of available tables from the identified database.

[0123] In step 404, the input module 314 receives a table identifier identifying a table from within the database. The input module 314 may then provide the user with a list of available ID fields from the table identifier. In step 406, the input module 314 receives the ID field identifier from the user and/or user device 202a. The ID field is, in some embodiments, the primary key.

[0124] Having selected the primary key, the input module 314 may generate a new interface window to allow the user to select data fields for analysis. In step 408, the input module 314 receives data field identifiers from the user device 202a. The data within the data fields may be later analyzed by the analysis module 320.

[0125] In step 410, the filter module 316 identifies a metric. In some embodiments, the filter module 316 and/or the input module 314 generates an interface window allowing the user of the user device 202a options for a variety of different metrics and filter preferences. The interface window may be a drop down menu identifying a variety of distance metrics to be used in the analysis. Metric options may include, but are not limited to, Euclidean, DB Metric, variance normalized Euclidean, and total normalized Euclidean. The metric and the analysis are further described herein.

[0126] In step 412, the filter module 316 selects one or more filters. In some embodiments, the user selects and provides filter identifier(s) to the filter module 316. The role of the filters in the analysis is also further described herein. The filters, for example, may be user defined, geometric, or based on data which has been pre-processed. In some embodiments, the data based filters are numerical arrays which can assign a set of real numbers to each row in the table or each point in the data generally.

[0127] A variety of geometric filters may be available for the user to choose. Geometric filters may include, but are not limited to:

- Density
- L1 Eccentricity
- L-infinity Eccentricity
- Witness based Density
- Witness based Eccentricity
- Eccentricity as distance from a fixed point
- Approximate Kurtosis of the Eccentricity

[0128] In step 414, the resolution module 218 defines the resolution to be used with a filter in the analysis. The resolution may comprise a number of intervals and an overlap parameter. In various embodiments, the resolution module 218 allows the user to adjust the number of intervals and overlap parameter (e.g., percentage overlap) for one or more filters.

[0129] In step 416, the analysis module 320 processes data of selected fields based on the metric, filter(s), and resolution(s) to generate the visualization. This process is discussed in FIG. 8.

[0130] In step 418, the visualization module 322 displays the interactive visualization. In various embodiments, the visualization may be rendered in two or three dimensional space. The visualization module 322 may use an optimization algorithm for an objective function which is correlated with good visualization (e.g., the energy of the embedding). The visualization may show a collection of nodes corresponding to each of the partial clusters in the analysis output and edges connecting them as specified by the output. The interactive visualization is further discussed in FIGS. 9-11.

[0131] Although many examples discuss the input module 314 as providing interface windows, it will be appreciated that all or some of the interface may be provided by a client on

the user device 202a. Further, in some embodiments, the user device 202a may be running all or some of the processing module 212.

[0132] FIGS. 5-7 depict various interface windows to allow the user to make selections, enter information (e.g., fields, metrics, and filters), provide parameters (e.g., resolution), and provide data (e.g., identify the database) to be used with analysis. It will be appreciated that any graphical user interface or command line may be used to make selections, enter information, provide parameters, and provide data.

[0133] FIG. 5 is an exemplary ID field selection interface window 500 in some embodiments. The ID field selection interface window 500 allows the user to identify an ID field. The ID field selection interface window 500 comprises a table search field 502, a table list 504, and a fields selection window 506.

[0134] In various embodiments, the input module 314 identifies and accesses a database from the database storage 324, user device 202a, or the data storage server 206. The input module 314 may then generate the ID field selection interface window 500 and provide a list of available tables of the selected database in the table list 504. The user may click on a table or search for a table by entering a search query (e.g., a keyword) in the table search field 502. Once a table is identified (e.g., clicked on by the user), the fields selection window 506 may provide a list of available fields in the selected table. The user may then choose a field from the fields selection window 506 to be the ID field. In some embodiments, any number of fields may be chosen to be the ID field(s).

[0135] FIG. 6A is an exemplary data field selection interface window 600a in some embodiments. The data field selection interface window 600a allows the user to identify data fields. The data field selection interface window 600a comprises a table search field 502, a table list 504, a fields selection window 602, and a selected window 604.

[0136] In various embodiments, after selection of the ID field, the input module 314 provides a list of available tables of the selected database in the table list 504. The user may click on a table or search for a table by entering a search query (e.g., a keyword) in the table search field 502. Once a table is identified (e.g., clicked on by the user), the fields selection window 506 may provide a list of available fields in the selected table. The user may then choose any number of fields from the fields selection window 602 to be data fields. The selected data fields may appear in the selected window 604. The user may also deselect fields that appear in the selected window 604.

[0137] It will be appreciated that the table selected by the user in the table list 504 may be the same table selected with regard to FIG. 5. In some embodiments, however, the user may select a different table. Further, the user may, in various embodiments, select fields from a variety of different tables.

[0138] FIG. 6B is an exemplary metric and filter selection interface window 600b in some embodiments. The metric and filter selection interface window 600b allows the user to identify a metric, add filter(s), and adjust filter parameters. The metric and filter selection interface window 600b comprises a metric pull down menu 606, an add filter from database button 608, and an add geometric filter button 610.

[0139] In various embodiments, the user may click on the metric pull down menu 606 to view a variety of metric options. Various metric options are described herein. In some embodiments, the user may define a metric. The user defined metric may then be used with the analysis.

[0140] In one example, finite metric space data may be constructed from a data repository (i.e., database, spreadsheet, or Matlab file). This may mean selecting a collection of fields whose entries will specify the metric using the standard Euclidean metric for these fields, when they are floating point or integer variables. Other notions of distance, such as graph distance between collections of points, may be supported.

[0141] The analysis module 320 may perform analysis using the metric as a part of a distance function. The distance function can be expressed by a formula, a distance matrix, or other routine which computes it. The user may add a filter from a database by clicking on the add filter from database button 608. The metric space may arise from a relational database, a Matlab file, an Excel spreadsheet, or other methods for storing and manipulating data. The metric and filter selection interface window 600b may allow the user to browse for other filters to use in the analysis. The analysis and metric function are further described in FIG. 8.

[0142] The user may also add a geometric filter 610 by clicking on the add geometric filter button 610. In various embodiments, the metric and filter selection interface window 600b may provide a list of geometric filters from which the user may choose.

[0143] FIG. 7 is an exemplary filter parameter interface window 700 in some embodiments. The filter parameter interface window 700 allows the user to determine a resolution for one or more selected filters (e.g., filters selected in the metric and filter selection

interface window 600). The filter parameter interface window 700 comprises a filter name menu 702, an interval field 704, an overlap bar 706, and a done button 708.

[0144] The filter parameter interface window 700 allows the user to select a filter from the filter name menu 702. In some embodiments, the filter name menu 702 is a drop down box indicating all filters selected by the user in the metric and filter selection interface window 600. Once a filter is chosen, the name of the filter may appear in the filter name menu 702. The user may then change the intervals and overlap for one, some, or all selected filters.

[0145] The interval field 704 allows the user to define a number of intervals for the filter identified in the filter name menu 702. The user may enter a number of intervals or scroll up or down to get to a desired number of intervals. Any number of intervals may be selected by the user. The function of the intervals is further discussed in FIG. 8.

[0146] The overlap bar 706 allows the user to define the degree of overlap of the intervals for the filter identified in the filter name menu 702. In one example, the overlap bar 706 includes a slider that allows the user to define the percentage overlap for the interval to be used with the identified filter. Any percentage overlap may be set by the user.

[0147] Once the intervals and overlap are defined for the desired filters, the user may click the done button. The user may then go back to the metric and filter selection interface window 600 and see a new option to run the analysis. In some embodiments, the option to run the analysis may be available in the filter parameter interface window 700. Once the analysis is complete, the result may appear in an interactive visualization which is further described in FIGS. 9-11.

[0148] It will be appreciated that that interface windows in FIGS. 4-7 are exemplary. The exemplary interface windows are not limited to the functional objects (e.g., buttons, pull down menus, scroll fields, and search fields) shown. Any number of different functional objects may be used. Further, as described herein, any other interface, command line, or graphical user interface may be used.

[0149] FIG. 8 is a flowchart 800 for data analysis and generating an interactive visualization in some embodiments. In various embodiments, the processing on data and user-specified options is motivated by techniques from topology and, in some embodiments, algebraic topology. These techniques may be robust and general. In one example, these techniques apply to almost any kind of data for which some qualitative idea of “closeness” or

“similarity” exists. The techniques discussed herein may be robust because the results may be relatively insensitive to noise in the data, user options, and even to errors in the specific details of the qualitative measure of similarity, which, in some embodiments, may be generally referred to as “the distance function” or “metric.” It will be appreciated that while the description of the algorithms below may seem general, the implementation of techniques described herein may apply to any level of generality.

[0150] In step 802, the input module 314 receives data S. In one example, a user identifies a data structure and then identifies ID and data fields. Data S may be based on the information within the ID and data fields. In various embodiments, data S is treated as being processed as a finite “similarity space,” where data S has a real-valued function d defined on pairs of points s and t in S, such that:

$$d(s, s) = 0$$

$$d(s, t) = d(t, s)$$

$$d(s, t) \geq 0$$

These conditions may be similar to requirements for a finite metric space, but the conditions may be weaker. In various examples, the function is a metric.

[0151] It will be appreciated that data S may be a finite metric space, or a generalization thereof, such as a graph or weighted graph. In some embodiments, data S be specified by a formula, an algorithm, or by a distance matrix which specifies explicitly every pairwise distance.

[0152] In step 804, the input module 314 generates reference space R. In one example, reference space R may be a well-known metric space (e.g., such as the real line). The reference space R may be defined by the user. In step 806, the analysis module 320 generates a map $ref()$ from S into R. The map $ref()$ from S into R may be called the “reference map.”

[0153] In one example, a reference of map from S is to a reference metric space R. R may be Euclidean space of some dimension, but it may also be the circle, torus, a tree, or other metric space. The map can be described by one or more filters (i.e., real valued functions on S). These filters can be defined by geometric invariants, such as the output of a density estimator, a notion of data depth, or functions specified by the origin of S as arising from a data set.

[0154] In step 808, the resolution module 218 generates a cover of R based on the resolution received from the user (e.g., filter(s), intervals, and overlap – see FIG. 7). The cover of R may be a finite collection of open sets (in the metric of R) such that every point in R lies in at least one of these sets. In various examples, R is k -dimensional Euclidean space, where k is the number of filter functions. More precisely in this example, R is a box in k -dimensional Euclidean space given by the product of the intervals $[\min_k, \max_k]$, where $\min_k$ is the minimum value of the k -th filter function on S , and $\max_k$ is the maximum value.

[0155] For example, suppose there are 2 filter functions, $F1$ and $F2$, and that $F1$'s values range from -1 to +1, and $F2$'s values range from 0 to 5. Then the reference space is the rectangle in the x/y plane with corners $(-1,0)$, $(1,0)$, $(-1, 5)$, $(1, 5)$, as every point s of S will give rise to a pair $(F1(s), F2(s))$ that lies within that rectangle.

[0156] In various embodiments, the cover of R is given by taking products of intervals of the covers of $[\min_k, \max_k]$ for each of the k filters. In one example, if the user requests 2 intervals and a 50% overlap for $F1$, the cover of the interval $[-1, +1]$ will be the two intervals $(-1.5, .5)$, $(-.5, 1.5)$. If the user requests 5 intervals and a 30% overlap for $F2$, then that cover of $[0, 5]$ will be $(-.3, 1.3)$, $(.7, 2.3)$, $(1.7, 3.3)$, $(2.7, 4.3)$, $(3.7, 5.3)$. These intervals may give rise to a cover of the 2-dimensional box by taking all possible pairs of intervals where the first of the pair is chosen from the cover for $F1$ and the second from the cover for $F2$. This may give rise to $2 * 5$, or 10, open boxes that covered the 2-dimensional reference space. However, it will be appreciated that the intervals may not be uniform, or that the covers of a k -dimensional box may not be constructed by products of intervals. In some embodiments, there are many other choices of intervals. Further, in various embodiments, a wide range of covers and/or more general reference spaces may be used.

[0157] In one example, given a cover, $C1, \dots, C_m$, of R , the reference map is used to assign a set of indices to each point in S , which are the indices of the C_j such that $\text{ref}(s)$ belongs to C_j . This function may be called $\text{ref_tags}(s)$. In a language such as Java, ref_tags would be a method that returned an $\text{int}[]$. Since the C 's cover R in this example, $\text{ref}(s)$ must lie in at least one of them, but the elements of the cover usually overlap one another, which means that points that “land near the edges” may well reside in multiple cover sets. In considering the two filter example, if $F1(s)$ is $-.99$, and $F2(s)$ is $.001$, then $\text{ref}(s)$ is $(-.99, .001)$, and this lies in the cover element $(-1.5, .5) \times (-.3, 1.3)$. Supposing that was labeled $C1$, the reference map may assign s to the set $\{1\}$. On the other hand, if t is mapped by $F1, F2$ to $(.1, 2.1)$, then $\text{ref}(t)$ will

be in $(-1.5, .5) \times (.7, 2.3)$, $(-.5, 1.5) \times (.7, 2.3)$, $(-1.5, .5) \times (1.7, 3.3)$, and $(-.5, 1.5) \times (1.7, 3.3)$, so the set of indices would have four elements for t .

[0158] Having computed, for each point, which “cover tags” it is assigned to, for each cover element, C_d , the points may be constructed, whose tags include d , as set $S(d)$. This may mean that every point s is in $S(d)$ for some d , but some points may belong to more than one such set. In some embodiments, there is, however, no requirement that each $S(d)$ is non-empty, and it is frequently the case that some of these sets are empty. In the non-parallelized version of some embodiments, each point x is processed in turn, and x is inserted into a hash-bucket for each j in $\text{ref_tags}(t)$ (that is, this may be how $S(d)$ sets are computed).

[0159] It will be appreciated that the cover of the reference space R may be controlled by the number of intervals and the overlap identified in the resolution (e.g., see FIG. 7). For example, the more intervals, the finer the resolution in S - that is, the fewer points in each $S(d)$, but the more similar (with respect to the filters) these points may be. The greater the overlap, the more times that clusters in $S(d)$ may intersect clusters in $S(e)$ - this means that more “relationships” between points may appear, but, in some embodiments, the greater the overlap, the more likely that accidental relationships may appear.

[0160] In step 810, the analysis module 320 clusters each $S(d)$ based on the metric, filter, and the space S . In some embodiments, a dynamic single-linkage clustering algorithm may be used to partition $S(d)$. It will be appreciated that any number of clustering algorithms may be used with embodiments discussed herein. For example, the clustering scheme may be k-means clustering for some k , single linkage clustering, average linkage clustering, or any method specified by the user.

[0161] The significance of the user-specified inputs may now be seen. In some embodiments, a filter may amount to a “forced stretching” in a certain direction. In some embodiments, the analysis module 320 may not cluster two points unless ALL of the filter values are sufficiently “related” (recall that while normally related may mean “close,” the cover may impose a much more general relationship on the filter values, such as relating two points s and t if $\text{ref}(s)$ and $\text{ref}(t)$ are sufficiently close to the same circle in the plane). In various embodiments, the ability of a user to impose one or more “critical measures” makes this technique more powerful than regular clustering, and the fact that these filters can be anything, is what makes it so general.

[0162] The output may be a simplicial complex, from which one can extract its 1-skeleton. The nodes of the complex may be partial clusters, (i.e., clusters constructed from subsets of S specified as the preimages of sets in the given covering of the reference space R).

[0163] In step 812, the visualization engine 322 identifies nodes which are associated with a subset of the partition elements of all of the $S(d)$ for generating an interactive visualization. For example, suppose that $S = \{1, 2, 3, 4\}$, and the cover is $C1, C2, C3$. Then if $\text{ref_tags}(1) = \{1, 2, 3\}$ and $\text{ref_tags}(2) = \{2, 3\}$, and $\text{ref_tags}(3) = \{3\}$, and finally $\text{ref_tags}(4) = \{1, 3\}$, then $S(1)$ in this example is $\{1, 4\}$, $S(2) = \{1, 2\}$, and $S(3) = \{1, 2, 3, 4\}$. If 1 and 2 are close enough to be clustered, and 3 and 4 are, but nothing else, then the clustering for $S(1)$ may be $\{1\} \{3\}$, and for $S(2)$ it may be $\{1, 2\}$, and for $S(3)$ it may be $\{1, 2\}, \{3, 4\}$. So the generated graph has, in this example, at most four nodes, given by the sets $\{1\}, \{4\}, \{1, 2\}$, and $\{3, 4\}$ (note that $\{1, 2\}$ appears in two different clusterings). Of the sets of points that are used, two nodes intersect provided that the associated node sets have a non-empty intersection (although this could easily be modified to allow users to require that the intersection is “large enough” either in absolute or relative terms).

[0164] Nodes may be eliminated for any number of reasons. For example, a node may be eliminated as having too few points and/or not being connected to anything else. In some embodiments, the criteria for the elimination of nodes (if any) may be under user control or have application-specific requirements imposed on it. For example, if the points are consumers, for instance, clusters with too few people in area codes served by a company could be eliminated. If a cluster was found with “enough” customers, however, this might indicate that expansion into area codes of the other consumers in the cluster could be warranted.

[0165] In step 814, the visualization engine 322 joins clusters to identify edges (e.g., connecting lines between nodes). Once the nodes are constructed, the intersections (e.g., edges) may be computed “all at once,” by computing, for each point, the set of node sets (not ref_tags , this time). That is, for each s in S , $\text{node_id_set}(s)$ may be computed, which is an $\text{int}[]$. In some embodiments, if the cover is well behaved, then this operation is linear in the size of the set S , and we then iterate over each pair in $\text{node_id_set}(s)$. There may be an edge between two node_id ’s if they both belong to the same $\text{node_id_set}()$ value, and the number of points in the intersection is precisely the number of different node_id sets in which that pair is seen. This means that, except for the clustering step (which is often quadratic in the size of the sets $S(d)$,

but whose size may be controlled by the choice of cover), all of the other steps in the graph construction algorithm may be linear in the size of S , and may be computed quite efficiently.

[0166] In step 816, the visualization engine 322 generates the interactive visualization of interconnected nodes (e.g., nodes and edges displayed in FIGS. 10 and 11).

[0167] It will be appreciated that it is possible, in some embodiments, to make sense in a fairly deep way of connections between various $\text{ref}()$ maps and/or choices of clustering. Further, in addition to computing edges (pairs of nodes), the embodiments described herein may be extended to compute triples of nodes, etc. For example, the analysis module 320 may compute simplicial complexes of any dimension (by a variety of rules) on nodes, and apply techniques from homology theory to the graphs to help users understand a structure in an automatic (or semi-automatic) way.

[0168] Further, it will be appreciated that uniform intervals in the covering may not always be a good choice. For example, if the points are exponentially distributed with respect to a given filter, uniform intervals can fail - in such case adaptive interval sizing may yield uniformly-sized $S(d)$ sets, for instance.

[0169] Further, in various embodiments, an interface may be used to encode techniques for incorporating third-party extensions to data access and display techniques. Further, an interface may be used to for third-party extensions to underlying infrastructure to allow for new methods for generating coverings, and defining new reference spaces.

[0170] FIG. 9 is an exemplary interactive visualization 900 in some embodiments. The display of the interactive visualization may be considered a “graph” in the mathematical sense. The interactive visualization comprises of two types of objects: nodes (e.g., nodes 902 and 906) (the colored balls) and the edges (e.g., edge 904) (the black lines). The edges connect pairs of nodes (e.g., edge 904 connects node 902 with node 906). As discussed herein, each node may represent a collection of data points (rows in the database identified by the user). In one example, connected nodes tend to include data points which are “similar to” (e.g., clustered with) each other. The collection of data points may be referred to as being “in the node.” The interactive visualization may be two-dimensional, three-dimensional, or a combination of both.

[0171] In various embodiments, connected nodes and edges may form a graph or structure. There may be multiple graphs in the interactive visualization. In one example, the interactive visualization may display two or more unconnected structures of nodes and edges.

[0172] The visual properties of the nodes and edges (such as, but not limited to, color, stroke color, text, texture, shape, coordinates of the nodes on the screen) can encode any data based property of the data points within each node. For example, coloring of the nodes and/or the edges may indicate (but is not limited to) the following:

- Values of fields or filters
- Any general functions of the data in the nodes (e.g., if the data were unemployment rates by state, then GDP of the states may be identifiable by color the nodes)
- Number of data points in the node

[0173] The interactive visualization 900 may contain a “color bar” 910 which may comprise a legend indicating the coloring of the nodes (e.g., balls) and may also identify what the colors indicate. For example, in FIG. 9, color bar 910 indicates that color is based on the density filter with blue (on the far left of the color bar 910) indicating “4.99e+03” and red (on the far right of the color bar 910) indicating “1.43e+04.” In general this might be expanded to show any other legend by which nodes and/or edges are colored. It will be appreciated that the, In some embodiments, the user may control the color as well as what the color (and/or stroke color, text, texture, shape, coordinates of the nodes on the screen) indicates.

[0174] The user may also drag and drop objects of the interactive visualization 900. In various embodiments, the user may reorient structures of nodes and edges by dragging one or more nodes to another portion of the interactive visualization (e.g., a window). In one example, the user may select node 902, hold node 902, and drag the node across the window. The node 902 will follow the user’s cursor, dragging the structure of edges and/or nodes either directly or indirectly connected to the node 902. In some embodiments, the interactive visualization 900 may depict multiple unconnected structures. Each structure may include nodes, however, none of the nodes of either structure are connected to each other. If the user selects and drags a node of the first structure, only the first structure will be reoriented with respect to the user action. The other structure will remain unchanged. The user may wish to reorient the structure in order to view nodes, select nodes, and/or better understand the relationships of the underlying data.

[0175] In one example, a user may drag a node to reorient the interactive visualization (e.g., reorient the structure of nodes and edges). While the user selects and/or drags the node, the nodes of the structure associated with the selected node may move apart from each other in order to provide greater visibility. Once the user lets go (e.g., deselects or drops the node that was dragged), the nodes of the structure may continue to move apart from each other.

[0176] In various embodiments, once the visualization module 322 generates the interactive display, the depicted structures may move by spreading out the nodes from each other. In one example, the nodes spread from each other slowly allowing the user to view nodes distinguish from each other as well as the edges. In some embodiments, the visualization module 322 optimizes the spread of the nodes for the user's view. In one example, the structure(s) stop moving once an optimal view has been reached.

[0177] It will be appreciated that the interactive visualization 900 may respond to gestures (e.g., multi-touch), stylus, or other interactions allowing the user to reorient nodes and edges and/or interacting with the underlying data.

[0178] The interactive visualization 900 may also respond to user actions such as when the user drags, clicks, or hovers a mouse cursor over a node. In some embodiments, when the user selects a node or edge, node information or edge information may be displayed. In one example, when a node is selected (e.g., clicked on by a user with a mouse or a mouse cursor hovers over the node), a node information box 908 may appear that indicates information regarding the selected node. In this example, the node information box 908 indicates an ID, box ID, number of elements (e.g., data points associated with the node), and density of the data associated with the node.

[0179] The user may also select multiple nodes and/or edges by clicking separate on each object, or drawing a shape (such as a box) around the desired objects. Once the objects are selected, a selection information box 912 may display some information regarding the selection. For example, selection information box 912 indicates the number of nodes selected and the total points (e.g., data points or elements) of the selected nodes.

[0180] The interactive visualization 900 may also allow a user to further interact with the display. Color option 914 allows the user to display different information based on color of the objects. Color option 914 in FIG. 9 is set to filter_Density, however, other filters may be chosen and the objects re-colored based on the selection. It will be appreciated that the objects may be colored based on any filter, property of data, or characterization. When a new option is

chosen in the color option 914, the information and/or colors depicted in the color bar 910 may be updated to reflect the change.

[0181] Layout checkbox 914 may allow the user to anchor the interactive visualization 900. In one example, the layout checkbox 914 is checked indicating that the interactive visualization 900 is anchored. As a result, the user will not be able to select and drag the node and/or related structure. Although other functions may still be available, the layout checkbox 914 may help the user keep from accidentally moving and/or reorienting nodes, edges, and/or related structures. It will be appreciated that the layout checkbox 914 may indicate that the interactive visualization 900 is anchored when the layout checkbox 914 is unchecked and that when the layout checkbox 914 is checked the interactive visualization 900 is no longer anchored.

[0182] The change parameters button 918 may allow a user to change the parameters (e.g., add/remove filters and/or change the resolution of one or more filters). In one example, when the change parameters button 918 is activated, the user may be directed back to the metric and filter selection interface window 600 (see FIG. 6) which allows the user to add or remove filters (or change the metric). The user may then view the filter parameter interface 700 (see FIG. 7) and change parameters (e.g., intervals and overlap) for one or more filters. The analysis module 320 may then re-analyze the data based on the changes and display a new interactive visualization 900 without again having to specify the data sets, filters, etc.

[0183] The find ID's button 920 may allow a user to search for data within the interactive visualization 900. In one example, the user may click the find ID's button 920 and receive a window allowing the user to identify data or identify a range of data. Data may be identified by ID or searching for the data based on properties of data and/or metadata. If data is found and selected, the interactive visualization 900 may highlight the nodes associated with the selected data. For example, selecting a single row or collection of rows of a database or spreadsheet may produce a highlighting of nodes whose corresponding partial cluster contains any element of that selection.

[0184] In various embodiments, the user may select one or more objects and click on the explain button 922 to receive in-depth information regarding the selection. In some embodiments, when the user selects the explain button 922, the information about the data from which the selection is based may be displayed. The function of the explain button 922 is further discussed with regard to FIG. 10.

[0185] In various embodiments, the interactive visualization 900 may allow the user to specify and identify subsets of interest, such as output filtering, to remove clusters or connections which are too small or otherwise uninteresting. Further, the interactive visualization 900 may provide more general coloring and display techniques, including, for example, allowing a user to highlight nodes based on a user-specified predicate, and coloring the nodes based on the intensity of user-specified weighting functions.

[0186] The interactive visualization 900 may comprise any number of menu items. The “Selection” menu may allow the following functions:

- Select singletons (select nodes which are not connected to other nodes)
- Select all (selects all the nodes and edges)
- Select all nodes (selects all nodes)
- Select all edges
- Clear selection (no selection)
- Invert Selection (selects the complementary set of nodes or edges)
- Select “small” nodes (allows the user to threshold nodes based on how many points they have)
- Select leaves (selects all nodes which are connected to long “chains” in the graph)
- Remove selected nodes
- Show in a table (shows the selected nodes and their associated data in a table)
- Save selected nodes (saves the selected data to whatever format the user chooses. This may allow the user to subset the data and create new datasources which may be used for further analysis.)

[0187] In one example of the “show in a table” option, information from a selection of nodes may be displayed. The information may be specific to the origin of the data. In various embodiments, elements of a database table may be listed, however, other methods specified by the user may also be included. For example, in the case of microarray data from gene expression data, heat maps may be used to view the results of the selections.

[0188] The interactive visualization 900 may comprise any number of menu items. The “Save” menu may allow the user to save the whole output in a variety of different formats such as (but not limited to):

- Image files (PNG/JPG/PDF/SVG etc.)

- Binary output (The interactive output is saved in the binary format. The user may reopen this file at any time to get this interactive window again)

In some embodiments, graphs may be saved in a format such that the graphs may be used for presentations. This may include simply saving the image as a pdf or png file, but it may also mean saving an executable .xml file, which may permit other users to use the search and save capability to the database on the file without having to recreate the analysis.

[0189] In various embodiments, a relationship between a first and a second analysis output / interactive visualization for differing values of the interval length and overlap percentage may be displayed. The formal relationship between the first and second analysis output / interactive visualization may be that when one cover refines the next, there is a map of simplicial complexes from the output of the first to the output of the second. This can be displayed by applying a restricted form of a three-dimensional graph embedding algorithm, in which a graph is the union of the graphs for the various parameter values and in which the connections are the connections in the individual graphs as well as connections from one node to its image in the following graph. The constituent graphs may be placed in its own plane in 3D space. In some embodiments, there is a restriction that each constituent graph remain within its associated plane. Each constituent graph may be displayed individually, but a small change of parameter value may result in the visualization of the adjacent constituent graph. In some embodiments, nodes in the initial graph will move to nodes in the next graph, in a readily visualizable way.

[0190] FIG. 10 is an exemplary interactive visualization 1000 displaying an explain information window 1002 in some embodiments. In various embodiments, the user may select a plurality of nodes and click on the explain button. When the explain button is clicked, the explain information window 1002 may be generated. The explain information window 1002 may identify the data associated with the selected object(s) as well as information (e.g., statistical information) associated with the data.

[0191] In some embodiments, the explain button allows the user to get a sense for which fields within the selected data fields are responsible for “similarity” of data in the selected nodes and the differentiating characteristics. There can be many ways of scoring the data fields. The explain information window 1002 (i.e., the scoring window in FIG. 10) is shown along with the selected nodes. The highest scoring fields may distinguish variables with respect to the rest of the data.

[0192] In one example, the explain information window 1002 indicates that data from fields day0-day6 has been selected. The minimum value of the data in all of the fields is 0. The explain information window 1002 also indicates the maximum values. For example, the maximum value of all of the data associated with the day0 field across all of the points of the selected nodes is 0.353. The average (i.e., mean) of all of the data associated with the day0 field across all of the points of the selected nodes is 0.031. The score may be a relative (e.g., normalized) value indicating the relative function of the filter; here, the score may indicate the relative density of the data associated with the day0 field across all of the points of the selected nodes. It will be appreciated that any information regarding the data and/or selected nodes may appear in the explain information window 1002.

[0193] It will be appreciated that the data and the interactive visualization 1000 may be interacted with in any number of ways. The user may interact with the data directly to see where the graph corresponds to the data, make changes to the analysis and view the changes in the graph, modify the graph and view changes to the data, or perform any kind of interaction.

[0194] FIG. 11 is a flowchart 1200 of functionality of the interactive visualization in some embodiments. In step 1202, the visualization engine 322 receives the analysis from the analysis module 320 and graphs nodes as balls and edges as connectors between balls 1202 to create interactive visualization 900 (see FIG. 9).

[0195] In step 1204, the visualization engine 322 determines if the user is hovering a mouse cursor (or has selected) a ball (i.e., a node). If the user is hovering a mouse cursor over a ball or selecting a ball, then information is displayed regarding the data associated with the ball. In one example, the visualization engine 322 displays a node information window 908.

[0196] If the visualization engine 322 does not determine that the user is hovering a mouse cursor (or has selected) a ball, then the visualization engine 322 determines if the user has selected balls on the graph (e.g., by clicking on a plurality of balls or drawing a box around a plurality of balls). If the user has selected balls on the graph, the visualization engine 322 may highlight the selected balls on the graph in step 1110. The visualization engine 322 may also display information regarding the selection (e.g., by displaying a selection information window 912). The user may also click on the explain button 922 to receive more information associated with the selection (e.g., the visualization engine 322 may display the explain information window 1002).

[0197] In step 1112, the user may save the selection. For example, the visualization engine 322 may save the underlying data, selected metric, filters, and/or resolution. The user may then access the saved information and create a new structure in another interactive visualization 900 thereby allowing the user to focus attention on a subset of the data.

[0198] If the visualization engine 322 does not determine that the user has selected balls on the graph, the visualization engine 322 may determine if the user selects and drags a ball on the graph in step 1114. If the user selects and drags a ball on the graph, the visualization engine 322 may reorient the selected balls and any connected edges and balls based on the user's action in step 1116. The user may reorient all or part of the structure at any level of granularity.

[0199] It will be appreciated that although FIG. 11 discussed the user hovering over, selecting, and/or dragging a ball, the user may interact with any object in the interactive visualization 900 (e.g., the user may hover over, select, and/or drag an edge). The user may also zoom in or zoom out using the interactive visualization 900 to focus on all or a part of the structure (e.g., one or more balls and/or edges).

[0200] Further, although balls are discussed and depicted in FIGS. 9-11, it will be appreciated that the nodes may be any shape and appear as any kind of object. Further, although some embodiments described herein discuss an interactive visualization being generated based on the output of algebraic topology, the interactive visualization may be generated based on any kind of analysis and is not limited.

[0201] For years, researchers have been collecting huge amounts of data on breast cancer, yet we are still battling the disease. Complexity, rather than quantity, is one of the fundamental issues in extracting knowledge from data. A topological data exploration and visualization platform may assist the analysis and assessment of complex data. In various embodiments, a predictive and visual cancer map generated by the topological data exploration and visualization platform may assist physicians to determine treatment options.

[0202] In one example, a breast cancer map visualization may be generated based on the large amount of available information already generated by many researchers. Physicians may send biopsy data directly to a cloud-based server which may localize a new patient's data within the breast cancer map visualization. The breast cancer map visualization may be annotated (e.g., labeled) such that the physician may view outcomes of patients with similar profiles as well as different kinds of statistical information such as survival probabilities. Each

new data point from a patient may be incorporated into the breast cancer map visualization to improve accuracy of the breast cancer map visualization over time.

[0203] Although the following examples are largely focused on cancer map visualizations, it will be appreciated that at least some of the embodiments described herein may apply to any biological condition and not be limited to cancer and/or disease. For example, some embodiments, may apply to different industries.

[0204] FIG. 12 is a flowchart for generating a cancer map visualization utilizing biological data of a plurality of patients in some embodiments. In various embodiments, the processing of data and user-specified options is motivated by techniques from topology and, in some embodiments, algebraic topology. As discussed herein, these techniques may be robust and general. In one example, these techniques apply to almost any kind of data for which some qualitative idea of “closeness” or “similarity” exists. It will be appreciated that the implementation of techniques described herein may apply to any level of generality.

[0205] In various embodiments, a cancer map visualization is generated using genomic data linked to clinical outcomes (i.e., medical characteristics) which may be used by physicians during diagnosis and/or treatment. Initially, publicly available data sets may be integrated to construct the topological map visualizations of patients (e.g., breast cancer patients). It will be appreciated that any private, public, or combination of private and public data sets may be integrated to construct the topological map visualizations. A map visualization may be based on biological data such as, but not limited to, gene expression, sequencing, and copy number variation. As such, the map visualization may comprise many patients with many different types of collected data. Unlike traditional methods of analysis where distinct studies of breast cancer appear as separate entities, the map visualization may fuse disparate data sets while utilizing many datasets and data types.

[0206] In various embodiments, a new patient may be localized on the map visualization. With the map visualization for subtypes of a particular disease and a new patient diagnosed with the disease, point(s) may be located among the data points used in computing the map visualization (e.g., nearest neighbor) which is closest to the new patient point. The new patient may be labeled with nodes in the map visualization containing the closest neighbor. These nodes may be highlighted to give a physician the location of the new patient among the patients in the reference data set. The highlighted nodes may also give the physician the location of the new patient relative to annotated disease subtypes. Nearest neighbor is further

described in U.S. Non-Provisional Patent Application Serial No. 13/648,237 filed October 9, 2012 and entitled “Systems and Methods for Mapping New Patient Information to Historic Outcomes for Treatment Assistance,” the entirety of which is incorporated herein by reference.

[0207] The visualization map may be interactive and/or searchable in real-time thereby potentially enabling extended analysis and providing speedy insight into treatment.

[0208] In step 1202, biological data and clinical outcomes of previous patients may be received. The clinical outcomes may be medical characteristics. Biological data is any data that may represent a condition (e.g., a medical condition) of a person. Biological data may include any health related, medical, physical, physiological, pharmaceutical data associated with one or more patients. In one example, biological data may include measurements of gene expressions for any number of genes. In another example, biological data may include sequencing information (e.g., RNA sequencing).

[0209] In various embodiments, biological data for a plurality of patients may be publicly available. For example, various medical health facilities and/or public entities may provide gene expression data for a variety of patients. In addition to the biological data, information regarding any number of clinical outcomes, treatments, therapies, diagnoses and/or prognoses may also be provided. It will be appreciated that any kind of information may be provided in addition to the biological data.

[0210] The biological data, in one example, may be similar to data S as discussed with regard to step 802 of FIG. 8. The biological data may include ID fields that identify patients and data fields that are related to the biological information (e.g., gene expression measurements).

[0211] FIG. 13 is an exemplary data structure 1302 including biological data 1304a-1304y for a number of patients 1308a-1308n that may be used to generate the cancer map visualization in some embodiments. Column 1302 represents different patient identifiers for different patients. The patient identifiers may be any identifier.

[0212] At least some biological data may be contained within gene expression measurements 1304a-1304y. In FIG. 13, “y” represents any number. For example, there may be 50,000 or more separate columns for different gene expressions related to a single patient or related to one or more samples from a patient. It will be appreciated that column 1304a may represent a gene expression measurement for each patient (if any for some patients) associated

with the patient identifiers in column 1302. The column 1304b may represent a gene expression measurement of one or more genes that are different than that of column 1304a. As discussed, there may be any number of columns representing different gene expression measurements.

[0213] Column 1306 may include any number of clinical outcomes, prognoses, diagnoses, reactions, treatments, and/or any other information associated with each patient. All or some of the information contained in column 1306 may be displayed (e.g., by a label or an annotation that is displayed on the visualization or available to the user of the visualization via clicking) on or for the visualization.

[0214] Rows 1308a-1308n each contains biological data associated with the patient identifier of the row. For example, gene expressions in row 1308a are associated with patient identifier P1. As similarly discussed with regard to “y” herein, “n” represents any number. For example, there may be 100,000 or more separate rows for different patients.

[0215] It will be appreciated that there may be any number of data structures that contain any amount of biological data for any number of patients. The data structure(s) may be utilized to generate any number of map visualizations.

[0216] In step 1204, the analysis server may receive a filter selection. In some embodiments, the filter selection is a density estimation function. It will be appreciated that the filter selection may include a selection of one or more functions to generate a reference space.

[0217] In step 1206, the analysis server performs the selected filter(s) on the biological data of the previous patients to map the biological data into a reference space. In one example, a density estimation function, which is well known in the art, may be performed on the biological data (e.g., data associated with gene expression measurement data 1304a-1304y) to relate each patient identifier to one or more locations in the reference space (e.g., on a real line).

[0218] In step 1208, the analysis server may receive a resolution selection. The resolution may be utilized to identify overlapping portions of the reference space (e.g., a cover of the reference space R) in step 1210.

[0219] As discussed herein, the cover of R may be a finite collection of open sets (in the metric of R) such that every point in R lies in at least one of these sets. In various examples, R

is k -dimensional Euclidean space, where k is the number of filter functions. It will be appreciated that the cover of the reference space R may be controlled by the number of intervals and the overlap identified in the resolution (e.g., see FIG. 7). For example, the more intervals, the finer the resolution in S (e.g., the similarity space of the received biological data) - that is, the fewer points in each $S(d)$, but the more similar (with respect to the filters) these points may be. The greater the overlap, the more times that clusters in $S(d)$ may intersect clusters in $S(e)$ - this means that more “relationships” between points may appear, but, in some embodiments, the greater the overlap, the more likely that accidental relationships may appear.

[0220] In step 1212, the analysis server receives a metric to cluster the information of the cover in the reference space to partition $S(d)$. In one example, the metric may be a Pearson Correlation. The clusters may form the groupings (e.g., nodes or balls). Various cluster means may be used including, but not limited to, a single linkage, average linkage, complete linkage, or k -means method.

[0221] As discussed herein, in some embodiments, the analysis module 320 may not cluster two points unless filter values are sufficiently “related” (recall that while normally related may mean “close,” the cover may impose a much more general relationship on the filter values, such as relating two points s and t if $\text{ref}(s)$ and $\text{ref}(t)$ are sufficiently close to the same circle in the plane where $\text{ref}()$ represents one or more filter functions). The output may be a simplicial complex, from which one can extract its 1-skeleton. The nodes of the complex may be partial clusters, (i.e., clusters constructed from subsets of S specified as the preimages of sets in the given covering of the reference space R).

[0222] In step 1214, the analysis server may generate the visualization map with nodes representing clusters of patient members and edges between nodes representing common patient members. In one example, the analysis server identifies nodes which are associated with a subset of the partition elements of all of the $S(d)$ for generating an interactive visualization.

[0223] As discussed herein, for example, suppose that $S = \{1, 2, 3, 4\}$, and the cover is $C1$, $C2$, $C3$. Suppose cover $C1$ contains $\{1, 4\}$, $C2$ contains $\{1, 2\}$, and $C3$ contains $\{1, 2, 3, 4\}$. If 1 and 2 are close enough to be clustered, and 3 and 4 are, but nothing else, then the clustering for $S(1)$ may be $\{1\}$, $\{4\}$, and for $S(2)$ it may be $\{1, 2\}$, and for $S(3)$ it may be $\{1, 2\}$, $\{3, 4\}$. So the generated graph has, in this example, at most four nodes, given by the sets $\{1\}$, $\{4\}$, $\{1, 2\}$, and $\{3, 4\}$ (note that $\{1, 2\}$ appears in two different clusterings). Of the sets of points that are

used, two nodes intersect provided that the associated node sets have a non-empty intersection (although this could easily be modified to allow users to require that the intersection is “large enough” either in absolute or relative terms).

[0224] As a result of clustering, member patients of a grouping may share biological similarities (e.g., similarities based on the biological data).

[0225] The analysis server may join clusters to identify edges (e.g., connecting lines between nodes). Clusters joined by edges (i.e., interconnections) share one or more member patients. In step 1216, a display may display a visualization map with attributes based on the clinical outcomes contained in the data structures (e.g., see FIG. 13 regarding clinical outcomes). Any labels or annotations may be utilized based on information contained in the data structures. For example, treatments, prognoses, therapies, diagnoses, and the like may be used to label the visualization. In some embodiments, the physician or other user of the map visualization accesses the annotations or labels by interacting with the map visualization.

[0226] The resulting cancer map visualization may reveal interactions and relationships that were obscured, untested, and/or previously not recognized.

[0227] FIG. 14 is an exemplary visualization displaying the cancer map visualization 1400 in some embodiments. The cancer map visualization 1400 represents a topological network of cancer patients. The cancer map visualization 1400 may be based on publicly and/or privately available data.

[0228] In various embodiments, the cancer map visualization 1400 is created using gene expression profiles of excised tumors. Each node (i.e., ball or grouping displayed in the map visualization 1400) contains a subset of patients with similar genetic profiles.

[0229] As discussed herein, one or more patients (i.e., patient members of each node or grouping) may occur in multiple nodes. A patient may share a similar genetic profile with multiple nodes or multiple groupings. In one example, of 50,000 different gene expressions of the biological data, multiple patients may share a different genetic profiles (e.g., based on different gene expression combinations) with different groupings. When a patient shares a similar genetic profile with different groupings or nodes, the patient may be included within the groupings or nodes.

[0230] The cancer map visualization 1400 comprises groupings and interconnections that are associated with different clinical outcomes. All or some of the clinical outcomes may be

associated with the biological data that generated the cancer map visualization 1400. The cancer map visualization 1400 includes groupings associated with survivors 1402 and groupings associated with non-survivors 1404. The cancer map visualization 1400 also includes different groupings associated with estrogen receptor positive non-survivors 1406, estrogen receptor negative non-survivors 1408, estrogen receptor positive survivors 1410, and estrogen receptor negative survivors 1412.

[0231] In various embodiments, when one or more patients are members of two or more different nodes, the nodes are interconnected by an edge (e.g., a line or interconnection). If there is not an edge between the two nodes, then there are no common member patients between the two nodes. For example, grouping 1414 shares at least one common member patient with grouping 1418. The intersection of the two groupings is represented by edge 1416. As discussed herein, the number of shared member patients of the two groupings may be represented in any number of ways including color of the interconnection, color of the groupings, size of the interconnection, size of the groupings, animations of the interconnection, animations of the groupings, brightness, or the like. In some embodiments, the number and/or identifiers of shared member patients of the two groupings may be available if the user interacts with the groupings 1414 and/or 1418 (e.g., draws a box around the two groupings and the interconnection utilizing an input device such as a mouse).

[0232] In various embodiments, a physician, on obtaining some data on a breast tumor, direct the data to an analysis server (e.g., analysis server 208 over a network such as the Internet) which may localize the patient relative to one or more groupings on the cancer map visualization 1400. The context of the cancer map visualization 1400 may enable the physician to assess various possible outcomes (e.g., proximity of representation of new patient to the different associations of clinical outcomes).

[0233] FIG. 15 is a flowchart of for positioning new patient data relative to a cancer map visualization in some embodiments. In step 1502, new biological data of a new patient is received. In various embodiments, an input module 314 of an analysis server (e.g., analysis server 208 of FIGS. 1 and 2) may receive biological data of a new patient from a physician or medical facility that performed analysis of one or more samples to generate the biological data. The biological data may be any data that represents a biological data of the new patient including, for example, gene expressions, sequencing information, or the like.

[0234] In some embodiments, the analysis server 208 may comprise a new patient distance module and a location engine. In step 1504, the new patient distance module determines distances between the biological data of each patient of the cancer map visualization 1600 and the new biological data from the new patient. For example, the previous biological data that was utilized in the generation of the cancer map visualization 1600 may be stored in mapped data structures. Distances may be determined between the new biological data of the new patient and each of the previous patient's biological data in the mapped data structure.

[0235] It will be appreciated that distances may be determined in any number of ways using any number of different metrics or functions. Distances may be determined between the biological data of the previous patients and the new patients. For example, a distance may be determined between a first gene expression measurement of the new patient and each (or a subset) of the first gene expression measurements of the previous patients (e.g., the distance between G1 of the new patient and G1 of each previous patient may be calculated). Distances may be determined between all (or a subset of) other gene expression measurements of the new patient to the gene expression measurements of the previous patients.

[0236] In various embodiments, a location of the new patient on the cancer map visualization 1600 may be determined relative to the other member patients utilizing the determined distances.

[0237] In step 1506, the new patient distance module may compare distances between the patient members of each grouping to the distances determined for the new patient. The new patient may be located in the grouping of patient members that are closest in distance to the new patient. In some embodiments, the new patient location may be determined to be within a grouping that contains the one or more patient members that are closest to the new patient (even if other members of the grouping have longer distances with the new patient). In some embodiments, this step is optional.

[0238] In various embodiments, a representative patient member may be determined for each grouping. For example, some or all of the patient members of a grouping may be averaged or otherwise combined to generate a representative patient member of the grouping (e.g., the distances and/or biological data of the patient members may be averaged or aggregated). Distances may be determined between the new patient biological data and the averaged or combined biological data of one or more representative patient members of one or more groupings. The location engine may determine the location of the new patient based on

the distances. In some embodiments, once the closest distance between the new patient and the representative patient member is found, distances may be determined between the new patient and the individual patient members of the grouping associated with the closest representative patient member.

[0239] In optional step 1508, a diameter of the grouping with the one or more of the patient members that are closest to the new patient (based on the determined distances) may be determined. In one example, the diameters of the groupings of patient members closest to the new patient are calculated. The diameter of the grouping may be a distance between two patient members who are the farthest from each other when compared to the distances between all patient members of the grouping. If the distance between the new patient and the closest patient member of the grouping is less than the diameter of the grouping, the new patient may be located within the grouping. If the distance between the new patient and the closest patient member of the grouping is greater than the diameter of the grouping, the new patient may be outside the grouping (e.g., a new grouping may be displayed on the cancer map visualization with the new patient as the single patient member of the grouping). If the distance between the new patient and the closest patient member of the grouping is equal to the diameter of the grouping, the new patient may be placed within or outside the grouping.

[0240] It will be appreciated that the determination of the diameter of the grouping is not required in determining whether the new patient location is within or outside of a grouping. In various embodiments, a distribution of distances between member patients and between member patients and the new patient is determined. The decision to locate the new patient within or outside of the grouping may be based on the distribution. For example, if there is a gap in the distribution of distances, the new patient may be separated from the grouping (e.g., as a new grouping). In some embodiments, if the gap is greater than a preexisting threshold (e.g., established by the physician, other user, or previously programmed), the new patient may be placed in a new grouping that is placed relative to the grouping of the closest member patients. The process of calculating the distribution of distances of candidate member patients to determine whether there may be two or more groupings may be utilized in generation of the cancer map visualization (e.g., in the process as described with regard to FIG. 12). It will be appreciated that there may be any number of ways to determine whether a new patient should be included within a grouping of other patient members.

[0241] In step 1510, the location engine determines the location of the new patient relative to the member patients and/or groupings of the cancer map visualization. The new location may be relative to the determined distances between the new patient and the previous patients. The location of the new patient may be part of a previously existing grouping or may form a new grouping.

[0242] In some embodiments, the location of the new patient with regard to the cancer map visualization may be performed locally to the physician. For example, the cancer map visualization 1400 may be provided to the physician (e.g., via digital device). The physician may load the new patient's biological data locally and the distances may be determined locally or via a cloud-based server. The location(s) associated with the new patient may be overlaid on the previously existing cancer map visualization either locally or remotely.

[0243] Those skilled in the art will appreciate that, in some embodiments, the previous state of the cancer map visualization (e.g., cancer map visualization 1400) may be retained or otherwise stored and a new cancer map visualization generated utilizing the new patient biological data (e.g., in a method similar to that discussed with regard to FIG. 12). The newly generated map may be compared to the previous state and the differences may be highlighted thereby, in some embodiments, highlighting the location(s) associated with the new patient. In this way, distances may be not be calculated as described with regard to FIG. 15, but rather, the process may be similar to that as previously discussed.

[0244] FIG. 16 is an exemplary visualization displaying the cancer map including positions for three new cancer patients in some embodiments. The cancer map visualization 1400 comprises groupings and interconnections that are associated with different clinical outcomes as discussed with regard to FIG. 14. All or some of the clinical outcomes may be associated with the biological data that generated the cancer map visualization 1400. The cancer map visualization 1400 includes different groupings associated with survivors 1402, groupings associated with non-survivors 1404, estrogen receptor positive non-survivors 1406, estrogen receptor negative non-survivors 1408, estrogen receptor positive survivors 1410, and estrogen receptor negative survivors 1412.

[0245] The cancer map visualization 1400 includes three locations for three new breast cancer patients. The breast cancer patient location 1602 is associated with the clinical outcome of estrogen receptor positive survivors. The breast cancer patient location 1604 is associated with the clinical outcome of estrogen receptor negative survivors. Unfortunately, breast cancer

patient location 1606 is associated with estrogen receptor negative non-survivors. Based on the locations, a physician may consider different diagnoses, prognoses, treatments, and therapies to maintain or attempt to move the breast cancer patient to a different location utilizing the cancer map visualization 1400.

[0246] In some embodiments, the physician may assess the underlying biological data associated with any number of member patients of any number of groupings to better understand the genetic similarities and/or dissimilarities. The physician may utilize the information to make better informed decisions.

[0247] The patient location 1604 is highlighted on the cancer map visualization 1400 as active (e.g., selected by the physician). It will be appreciated that the different locations may be of any color, size, brightness, and/or animated to highlight the desired location(s) for the physician. Further, although only one location is identified for three different breast cancer patients, any of the breast cancer patients may have multiple locations indicating different genetic similarities.

[0248] It will be appreciated that the cancer map visualization 1400 may be updated with new information at any time. As such, as new patients are added to the cancer map visualization 1400, the new data updates the visualization such that as future patients are placed in the map, the map may already include the updated information. As new information and/or new patient data is added to the cancer map visualization 1400, the cancer map visualization 1400 may improve as a tool to better inform physicians or other medical professionals.

[0249] In various embodiments, the cancer map visualization 1400 may track changes in patients over time. For example, updates to a new patient may be visually tracked as changes in are measured in the new patient's biological data. In some embodiments, previous patient data is similarly tracked which may be used to determine similarities of changes based on condition, treatment, and/or therapies, for example. In various embodiments, velocity of change and/or acceleration of change of any number of patients may be tracked over time using or as depicted on the cancer map visualization 1400. Such depictions may assist the treating physician or other personnel related to the treating physician to better understand changes in the patient and provide improved, current, and/or updated diagnoses, prognoses, treatments, and/or therapies.

[0250] FIG. 17 is a flowchart of utilization the visualization and positioning of new patient data in some embodiments. In various embodiments, a physician may collect amounts of genomic information from tumors removed from a new patient, input the data (e.g., upload the data to an analysis server), and receive a map visualization with a location of the new patient. The new patient's location within the map may offer the physician new information about the similarities to other patients. In some embodiments, the map visualization may be annotated so that the physician may check the outcomes of previous patients in a given region of the map visualization are distributed and then use the information to assist in decision-making for diagnosis, treatment, prognosis, and/or therapy.

[0251] In step 1702, a medical professional or other personnel may remove a sample from a patient. The sample may be of a tumor, blood, or any other biological material. In one example, a medical professional performs a tumor excision. Any number of samples may be taken from a patient.

[0252] In step 1704, the sample(s) may be provided to a medical facility to determine new patient biological data. In one example, the medical facility measures genomic data such as gene expression of a number of genes or protein levels.

[0253] In step 1706, the medical professional or other entity associated with the medical professional may receive the new patient biological data based on the sample(s) from the new patient. In one example, a physician may receive the new patient biological data. The physician may provide all or some of the new patient biological data to an analysis server over the Internet (e.g., the analysis server may be a cloud-based server). In some embodiments, the analysis server is the analysis server 208 of FIG. 1. In some embodiments, the medical facility that determines the new patient biological data provides the biological data in an electronic format which may be uploaded to the analysis server. In some embodiments, the medical facility that determines the new patient biological data (e.g., the medical facility that measures the genomic data) provide the biological data to the analysis server at the request of the physician or others associated with the physician. It will be appreciated that the biological data may be provided to the analysis server in any number of ways.

[0254] The analysis server may be any digital device and may not be limited to a digital device on a network. In some embodiments, the physician may have access to the digital device. For example, the analysis server may be a table, personal computer, local server, or any other digital device.

[0255] Once the analysis server receives the biological data of the new patient, the new patient may be localized in the map visualization and the information may be sent back to the physician in step 1708. The visualization may be a map with nodes representing clusters of previous patient members and edges between nodes representing common patient members. The visualization may further depict one or more locations related to the biological data of the new patient.

[0256] The map visualization may be provided to the physician or other associated with the physician in real-time. For example, once the biological data associated with the new patient is provided to the analysis server, the analysis server may provide the map visualization back to the physician or other associated with the physician within a reasonably short time (e.g., within seconds or minutes). In some embodiments, the physician may receive the map visualization over any time.

[0257] The map visualization may be provided to the physician in any number of ways. For example, the physician may receive the map visualization over any digital device such as, but not limited to, an office computer, iPad, tablet device, media device, smartphone, e-reader, or laptop.

[0258] In step 1710, the physician may assess possible different clinical outcomes based on the map visualization. In one example, the map-aided physician may make decisions on therapy and treatments depending on where the patient lands on the visualization (e.g., survivor or non-survivor). The map visualization may include annotations or labels that identify one or more sets of groupings and interconnections as being associated with one or more clinical outcomes. The physician may assess possible clinical outcomes based on the position(s) on the map associated with the new patient.

[0259] As described above, interesting continuous functions on a metric space (e.g., a similarity space) allow the application of systems and methods described herein. In various embodiments, functions may be performed on data within the metric space to project data into the reference space. Having the function(s) to project the data from the metric space to the similarity space (i.e., a lens function) dependent on a small number of coordinates (e.g., counting a number of uses of a small collection of words) is a fairly simple way to achieve continuity in most metrics, and the resulting lenses may be suitable for interpolation. However, such lenses may be of limited use on high-dimensional data, and if the interesting

features of the space were captured in those few dimensions, there may be no point keeping the rest of the coordinates.

[0260] In practice, lenses which incorporate intrinsic properties of the metric (e.g., the function on the data to generate the metric space), such as density or centrality, are more likely to capture features of the space, absent special knowledge of the particular data set, than functions which depend on a few coordinates. One example method of dimensionality reduction (which is a way to think of a small collection of lenses applied jointly) are variants of “Stochastic Neighbor Embedding” (aka SNE). The underlying intuition in stochastic neighbor embedding is to map the high dimensional space to points in a low-dimensional Euclidean space, typically two or three dimensions, define a potential function on the points which penalizes them for being either closer or farther apart in the embedding than they are in the high-dimensional space, and move points around to minimize the potential. This may be effectively like a graph-layout problem, where a (potentially) high-dimensional space, an arbitrary combinatorial graph, is to be faithfully represented by a two-dimensional picture.

[0261] Some example methods amount to computing a global potential and then optimizing the placement by the same optimization techniques used in applications of artificial neural network. These methods produce very nice pictures and the lenses can be remarkably effective with TDA, but they may be computationally expensive. Some embodiments described herein allow for the use of less computationally expensive layout mechanisms and methods.

[0262] FIG. 18 is a block diagram of an exemplary digital device 1800. The digital device 1800 comprises a processor 1802, a memory system 1804, a storage system 1806, a communication network interface 1808, an I/O interface 1810, and a display interface 1812 communicatively coupled to a bus 1814. The processor 1802 may be configured to execute executable instructions (e.g., programs). In some embodiments, the processor 1802 comprises circuitry or any processor capable of processing the executable instructions.

[0263] The memory system 1804 is any memory configured to store data. Some examples of the memory system 1804 are storage devices, such as RAM or ROM. The memory system 1804 can comprise the ram cache. In various embodiments, data is stored within the memory system 1804. The data within the memory system 1804 may be cleared or ultimately transferred to the storage system 1806.

[0264] The storage system 1806 is any storage configured to retrieve and store data. Some examples of the storage system 1806 are flash drives, hard drives, optical drives, and/or magnetic tape. In some embodiments, the digital device 1800 includes a memory system 1804 in the form of RAM and a storage system 1806 in the form of flash data. Both the memory system 1804 and the storage system 1806 comprise computer readable media which may store instructions or programs that are executable by a computer processor including the processor 1802.

[0265] The communication network interface (com. network interface) 1808 can be coupled to a communication network (e.g., communication network 204) via the link 1816. The communication network interface 1808 may support communication over an Ethernet connection, a serial connection, a parallel connection, or an ATA connection, for example. The communication network interface 1808 may also support wireless communication (e.g., 1802.11 a/b/g/n, WiMax). It will be apparent to those skilled in the art that the communication network interface 1808 can support many wired and wireless standards.

[0266] The optional input/output (I/O) interface 1810 is any device that receives input from the user and output data. The optional display interface 1812 is any device that may be configured to output graphics and data to a display. In one example, the display interface 1812 is a graphics adapter.

[0267] It will be appreciated by those skilled in the art that the hardware elements of the digital device 1800 are not limited to those depicted in FIG. 18. A digital device 1800 may comprise more or less hardware elements than those depicted. Further, hardware elements may share functionality and still be within various embodiments described herein. In one example, encoding and/or decoding may be performed by the processor 1802 and/or a co-processor located on a GPU.

[0268] Systems and methods described herein include a framework for detection of fraud, waste, and/or abuse. In some embodiments, the framework may assist in identifying and/or prioritizing suspicious entities to investigate for potential fraud, waste, and/or abuse. The framework may take information provided by multiple entities and/or from monitoring of multiple entities.

[0269] In one example, the detection process may utilize information from health care system(s). Examples of information from a health care system may include data from medical billing, medical coding, reimbursement, coverage, equipment usage, prescriptions, treatments,

and/or clinical workflow. Entities can be any unit level, including but not limited to claimants, beneficiaries, physicians, and providers.

[0270] In some embodiments, the framework groups entities based on information from one or more health care systems. The framework may improve detection within these groups by, for example, prioritizing information that characterizes entity behavior. The framework may also allow for integration of new information associated with one or more entities that allow new groupings of entities and/or re-weightings (e.g., updated assessment).

[0271] FIG. 19 is a block diagram of an exemplary analysis server 208 including a normalization module 1902, a fraud or abuse grouping module 1904, a reporting module 1906, and a ranking module 1908 in some embodiments. The exemplary analysis server 208 depicted in FIG. 19 may be similar to the exemplary analysis server 208 depicted in FIG. 2 and FIG. 3. In exemplary embodiments, the analysis server 208 comprises a processor 302, input/output (I/O) interface 304, a communication network interface 306, a memory system 308, and a storage system 310.

[0272] The storage system 310 comprises a plurality of modules utilized by embodiments of the present invention. A module may be hardware (e.g., an ASIC), software (e.g., including instructions executable by a processor), or a combination of both. In one embodiment, the storage system 310 comprises a processing module 312 which comprises an input module 314, a filter module 316, a resolution module 318, an analysis module 320, a visualization engine 322, a database storage 324, a normalization module 1902, a fraud or abuse grouping module 1904, a reporting module 1906, and a ranking module 1908. Alternative embodiments of the analysis server 208 and/or the storage system 310 may comprise more, less, or functionally equivalent components and modules.

[0273] The input module 314 may be configured to receive commands and preferences from the user device 202a. In various examples, the input module 314 receives selections from the user which will be used to perform the analysis. The output of the analysis may be an interactive visualization.

[0274] The input module 314 may provide the user a variety of interface windows allowing the user to select and access a database, choose fields associated with the database, choose a metric, choose one or more filters, and identify resolution parameters for the analysis. In one example, the input module 314 receives a database identifier and accesses a large multi-dimensional database. The input module 314 may scan the database and provide the user with

an interface window allowing the user to identify an ID field. An ID field is an identifier for each data point. In one example, the identifier is unique. The same column name may be present in the table from which filters are selected. After the ID field is selected, the input module 314 may then provide the user with another interface window to allow the user to choose one or more data fields from a table of the database.

[0275] Although interactive windows may be described herein, it will be appreciated that any window, graphical user interface, and/or command line may be used to receive or prompt a user or user device 202a for information.

[0276] The filter module 316 may subsequently provide the user with an interface window to allow the user to select a metric to be used in analysis of the data within the chosen data fields. The filter module 316 may also allow the user to select and/or define one or more filters.

[0277] The resolution module 318 may allow the user to select a resolution, including filter parameters. In one example, the user enters a number of intervals and a percentage overlap for a filter.

[0278] The analysis module 320 may perform data analysis based on the database and the information provided by the user. In various embodiments, the analysis module 320 performs an algebraic topological analysis to identify structures and relationships within data and clusters of data. It will be appreciated that the analysis module 320 may use parallel algorithms or use generalizations of various statistical techniques (e.g., generalizing the bootstrap to zig-zag methods) to increase the size of data sets that can be processed. The analysis is further discussed in FIG. 8. It will be appreciated that the analysis module 320 is not limited to algebraic topological analysis but may perform any analysis.

[0279] The visualization engine 322 generates an interactive visualization including the output from the analysis module 320. The interactive visualization allows the user to see all or part of the analysis graphically. The interactive visualization also allows the user to interact with the visualization. For example, the user may select portions of a graph from within the visualization to see and/or interact with the underlying data and/or underlying analysis. The user may then change the parameters of the analysis (e.g., change the metric, filter(s), or resolution(s)) which allows the user to visually identify relationships in the data that may be otherwise undetectable using prior means. The interactive visualization is further described in FIGS. 9-11.

[0280] The database storage 324 is configured to store all or part of the database that is being accessed. In some embodiments, the database storage 324 may store saved portions of the database. Further, the database storage 324 may be used to store user preferences, parameters, and analysis output thereby allowing the user to perform many different functions on the database without losing previous work.

[0281] The normalization module 1902, the fraud or abuse grouping module 1904, the reporting module 1906, and the ranking module 1908 may work together to report or assist in identifying those entities that merit further investigation for potential fraud or abuse. In some embodiments, the normalization module 1902, the fraud or abuse grouping module 1904, the reporting module 1906, and the ranking module 1908 may work together to generate a report or assist in identifying those features (e.g., columns of data) that are most significantly related to possible cases of fraud or abuse.

[0282] In various embodiments, the input module 314 may receive data regarding a number of entities (e.g., providers of health care, customers, network service providers, or the like). Each entity may be associated with a variety of features (or characteristics) which may be structured into columns. In a health care example, the received data may include health data for services claimed to have been rendered to any number of patients by any number of health service providers. The data may be structured such that each row is associated with a different health care provider and each column (i.e., feature) is associated with health care information. Health care information may include any number of billing code(s), cost(s) for procedure(s), tests, medical coding, coverage, drug purchase(s), equipment rental(s), equipment purchase(s), health service rendered, care provided, treatment(s), facility usage, clinical workflow, or the like. For example, each column each column (i.e., feature) may include a different billing code and/or cost for a procedure, test, drug purchase, equipment rental, equipment purchase, health service rendered, care provided, or the like. In this example, at least one column of the data may identify the health care provider by name, identification number, code, or the like. The data may further include a provider's health summary information such as number of visits by the same or different patients, number of similar tests performed, number of days service was provided to one or more patients, number of days one or more patients was admitted (e.g., at a hospital) for care, outcomes, and the like.

[0283] The input module 314 may receive data from any number of sources (e.g., from any number of health service providers and/or insurers). The data may be structured or

unstructured. In one example of unstructured data, the input module 314 or another entity may receive any number of reimbursement requests, invoices, etc., from any number of health care providers (e.g., provided by an insurer, health care system, or the like). The documents or records may include any amount of health care information discussed herein. In some embodiments, the documents or records may include any amount of health summary information.

[0284] The input module 314, the normalization module 1902, or another entity (e.g., a hospital system or insurer) may structure the data for analysis based on the documents, records, and/or any other information. In some embodiments, the input module 314, the normalization module 1902, or another entity may generate new features based on existing features (e.g., by creating a new column based on previously provided information that may be within other columns).

[0285] The normalization module 1902 may normalize data within columns to assist with analysis. In some embodiments, the normalization module 1902 normalizes and/or scales data within one or more columns and creates new columns (e.g., new features) based on the previous normalized and/or scaled data. For example, the normalization module 1902 may create a new column based on previously received utilization features (e.g., contained within the health summary information). Utilization features may include, for example, a number of visits, number of procedures, number of units (e.g., of a drug or equipment) per patient, number of units per visit, and/or the like.

[0286] To normalize, the normalization module 1902 may normalize data within a column. For example, the normalization module 1902 may generate a z score for each entry in the column. The z score may be, for example, the original value for the entry subtracting the mean for the column divided by the standard deviation for the column. In various embodiments, the normalization module 1902 does not change the previous column but rather creates a new column (e.g., a new feature set) with the computed z scores.

[0287] In some embodiments, the normalization module 1902 may scale existing or new columns. In scaling, the normalization module 1902 may scale variables from 0 to 1 for equal contribution to ranking a calculating. In one example, the normalization module 1902 may scale the normalized data of each entry in the new normalized column (e.g., the z scores) by the following formula: $(\text{normalized data entry} - \text{minimum of the normalized column}) / (\text{maximum of the normalized column} - \text{minimum of the normalized column})$. It will be appreciated that

any data entries indicating no value or “0” may be replaced with a nonce of some kind or, for example, 0.5, to avoid an error by dividing for 0 and assist nulls from being outlier(s).

[0288] The input module 314 and/or the normalization module 1902 may select a subset of columns for analysis. For example, the input module 314 and/or the normalization module 1902 may select any number of newly created columns and/or features and not select one or more previously created or received columns and/or features for analysis.

[0289] The fraud or abuse grouping module 1904 may identify fraud and/or abuse in many different ways. In some embodiments, the fraud or abuse grouping module 1904 identifies providers with health care information and/or health summary information that is similar to known cases of fraud and abuse (e.g., utilizing TDA analysis as described herein). In another example, the fraud or abuse grouping module 1904 may identify those providers with health care information and/or health summary information as being extreme or significantly different from other providers (e.g., based on centrality), other providers’ health care information, and/or other providers’ health summary information.

[0290] The following is a discussing regarding identifying providers, some or all of a provider’s health care information, and/or some or all of a provider’s health summary information as being similar to other providers (and their information) previously identified as being associated with fraud and/or abuse. In various embodiments, the input module 314 may receive analysis data (e.g., a first set of health care information which may include health summary information) and known fraud or abuse data (e.g., a second set of health care information associated with providers known or suspected to be associated with fraud, waste, or abuse). Analysis data may include data to be analyzed (e.g., structured or unstructured data based on or including health care information and health summary information). Providers identified in the analysis data may not have been previously investigated and/or it is unknown if the provider is linked to information indicating or suggesting fraud, abuse, or waste. Known fraud or abuse data may be associated with cases of identified as being associated (e.g., as a result for further analysis or investigation) with fraud, abuse, and/or waste.

[0291] The analysis data may include identifiers for a plurality of providers to be analyzed. The analysis data may also include health care information for any number of providers and/or health summary information of any number of providers. Different providers may have different health care information and/or different health summary information.

[0292] It will be appreciated that insurers, hospital systems, enforcement entities, healthcare systems, providers, and the like may analyze or investigate to determine or confirm evidence (e.g., providers and/or health care data) as being associated with fraud and/or abuse. The known data may include health care data and health summary data associated with suspected or confirmed cases of fraud and/or abuse. For example, an investigator or analyst may investigate a provider, the provider's health care information, and/or the provider's health summary information to determine if the provider is involved in fraud, waste, and/or abuse. As a result of the investigation, the investigator or analyst may identify the provider, all or some of the provider's health care information, and/or all or some of the provider's health summary information as being involved in fraud, waste, and/or abuse. The provider, all or some of the provider's health care information, and/or all or some of the provider's health summary information associated with the investigation for fraud, waste, and/or abuse may be identified as known data. Known data may include other providers' health care information and other providers' health summary information that have been previously identified (e.g., as a result of investigation) as being involved in fraud, waste, and/or abuse.

[0293] The fraud or abuse grouping module 1904 may identify those providers (e.g., providers associated with the analysis data) that are similar to known or confirmed providers previously found or suspected to be involved in fraud or abuse (e.g., providers associated with the known fraud or abuse data). In various embodiments, the filter module 316 may receive one or more metric selection(s), one or more lens selection(s), a resolution and a gain as described herein. In this example, the filter module 316 may receive a section for variance normalized Euclidean (VME) with neighborhood lenses. It will be appreciated that the filter module 316 may apply any metric or combination of metrics. Similarly, the filter module 316 may apply any lens or combination of lenses.

[0294] As discussed herein, the analysis module 320 and/or the GUI engine 322 may generate a TDA graph using the analysis data and the known fraud or abuse data to generate a TDA graph with nodes and edges. Each node may have membership of one or more providers (e.g., provider data points). Nodes that are connected share at least one provider as a member.

[0295] The fraud or abuse grouping module 1904 may identify providers from the analysis data (i.e., those providers that are not a part of the known fraud or abuse data) that include similarities (e.g., similar features based on that provider's health care information and/or that provider's health summary information) with providers from the known fraud or abuse data.

In one example, the fraud or abuse grouping module 1904 may identify those providers from the analysis data that are members of a node which also has at least one other member that is a provider from the known fraud or abuse data. In another example, the fraud or abuse grouping module 1904 may identify those providers from the analysis data that are members of a node that also has a predetermined number of members that are providers from the known fraud or abuse data. In a further example, the fraud or abuse grouping module 1904 may identify those providers from the analysis data that are members of a node that also has a number of members that are providers from the known fraud or abuse data that is equal or greater than a predetermined threshold or percentage of node membership (e.g., the number of providers from the known fraud or abuse data is greater or equal to a threshold value when compared to a total number of members of the node).

[0296] In some embodiments, the fraud or abuse grouping module 1904 may also identify providers from the analysis data that are in nodes that are linked (i.e., with an edge) to a node containing at least one provider from the known fraud or abuse data, to a node containing a number of providers from the known fraud or abuse data that is equal to or greater than a predetermined number, or to a node containing a number of providers from the known fraud or abuse data that is greater or equal to a threshold value when compared to a total number of members of the node (e.g., equal to or greater than a percentage).

[0297] In some embodiments, the fraud or abuse grouping module 1904 may receive node or provider selections from a user or another digital device based on the TDA graph. For example, the GUI engine 322 may generate a visualization of the TDA graph for the user or the digital device. The user or digital device may identify one or more providers for a report based on proximity, linking, node membership, or any other information. The providers identified in the report may be identified for further investigation for possible fraud or abuse.

[0298] In one example, a user or a digital device may provide commands to the analysis server 208 to perform an additional function(s) on the features of the providers (e.g., a Euclidean function and/or L1) and color the visualization by the result (based on the result of the function(s)). In this example, the function(s) may be performed for a set of features of a provider. For each feature, the function(s) may be performed relative to that feature for that particular provider relative to the features of the other providers (e.g., all providers or those providers from the analysis data). The user may select any number of providers from the analysis data based on shared-membership of nodes (e.g., being a member of node that has

members of one or more providers from the known data) and/or color of the visualization (e.g., based on the result of the function(s)).

[0299] The reporting module 1906 may, in some embodiments, generate a report listing those providers identified by the fraud or abuse grouping module 1904. The reporting module 1906 may generate the listing based on providers identified by the fraud or abuse grouping module 1904, another user, or another digital device.

[0300] The ranking module 1908 may rank the providers. In one example, the ranking module 1908 ranks (e.g., sorts) each identified provider. The ranking module 1908 may rank a provider based on shared node membership. For example, a provider that shares a node with 10 providers from the known fraud or abuse data may be ranked higher than a provider that shares a node with 2 providers from the known fraud or abuse data. In some embodiments, the ranking module 1908 ranks providers in order of the number of shared node members from the known fraud or abuse data and/or the number of nodes (or membership of nodes) linked to the ranked provider's node (e.g., based on the number of nodes and/or members of nodes that are from the known fraud or abuse data).

[0301] In some embodiments, the ranking module 1908 may identify and rank features of the listed providers that may be outliers based on L1, L Infinity, or the like. As discussed herein, the analysis server 208 to perform an additional function(s) on the features of the providers (e.g., a Euclidean function and/or L1). The function(s) may be performed for a set of features of a provider. For each feature, the function(s) may be performed relative to that feature for that particular provider relative to the features of the other providers (e.g., all providers or those providers from the analysis data). The ranking module 1908 may rank a predetermined number of features for each provider based on the function score(s). In some embodiments, the ranking module 1908 may rank or identify features with function score(s) at or above a predetermined threshold.

[0302] The following is a discussing regarding identifying providers, some or all of a provider's health care information, and/or some or all of a provider's health summary information as being outliers or extreme when compared to other providers. In this example, there is no known fraud or abuse data as described above.

[0303] In various embodiments, metric(s) and/or lens(es) may be utilized in generation of the TDA graph to identify nodes that are separated from others (e.g., extreme or significant

dissimilarities with other providers' health care information and/or other providers' health summary information which may indicate outliers).

[0304] The fraud or abuse grouping module 1904 may identify those providers (e.g., providers associated with the analysis data) that are dissimilar (e.g., outliers or extreme) when compared to other providers (e.g., other providers' features). In various embodiments, the filter module 316 may receive one or more metric selection(s), one or more lens selection(s), a resolution and a gain as described herein. In this example, the filter module 316 may receive selections of Cosine with Gaussian Density and L-infinity Eccentricity. It will be appreciated that the filter module 316 may receive and utilize any number of metric(s) and any number of lens(es).

[0305] The analysis module 320 and/or the GUI engine 322 may generate a TDA graph using the analysis data and the known fraud or abuse data to generate a TDA graph with nodes and edges. Each node may have membership of one or more providers (e.g., provider data points). Nodes that are connected share at least one provider as a member.

[0306] The fraud or abuse grouping module 1904 may further apply functions to the analysis data such as, for example, L1, L Infinity, or the like. As discussed herein, the analysis server 208 to perform an additional function(s) on the features of the providers (e.g., a Euclidean function and/or L1). The function(s) may be performed for a set of features of a provider to generate a provider score (i.e., using the function(s)) for the provider and/or a feature score for any number of features of the provider using the function(s). For each feature, the function(s) may be performed relative to that feature for that particular provider relative to the features of the other providers (e.g., all providers or those providers from the analysis data).

[0307] The fraud or abuse grouping module 1904 may identify the providers with the highest provider scores and/or highest feature score(s). For example, the fraud or abuse grouping module 1904 may identify a predetermined number of providers based on highest provider score and/or highest feature score. In another example, the fraud or abuse grouping module 1904 may identify providers in a provider list with a provider score and/or feature score above a predetermined threshold. It will be appreciated that the fraud or abuse grouping module 1904 may identify any number of providers in the provider list using or based on the provider score(s) and/or the feature score(s) in any number of ways.

[0308] In some embodiments, the fraud or abuse grouping module 1904 may identify nodes with at least one provider from the provider list as a member. The other providers that share membership with the at least one provider from the provider list may be identified by the fraud or abuse grouping module 1904 as possible candidates for the provider list. In some embodiments, the fraud or abuse grouping module 1904 may add the possible candidates to the provider list. In various embodiments, the fraud or abuse grouping module 1904 may added the possible candidates to the provider list based on the provider score and/or the feature score(s) of each possible candidate. For example, if the possible candidate has a provider score and/or feature score(s) above one or more thresholds, the fraud or abuse grouping module 1904 adds the possible candidate to the provider list.

[0309] In some embodiments, the fraud or abuse grouping module 1904 may apply L1 for a feature of a provider against that feature of other providers to generate a score. The fraud or abuse grouping module 1904 may perform this calculation for any number of features of any number of providers. The value of the score (e.g., the largest scores above a predetermined threshold or top largest scores) of any feature or combination of features may indicate provider's that may be further assessed, analyzed, or investigated.

[0310] In some embodiments, the analysis module 320 and/or the GUI engine 322 may identify those providers with the highest L1 extremities score, based on position in the TDA graph which the least density of other nodes, lack of links with other nodes, lack of membership of a node with other providers (e.g., no other members of the nodes or other members being below a threshold), or the like.

[0311] In some embodiments, the fraud or abuse grouping module 1904 may receive node or provider selections from a user or another digital device based on the TDA graph. For example, the GUI engine 322 may generate a visualization of the TDA graph for the user or the digital device. The user or digital device may identify one or more providers for a report based on proximity, linking, node membership, or any other information. The providers identified in the report may be identified for further investigation for possible fraud or abuse.

[0312] The reporting module 1906 may, in some embodiments, generate a report listing those providers identified by the fraud or abuse grouping module 1904 (e.g. from the provider list). The reporting module 1906 may generate the listing based on providers identified by the fraud or abuse grouping module 1904, another user, or another digital device.

[0313] While, as previously discussed, examples of health care fraud or abuse are used herein, it will be appreciated that systems and methods herein may be applied to any industry or field (i.e., not limited to health care). For example, in some embodiments, systems and methods described herein may be applied to determining if entities are utilizing resources or systems without authorization (as discussed with regard to FIG. 25), financial fraud (e.g., by financial entities, money or financial instrument transferors or transferees, money laundering, laboratory mistakes or abuse, or the like).

[0314] FIG. 20 is a flowchart for identifying health care providers to investigate for possible fraud or abuse in some embodiments. In step 2002, the input module 314 receives a first set of health care information from a plurality of providers regarding services and materials provided to a plurality of patients. The health care information may include healthcare information and health summary information as discussed herein. The first set of health care information may be analysis data.

[0315] For example, rows of data may be associated with different providers. Features or columns of the data may include a provider identifier, codes for materials or services provided to patients, costs for materials or services, utilization of equipment or facilities, summary information, and/or the like.

[0316] As discussed herein, the input module 314 and/or normalization module 1902 may normalize any number of the columns and/or scale any number of the columns. The normalized and/or scaled information may replace the original information in the column or, alternately, may be included in a new column added to the data set or may be the basis for new calculations to add new columns to the data set.

[0317] In step 2004, the input module 314 may receive a second set of health care information associated with cases of known fraud and/or abuse. The second set of health care information may also include health summary information. In one example, the second set of health care information may be known data may be associated with cases of known fraud or abuse (e.g., the second set of health care information may be known fraud or abuse data).

[0318] As discussed herein, insurers, hospital systems, enforcement entities, healthcare systems, providers, and the like may investigate, analyze, and confirm evidence (e.g., providers and/or health care data) as being associated with fraud and/or abuse. The known data may include health care data and health summary data associated with cases of fraud and/or abuse.

[0319] In step 2006, the normalization module 1902 may normalize and/or scale features of at least the first set of health care information. In various embodiments, the normalization module 1902 and/or the input module 314 selects one or more features of the data. The normalization module 1902 may normalize the data any number of the selected columns and either replace the data or create one or more new columns with the normalized data. In some examples, the normalization module 1902 may normalize the data using z scores, term frequency-inverse document frequency (TF-IDF), or in any other method.

[0320] In some embodiments, the normalization module 1902 may scale the data in the one or more selected columns as discussed herein. The normalization module 1902 may either replace the data or create one or more new columns with the scaled data.

[0321] Steps 2008-2018 are directed to performing TDA on the data set including the first set of health care information, the second set of health care information, and any additional columns or new information. In various embodiments, the analysis server 208 may perform TDA on selected columns (e.g., a subset of the first set of health care information as well as at least a portion of the second set of health care information).

[0322] In step 2008, the analysis server 208 (e.g., the filter module 316) may receive metric and lens selections from a user or another digital device. The metric selection(s) may include any number of metrics from a set of metrics or pseudo-metrics (e.g., does not satisfy the triangle inequality) from a set of metrics. The lens function may project the data into the reference space as discussed herein. In one example, the metric and lens may be or include Variance Normalized Euclidean (VNE) with Neighborhood Lenses.

[0323] In step 2010, the analysis server 208 (e.g., the resolution module) receives a resolution selection (e.g., resolution parameters) and possibly gain for analysis. The resolution parameters may be received from users or another digital device. In some embodiments, the resolution module 320 may determine one or more possible resolutions based on one or both sets of health care information. The resolution parameters and gain may be utilized to create a covering in the reference space to assist in clustering and/or for node membership.

[0324] In step 2012, the analysis server 208 (e.g., the analysis module 320) performs metric and lens functions on the first and second set of health care information (e.g., or on the selected columns of the first and second set of health care information) to map health care information to a reference space. The process is further discussed herein regarding TDA. As similarly discussed, in step 2014, the analysis server 208 generates a cover in the reference

space using the selected resolution and in step 2016 clusters the mapped health care information. In step 2016, the analysis server 208 clusters the mapped health care information.

[0325] In step 2018, the analysis server 208 (e.g., the processing module 212) generates a topological graph using at least some of the metric-lens(es) combinations from the subset of metric-lens combinations and at least one of the resolutions from subset of resolutions. The topological graph may be displayed (i.e., a visualization) or may be generated in memory.

[0326] FIG. 21 is an example TDA visualization 2100 that may be generated from the steps of FIG. 20 in some embodiments. The nodes of TDA visualization 2100 may include membership of providers from the first and second sets of health care information. It will be appreciated that a subset of the nodes of FIG. 21 will include providers from the second set of health care information (e.g., from the known fraud or abuse data) as members.

[0327] Each provider may be a member of at least one node in the TDA graph generated by the analysis server 208. As a result of the TDA analysis of steps 2008-2018, providers with similar health care information and/or health summary information may be a member of the same node. Nodes that share members (e.g., two or more nodes that share the same provider) will be connected by an edge or line.

[0328] One or more nodes of the of the TDA graph may include members of providers from the second set of health care information. In other words, one or more nodes may include members associated with known or identified cases of fraud and abuse. In some embodiments, the analysis server 208 (e.g., a group identification module) is configured to color or otherwise denote (e.g., based on size of the node) each node of the TDA graph based on whether one or more members of a node are associated with known or identified cases of fraud and abuse. For example, nodes and/or edges may be colored red to indicate the presence of one or more members associated with known or identified cases of fraud and abuse. Other nodes and/or edges without such members may be colored a different color (e.g., green or blue). In some embodiments, nodes and/or edges linked (e.g., by an edge) to nodes with members associated with known or identified cases of fraud and abuse may be colored (e.g., red or pink). In various embodiments, the analysis server 208 may color nodes a deeper color (e.g., deeper red) or another color based on the number or percentage of members of a node that are associated with known or identified cases of fraud or abuse.

[0329] In some embodiments, the analysis server 208 may color nodes of the TDA graph visualization in a manner that reflect an average value in the networks by features of interest,

for example, paid per claim and total paid. A group of nodes may be generated from the network that visually shows the most localization by coloring schemes of interest.

[0330] In step 2020, the analysis server 208 (e.g., fraud or abuse grouping module 1904) identifies the nodes with at least one member associated with the known or identified cases of fraud or abuse. In various embodiments, the fraud or abuse grouping module 1904 identifies nodes that include at least one member (i.e., provider) from the second set of health care information. In one example, the fraud or abuse grouping module 1904 may identify node membership for each provider of the second set of health care information.

[0331] FIG. 22 is an example TDA visualization 2200 that is a portion of the TDA visualization 2100 of FIG. 21. Node 2202 may be a node with at least one member is a provider from the second set of health care information. The other members of node 2202 that are providers from the first set of health care information may be identified by the fraud or abuse grouping module 1904 and/or listed for further investigation. It will be appreciated that members of the node 2202 are grouped through the TDA process because of feature similarities.

[0332] The fraud or abuse grouping module 1904 may also identify all providers from the first set of health care information that are members of the identified nodes. For example, the fraud or abuse grouping module 1904 identifies each node that includes at least one member from the second set of health care information and then identifies all other members from the first set of health care information of those particular nodes as possible providers to further investigate.

[0333] In various embodiments, fraud or abuse grouping module 1904 may identify those nodes linked to a node that contains one or more members from the second set of health care information. It will be appreciated that two or more nodes may be linked if they share one or more of the same providers. The fraud or abuse grouping module 1904 may identify nodes linked to nodes with at least one member from the second set of health care information and may also identify members of nodes of the linked nodes from the first set of health care information.

[0334] For example, the fraud or abuse grouping module 1904 may identify nodes 2202a-e as being linked to node 2202. As previously discussed in this example, node 2202 contains at least one member that is a provider from the second set of health care information. Nodes 2202a-e as well as node 2202 are identified as being enclosed by shape 2204. The fraud or

abuse grouping module 1904 may identify each provider from the first set of health care information in the nodes 2202a-e as possible providers for investigation.

[0335] In step 2022, the reporting module 1906 may generate a report listing one or more of the members from the first set of health care information identified by the fraud or abuse grouping module 1904. Providers identified or listed by the reporting module 1906 may be providers from the first set of health care information that are members of nodes that include providers from the second set of health care information. Other providers identified or listed by the reporting module 1906 may include providers from the first set of health care information that are linked to nodes that include providers from the second set of health care information.

[0336] The list may identify providers from the first set of health care information that merit further review or investigation for possible fraud or abuse. In other words, the listed members from the first set of health care information share membership with the node including members from the second set of health care information because of feature similarities. Since one or more features of the provider is similar to those features of providers associated with known or identified fraud or abuse, further review or investigation may be necessary.

[0337] FIG. 23 is a flowchart of for positioning new health care information of a particular provider in a TDA graph which may be visualized or not visualized in some embodiments. It will be appreciated that after a TDA graph is created based on the first and second set of health care information, additional information may be later received. For example, a provider that previously provided health care information (e.g., a provider identified in the first set of health care information previously received), may provide new health care information and/or health summary information (e.g., from one or more new reimbursement requests, other insurers, other health care systems, or the like). Rather than re-analyzing all of the data, some embodiments described herein enable the information of the provider to be updated relative to the existing TDA graph.

[0338] Further, it will be appreciated that after the TDA graph is created based on the first and second set of health care information, additional information from a new provider (e.g., a provider not previously identified in the first or second sets of health care information) may be later received. Rather than re-analyzing all of the data, some embodiments described herein enable the information of the new provider to be analyzed relative to the existing TDA graph.

[0339] In the following example, new health care information and/or health summary information of a provider previously identified in the first and second set of health care information is received. In step 2302, new health care information and/or health summary information of the previously identified provider (i.e., a particular provider) is received. The new health care information and/or health summary information may be received, for example, by the input module 314 of the analysis server 208.

[0340] In optional step 2304, the input module 314 may combine any amount of the newly received health care information and/or health summary information of the particular provider with any amount of the previously received health care information and/or health summary information of the particular provider. In some embodiments, a user or the input module 314 may select one or more features of the newly received health care information and/or health summary information to combine or replace any number of features of the previously received health care information and/or health summary information to generate updated health care information and/or updated health summary information of the particular provider. In some embodiments, features from the newly received health care information and/or health summary information are added as new features to the previously received health care information and/or health summary information of the particular provider.

[0341] In various embodiments, the normalization module 1902 may normalize and/or scale the newly received health care information and/or health summary information and/or combinations of the newly received and previously received health care information and/or health summary information as discussed herein.

[0342] In an alternative to step 2304, the analysis server 208 may create a new data point for the particular provider as if the particular provider was new (e.g., without replacing or combining previously received health care information and/or health summary information of the particular provider with any of the new information).

[0343] In step 2306, the analysis server 208 (e.g., a new patient distance module) determines distances between the updated information of the particular provider and the health care information and/or health summary information of each previous provider of the first and second sets of health care information. For example, the previous first and second sets of health care information that was utilized in the generation of the TDA graph may be stored in mapped data structures. Distances may be determined between the updated information of the provider and each of the providers of the first and second set of health care information.

[0344] It will be appreciated that distances may be determined in any number of ways using any number of different metrics or functions. Distances may be determined between the updated information of the particular provider with the health care information and/or health summary information of the providers of the first and second sets of information. For example, a distance may be determined between a first feature of the updated information of a particular provider and each (or a subset) of the first feature of the first and second sets of health care information of the other providers.

[0345] In various embodiments, a location of the particular provider on the TDA graph may be determined relative to the other member patients utilizing the determined distances. In some embodiments, prior to analysis or locating the provider on the TDA graph, the analysis server 208 may change the previously created TDA graph by removing the provider as a member of any or all nodes.

[0346] In step 2308, the new patient distance module may compare distances between the previously received health care information of each other provider and new updated information of the particular provider. The particular provider may be located in the grouping of provider members that are closest in distance to the particular provider. In some embodiments, the particular provider location may be determined to be within a grouping that contains the one or more provider members that are closest to the particular provider (even if other members of the grouping have longer distances with the particular provider). In some embodiments, this step is optional.

[0347] In various embodiments, a representative provider member may be determined for each grouping. For example, some or all of the provider members of a grouping may be averaged or otherwise combined to generate a representative provider member of the grouping (e.g., the distances and/or biological data of the patient members may be averaged or aggregated). Distances may be determined between the particular provider health care information and the averaged or combined health care information and/or health summary information of one or more representative provider members of one or more groupings. The location engine may determine the location of the particular provider based on the distances. In some embodiments, once the closest distance between the particular provider and the representative provider member is found, distances may be determined between the particular provider and the individual provider members of the grouping associated with the closest representative provider member.

[0348] In optional step 2310, a diameter of the grouping with the one or more of the provider members that are closest to the particular provider (based on the determined distances) may be determined. In one example, the diameters of the groupings of provider members closest to the particular provider are calculated. The diameter of the grouping may be a distance between two provider members who are the farthest from each other when compared to the distances between all provider members of the grouping. If the distance between the particular provider and the closest provider member of the grouping is less than the diameter of the grouping, the particular provider may be located within the grouping. If the distance between the particular provider and the closest provider member of the grouping is greater than the diameter of the grouping, the particular provider may be outside the grouping (e.g., a new grouping may be displayed on the TDA visualization with the particular provider as the single provider member of the grouping). If the distance between the particular provider and the closest provider member of the grouping is equal to the diameter of the grouping, the particular provider may be placed within or outside the grouping.

[0349] It will be appreciated that the determination of the diameter of the grouping is not required in determining whether the particular provider location is within or outside of a grouping. In various embodiments, a distribution of distances between member providers and between member providers and the particular provider is determined. The decision to locate the particular provider within or outside of the grouping may be based on the distribution. For example, if there is a gap in the distribution of distances, the particular provider may be separated from the grouping (e.g., as a new grouping). In some embodiments, if the gap is greater than a preexisting threshold (e.g., established by a user, digital device, physician, administrator, or previously programmed), the particular provider may be placed in a new grouping that is placed relative to the grouping of the closest member providers. The process of calculating the distribution of distances of candidate member providers to determine whether there may be two or more groupings may be utilized in generation of the TDA graph (e.g., in the process as described with regard to FIG. 12). It will be appreciated that there may be any number of ways to determine whether a particular provider should be included within a grouping of other provider members.

[0350] In step 2312, the location engine determines the location of the particular provider relative to the member providers and/or groupings of the TDA graph. The new location may be relative to the determined distances between the particular provider and the previous

providers. The location of the particular provider may be part of a previously existing grouping or may form a new grouping.

[0351] In some embodiments, the location of the particular provider with regard to the TDA graph may be performed locally by an administrator, analyst, investigator, or the like. For example, the graph may be provided to the analyst (e.g., via digital device). The analyst may load the provider's updated information locally and the distances may be determined locally or via a cloud-based server. The location(s) associated with the particular provider may be overlaid on the previously existing TDA graph either locally or remotely.

[0352] Those skilled in the art will appreciate that, in some embodiments, the previous state of the TDA graph may be retained or otherwise stored and a new TDA graph generated utilizing the updated particular provider's health care information and/or health summary information (e.g., in a method similar to that discussed with regard to FIG. 12). The newly generated map may be compared to the previous state and the differences may be highlighted thereby, in some embodiments, highlighting the location(s) associated with the particular provider. In this way, distances may be not be calculated as described with regard to FIG. 15, but rather, the process may be similar to that as previously discussed.

[0353] As similarly discussed, once a location of the particular provider is determined, the fraud or abuse grouping module 1904 may determine if the particular provider should be investigated. For example, the fraud or abuse grouping module 1904 may identify the particular provider if the particular provider (i.e., based on the updated information) is a member of a node with one or more other members from the second set of health care information. In some embodiments, the fraud or abuse grouping module 1904 may identify the particular provider for further analysis or investigation if the particular provider is a member of a node that is linked to a node with one or more other members from the second set of health care information. In various embodiments, the fraud or abuse grouping module 1904 may identify the particular provider if the provider is a member of a node or linked to a node that has number of members that are members of the second set of health care information (e.g., over a threshold, compared to a percentage, or the like). It will be appreciated that, in some embodiments, the input module 314 may receive an indication of a previous provider associated with the first set of health care information but now is identified as being related to fraud or abuse. The fraud or abuse grouping module 1904 may correct the TDA graph to reflect that the particular provider is now a member of the second set of health care information

and that any nodes of which that particular provider is a member is now flagged as having at least one member of the second set of health care information.

[0354] In some embodiments, previous health care information and/or health summary information may be removed from the first and/or second sets of health care information based on age. For example, information over a year old (or any chronology) may be removed and the TDA graph recalculated. By removing older information, the TDA graph may better depict new or recent changes in provider behavior without being weighted by previous good or bad conduct. Once information is removed, the analysis server 208 may re-analyze the remaining data and regenerate the TDA graph.

[0355] As discussed herein, it will be appreciated that after the TDA graph is created based on the first and second set of health care information, additional information from a new provider (e.g., a provider not previously identified in the first or second sets of health care information) may be later received. The new provider may be associated with new information (e.g., new health care information and/or new health summary information) that has not yet been determined to be associated with fraud or abuse. Alternately, the new provider may include health care information and/or health summary information associated with fraud or abuse (e.g., an investigator identified the provider as well as at least some information associated with the provider as being related to abuse or fraud). If the new provider is associated with fraud or abuse, then the input module 314 may remove any previous information associated with the new provider (e.g., the new provider information identifies or is a provider associated with the second set of health care information) and treat new information regarding the provider as if the provider was new and associated with fraud or abuse.

[0356] FIG. 24 is a flowchart of for positioning new health care information of a new provider in a TDA graph which may be visualized or not visualized in some embodiments. In step 2402, new health care information and/or health summary information of the new provider is received. The new health care information and/or health summary information may be received, for example, by the input module 314 of the analysis server 208.

[0357] In step 2404 the analysis server 208 (e.g., a new patient distance module), may determine distances between the new information of the new provider and the health care information and/or health summary information of each previous provider of the first and second sets of health care information. For example, the previous first and second sets of

health care information that was utilized in the generation of the TDA graph may be stored in mapped data structures. Distances may be determined between the new information of the new provider and each of the providers of the first and second set of health care information.

[0358] It will be appreciated that distances may be determined in any number of ways using any number of different metrics or functions. Distances may be determined between the new information of the new provider with the health care information and/or health summary information of the providers of the first and second sets of information. For example, a distance may be determined between a first feature of the new information of a new provider and each (or a subset) of the first feature of the first and second sets of health care information of the other providers.

[0359] In various embodiments, a location of the new provider on the TDA graph may be determined relative to the other member patients utilizing the determined distances. The new patient distance module may compare distances between the previously received health care information of each other provider and new information of the new provider. As similarly previously discussed, the new provider may be located in the grouping of provider members that are closest in distance to the new provider. In some embodiments, the new provider location may be determined to be within a grouping that contains the one or more provider members that are closest to the new provider (even if other members of the grouping have longer distances with the new provider). In some embodiments, this step is optional. In various embodiments, a representative provider member may be determined for each grouping and used to determine location of the new provider as discussed regarding step 2406.

[0360] In step 2408, a diameter of the grouping with the one or more of the provider members that are closest to the new provider (based on the determined distances) may be determined. In one example, the diameters of the groupings of provider members closest to the new provider are calculated. The diameter of the grouping may be a distance between two provider members who are the farthest from each other when compared to the distances between all provider members of the grouping. If the distance between the new provider and the closest provider member of the grouping is less than the diameter of the grouping, the new provider may be located within the grouping. If the distance between the new provider and the closest provider member of the grouping is greater than the diameter of the grouping, the particular provider may be outside the grouping (e.g., a new grouping may be displayed on the TDA visualization with the particular provider as the single provider member of the grouping).

If the distance between the new provider and the closest provider member of the grouping is equal to the diameter of the grouping, the particular provider may be placed within or outside the grouping.

[0361] It will be appreciated that the determination of the diameter of the grouping is not required in determining whether the new provider location is within or outside of a grouping. In various embodiments, a distribution of distances between member providers and between member providers and the new provider is determined. The decision to locate the new provider within or outside of the grouping may be based on the distribution.

[0362] In step 2410, the location engine may determine the location of the new provider relative to the member providers and/or groupings of the TDA graph. The new location may be relative to the determined distances between the new provider and the previous providers. The location of the new provider may be part of a previously existing grouping or may form a new grouping.

[0363] In some embodiments, the location of the new provider with regard to the TDA graph may be performed locally by an administrator, investigator, or the like. For example, the graph may be provided to the analyst (e.g., via digital device). The analyst may load the new provider's new information locally and the distances may be determined locally or via a cloud-based server. The location(s) associated with the particular provider may be overlaid on the previously existing TDA graph either locally or remotely.

[0364] Those skilled in the art will appreciate that, in some embodiments, the previous state of the TDA graph may be retained or otherwise stored and a new TDA graph generated utilizing the updated provider's health care information and/or health summary information (e.g., in a method similar to that discussed with regard to FIG. 12). The newly generated map may be compared to the previous state and the differences may be highlighted thereby, in some embodiments, highlighting the location(s) associated with the particular provider. In this way, distances may be not be calculated as described with regard to FIG. 15, but rather, the process may be similar to that as previously discussed.

[0365] If the new provider has not been previously associated with fraud or abuse, once a location of the new provider is determined, the fraud or abuse grouping module 1904 may determine if the new provider should be investigated. For example, the fraud or abuse grouping module 1904 may identify the new provider if the new provider is a member of a node with one or more other members from the second set of health care information. In some

embodiments, the fraud or abuse grouping module 1904 may identify the new provider if the particular provider is a member of a node that is linked to a node with one or more other members from the second set of health care information. In various embodiments, the fraud or abuse grouping module 1904 may identify the new provider if the provider is a member of a node or linked to a node that has number of members that are members of the second set of health care information (e.g., over a threshold, compared to a percentage, or the like).

[0366] FIG. 25 is a flowchart for identifying health care providers to investigate for possible fraud or abuse in various embodiments. Fraud, waste, and/or abuse may be detected when behavior of one or more entities is unusual compared with others. In this example, particular entities may be identified as possible candidates for further investigation if their performance information and/or performance summary information is an outlier or an extreme (e.g., dissimilar) to other providers in a TDA graph. As discussed herein, the TDA graph enable for insights to be determined from the shape of data (e.g., without a query to build the TDA graph).

[0367] In this example, performance information and/or performance summary information may indicate utilizing of one or more resources such as, for example, when entities log in, utilize, perform actions, or the like on one or more networks. An administrator may wish to analyze whether the entities are abusing or stealing (e.g., by unauthorized hacking) services using the one or more networks or any resources based on the performance information and/or performance summary information. In this example, a TDA graph may be constructed based on the entities' performance information and/or performance summary information. Entities that are unusual when compared with others (e.g., their performance is unusual when compared to the performance of others), may be identified and listed for further investigation.

[0368] In step 2502, the input module 314 receives a first set of performance information from a plurality of entities regarding actions taken using resources or networks (e.g., of a service provider). For example, rows of data may be associated with different entities (e.g., users, third-party network service providers, customers, organizations, other networks, and the like). Features or columns of the data may include an entity identifier, log-in time, minutes of use, authentication time, indications of successful or unsuccessful authentication, resources utilized, number of networks accessed, databases or applications accessed, number of logins, number of resources accessed, and the like.

[0369] As discussed herein, the input module 314 and/or normalization module 1902 may normalize any number of the columns and/or scale any number of the columns. The normalized and/or scaled information may replace the original information in the column or, alternately, may be included in a new column added to the data set or may be the basis for new calculations to add new columns to the data set.

[0370] In step 2504, the normalization module 1902 may normalize and/or scale features of at least the first set of performance information. In various embodiments, the normalization module 1902 and/or the input module 314 selects one or more features of the data. The normalization module 1902 may normalize the data any number of the selected columns and either replace the data or create one or more new columns with the normalized data. In some examples, the normalization module 1902 may normalize the data using z scores, term frequency-inverse document frequency (TF-IDF), or in any other method.

[0371] In some embodiments, the normalization module 1902 may scale the data in the one or more selected columns as discussed herein. The normalization module 1902 may either replace the data or create one or more new columns with the scaled data.

[0372] Steps 2506-2516 are directed to performing TDA on the data set including the first set of performance information and any additional columns or new information. In various embodiments, the analysis server 208 may perform TDA on selected columns (e.g., a subset of the first set of performance information).

[0373] In step 2506, the analysis server 208 (e.g., the filter module 316) may receive metric and lens selections from a user or another digital device. The metric selection(s) may include any number of metrics from a set of metrics or pseudo-metrics (e.g., does not satisfy the triangle inequality) from a set of metrics. The lens function may project the data into the reference space as discussed herein. In one example, the metric and lens may include Euclidean as a metric and L1-Centrality as a lens. Another lens that may be utilized may be L-Infinity Centrality. It will be appreciated that the filter module 316 may apply any metric or combination of metrics. Similarly, the filter module 316 may apply any lens or combination of lenses.

[0374] In step 2508, the analysis server 208 (e.g., the resolution module) receives a resolution selection (e.g., resolution parameters) and possibly gain for analysis. The resolution parameters may be received from users or another digital device. In some embodiments, the resolution module 320 may determine one or more possible resolutions based on the set of

performance information. The resolution parameters and gain may be utilized to create a covering in the reference space to assist in clustering and/or for node membership.

[0375] In step 2510, the analysis server 208 (e.g., the analysis module 320) performs metric and lens functions on the performance information (e.g., or on the selected columns of the performance information) to map performance information to a reference space. The process is further discussed herein regarding TDA. As similarly discussed, in step 2512, the analysis server 208 generates a cover in the reference space using the selected resolution and in step 2516 clusters the mapped performance information. In step 2514, the analysis server 208 clusters the performance care information.

[0376] In step 2516, the analysis server 208 (e.g., the processing module 212) generates a TDA graph using at least some of the metric-lens(es) combinations from the subset of metric-lens combinations and at least one of the resolutions from subset of resolutions. The topological graph may be displayed (i.e., a visualization) or may be generated in memory.

[0377] In step 2518, the fraud or abuse grouping module 1904 may identify nodes with entities that are outliers or at extremes compared to other entities who are members of nodes in the TDA graph. For example, the fraud or abuse grouping module 1904 may apply L1 for a feature of a particular entity against that feature of other entities to generate a score. The fraud or abuse grouping module 1904 may perform this calculation for any number of features of any number of entities. The value of the score (e.g., the largest scores above a predetermined threshold or top largest scores) of any feature or combination of features may indicate entities that may be further assessed, analyzed, or investigated. The fraud or abuse grouping module 1904 may identify those entities with high entity scores and/or high feature scores as well as those entities that share membership in a node with entities with high entity scores and/or high feature scores.

[0378] In some embodiments, the analysis module 320 and/or the GUI engine 322 may identify those entities with the highest L1 extremities score (e.g., an entity score or feature score of any number of features of the entity), based on position in the TDA graph which the least density of other nodes, lack of links with other nodes, lack of membership of a node with other providers (e.g., no other members of the nodes or other members being below a threshold), or the like. Similarly, the analysis module 320, GUI engine 322, or fraud or abuse grouping module 1904 may identify entities that share node membership with entities with the highest L1 extremities score.

[0379] In another example, using the members (i.e., entities or data points of entities), the entities may be sorted by a measure of centrality (e.g., L1-Centrality) generated on the dataset. The contribution of each feature may be computed to the computed measure of centrality. For instance, a script may calculate a Euclidean distance matrix by each feature, squared, and then sum up the distances to determine the L1-Centrality lens value for each feature.

[0380] A measure of centrality lens values and a subset containing the largest lens values for each feature along with their features names may be written into a comma separated file. As an example, a script may write a file that contains eleven columns for each row, including the L1-Centrality lens value, top five contributing feature names and their values in alternating order.

[0381] For validation of the ranking, the file may be appended to the original source data file. Ranking may be evaluated based on, for example, whether the ranking reflects a good weighting of the features. In the health care example of FIG. 20, a good weighting may be assessed based on knowledge about the healthcare system, medical billing, medical coding, reimbursement, coverage and/or clinical workflow for some examples. In the entities example of FIG. 25, a good weighting may be assessed on knowledge about the resources and network utilized, expected performance, applications, networks, equipment, and/or workflow for some examples.

[0382] If it is not a good weighting, the dataset may need to be re-transformed, uploaded and/or analyzed in networks (e.g., one or more network(s) of nodes). For example, a transformation may re-scale the variables to reflect that some variables are more important. For instance, paid per claim and total paid may have a higher upper bound than other variables. So if all variables are re-scaled to be between zero and one, paid per claim and total paid may be re-scaled to be between zero and two. Sometimes more networks may be re-created to find a better localization or to pull out dense networks regions. Other times other previously created networks may be revisited and colored by other (e.g., combinations of) features.

[0383] If the ranking is considered good enough such that it provides a good description of behavior warranting investigation, the reporting module 1906 may generate a ranked list of entities. The ranked list of entities may be the subject of an investigation or analysis team for confirmation, verification, and/or further investigation. The investigations or analysis team may audit electronic medical records of a provider. Any information as to whether the provider or claim need to be rectified may be recorded and/or provided to an analyst or investigator.

This information may be used to improve the ranking, such as creating new features or adjusting the weights of features that contribute to the ranking.

[0384] In various embodiments, the fraud or abuse grouping module 1904 may utilize L1 Eccentricity or L-infinity Eccentricity to measure a feature of each particular entity against the other entities. The resulting measure may indicate features and/or entities that are outliers or extremes (e.g., based on any number of outlying or extreme features). The reporting module 1906 may list a predetermined number of entities in order of the size of value from the resulting utilization of L1 function. In some embodiments, the reporting module 1906 may further identify those functions that have the greatest value from the resulting utilization of L1 function(s) of those listed providers.

[0385] In step 2520, the reporting module 1906 generates a report with a list of those entities identified by the fraud or abuse grouping module 1904 (e.g., based on the measure of centrality) The entities may be ordered or ranked in order of distance from other entities (e.g., based on significance of outlay or value of L1 function) or relationship with such entities (e.g., node membership with such entities). In some embodiments, the ranking module 1908 may list one or more features (e.g., a predetermined number of features) of identified entities that contribute the most to the entity being an outlier (e.g., by highest entity score or feature score based on, for example, L1 or L infinity score). It will be appreciated that each entity may have a different set of identified features because different features for different entities may contribute to that entity being an outlier.

[0386] FIG. 26 is a sample report generated by the ranking module 1908. In this example, the sample report lists entities by code value (e.g., mpin), the Euclidian distance score that contributed to being listed in report (e.g., generated by the fraud or abuse grouping module 1904), a listing of features and associated values that contributed to that entity being listed. The report may be sorted by Euclidean distance score, by entity code value, by any number of features that most contributed to that particular entity being listed, by value for one or more features, or any other number of ways.

[0387] It will be appreciated that many graphs may be generated for the same data. For example, TDA may be utilized with different filters, metrics, resolutions, or any combination to generate different graphs (and different visualizations) using the same large data set. Each graph may provide different insights based on the data's shape, character, or both. Different filters, metrics, resolutions, or combinations may generate different graphs that include

different groupings of nodes. It may be advantageous for a data scientist to generate at least one graph that includes or depicts groupings of nodes that share a similar characteristic (e.g., outcome). These graphs with this type of groupings (e.g., if the groupings are coherent relative to other nodes in the same graph, relative to other nodes in other graphs, or both) may suggest insights or relationships within the data that were not previously available through query methods.

[0388] In some embodiments, different graphs may be generated on the same data using TDA methods described herein. Each different graph may be generated based on different filters, metrics, resolutions, or any combination (e.g., for fraud or abuse detection or generating a list of entities for further analysis or investigation). Even though a user or analyst may provide a selection of one or more metrics and lenses, some embodiments described herein may perform TDA on different combinations of metrics and lenses, different metrics, different lenses, different resolutions, and different gains to identify a preferred metric(s), lens(es), resolution, and/or gain. In some embodiments, a user or analyst does not provide any selection of metrics and lenses. In that example, various embodiments may perform TDA on different combinations of metrics and lenses, different metrics, different lenses, different resolutions, and different gains to identify a preferred metric(s), lens(es), resolution, and/or gain.

[0389] All or some nodes within each graph may be grouped based on a shared characteristic such as outcome. For example, if the outcome is survival, then nodes may be grouped based on those that survived and those that did not survive. In some embodiments, nodes are only grouped if they share the same outcome (e.g., they share the same shared characteristic), there is a sufficient number of nodes in the group, there is a sufficient number of connections between nodes between groups, or any combination. Each graph may be scored based on the groupings within that particular group. Each graph may then be ordered based on score and one or more of the graphs may be identified or provided to a user based on the ordered score (e.g., visualizations of the top three graphs based on best score is displayed or otherwise provided to the user).

[0390] It will be appreciated that this system of automatic outcome analysis may allow a data scientist to have data assessed through a variety of different filters, metrics, resolutions, or any combination to find a preferred graph without manual selection of each filter, metric, and resolution. Further, the data scientist may not be required to manually inspect each

visualization of each graph to identify one or more graphs that have the best grouping of nodes based on outcome.

[0391] In various embodiments, systems and methods described herein may assist in identifying one or more graphs that best localizes an outcome or shared characteristic (e.g., of one or more columns in the data set). For example, a graph that best localizes an outcome may include or depict regions of the graph that have categories of or similar values from the outcome concentrated together). In some embodiments, systems and methods described herein may generate scores to identify metrics that are “continuous” with respect to the outcome column, identify lenses that have distributions of the outcome categories that are “localized,” and choose lens parameters that are localized and separate outcome categories (or similar outcome values) without having too many singletons.

[0392] FIG. 27 depicts a visualization 2700 of a graph that illustrates outcomes that are not significantly localized. The visualization 2700 is colored in greyscale based on outcome. The nodes of the visualization 2700 appear generally connected and black nodes appear to collect on three sides of the visualization 2700.

[0393] FIG. 28 depicts a visualization 2800 of a graph that illustrates outcomes that are more localized than FIG. 27. Figures 27 and 28 may depict different visualizations of the same data (e.g., TDA, as discussed herein, is performed on the same data but different filters, metrics, resolutions, or combinations were used to generate different graphs and visualizations) Like FIG. 27, the visualization 2800 is colored in greyscale based on outcome. It will be appreciated that nodes that share similar outcome and are similarly colored are more densely packed and more interconnected than those depicted in FIG. 27. A data scientist may perceive additional insights, identify interrelationships in the data, or both using FIG. 28 rather than FIG. 27.

[0394] FIG. 29 is a block diagram of an exemplary analysis server 208 including an outcome analysis module 2902. The exemplary analysis server 208 depicted in FIG. 20 may be similar to the exemplary analysis server 208 depicted in FIGS. 2, 3, and 19. In exemplary embodiments, the analysis server 208 comprises a processor 302, input/output (I/O) interface 304, a communication network interface 306, a memory system 308, and a storage system 310.

[0395] The storage system 310 comprises a plurality of modules utilized by embodiments of the present invention. A module may be hardware (e.g., an ASIC), software (e.g., including instructions executable by a processor), or a combination of both. In one embodiment, the

storage system 310 comprises a processing module 312 which comprises an input module 314, a filter module 316, a resolution module 318, an analysis module 320, a visualization engine 322, a database storage 324, the normalization module 1902, the fraud or abuse grouping module 1904, the reporting module 1908, and the ranking module 1908. Alternative embodiments of the analysis server 208 and/or the storage system 310 may comprise more, less, or functionally equivalent components and modules.

[0396] The input module 314 may be configured to receive commands and preferences from the user device 202a. In various examples, the input module 314 receives selections from the user which will be used to perform the analysis. The output of the analysis may be an interactive visualization.

[0397] The input module 314 may provide the user a variety of interface windows allowing the user to select and access a database, choose fields associated with the database, choose a metric, choose one or more filters, and identify resolution parameters for the analysis. In one example, the input module 314 receives a database identifier and accesses a large multi-dimensional database. The input module 314 may scan the database and provide the user with an interface window allowing the user to identify an ID field. An ID field is an identifier for each data point. In one example, the identifier is unique. The same column name may be present in the table from which filters are selected. After the ID field is selected, the input module 314 may then provide the user with another interface window to allow the user to choose one or more data fields from a table of the database.

[0398] Although interactive windows may be described herein, it will be appreciated that any window, graphical user interface, and/or command line may be used to receive or prompt a user or user device 202a for information.

[0399] The filter module 316 may subsequently provide the user with an interface window to allow the user to select a metric to be used in analysis of the data within the chosen data fields. The filter module 316 may also allow the user to select and/or define one or more filters.

[0400] The resolution module 318 may allow the user to select a resolution, including filter parameters. In one example, the user enters a number of intervals and a percentage overlap for a filter.

[0401] The analysis module 320 may perform data analysis based on the database and the information provided by the user. In various embodiments, the analysis module 320 performs

an algebraic topological analysis to identify structures and relationships within data and clusters of data. It will be appreciated that the analysis module 320 may use parallel algorithms or use generalizations of various statistical techniques (e.g., generalizing the bootstrap to zig-zag methods) to increase the size of data sets that can be processed. The analysis is further discussed in FIG. 8. It will be appreciated that the analysis module 320 is not limited to algebraic topological analysis but may perform any analysis.

[0402] The visualization engine 322 generates an interactive visualization including the output from the analysis module 320. The interactive visualization allows the user to see all or part of the analysis graphically. The interactive visualization also allows the user to interact with the visualization. For example, the user may select portions of a graph from within the visualization to see and/or interact with the underlying data and/or underlying analysis. The user may then change the parameters of the analysis (e.g., change the metric, filter(s), or resolution(s)) which allows the user to visually identify relationships in the data that may be otherwise undetectable using prior means. The interactive visualization is further described in FIGS. 9-11.

[0403] The database storage 324 is configured to store all or part of the database that is being accessed. In some embodiments, the database storage 324 may store saved portions of the database. Further, the database storage 324 may be used to store user preferences, parameters, and analysis output thereby allowing the user to perform many different functions on the database without losing previous work.

[0404] The normalization module 1902, fraud or abuse grouping module 1904, reporting module 1908, and ranking module 1908 are further discussed with regard to FIG. 19.

[0405] The outcome analysis module 2902 is configured to select metric-lens combinations, select resolutions, generate graphs using the selected metric-lens(es) combinations and resolutions, group nodes by distribution of outcomes of member data points to score each graph, and select one or more graphs based on score (e.g., based on similar or same outcomes or distribution of outcomes). The outcome analysis module 2902 may generate visualizations of the selected graphs. The outcome analysis module 2902 is further described with regard to FIG. 30.

[0406] It will be appreciated that that all or part of the processing module 212 may be at the user device 202a or the database storage server 206. In some embodiments, all or some of the functionality of the processing module 312 may be performed by the user device 202a.

[0407] FIG. 30 depicts an example outcome analysis module 3002 in some embodiments. The outcome analysis module 3002 may comprise a metric selection module 3002, a metric-lens combination selection module 3004, a lens parameter selection module 3006, group identification module 3008, group score module 3010, a graph score module 3012, a graph selection module 3014, and an outcome visualization module 3016. The metric selection module 3002 may select one or more metrics for testing from a set of metrics. In some embodiments, a user may provide the metric selection module 3002 with one or more metrics to test. The metric selection module 3002 may test a metric in a number of ways. In one example, the metric selection module 3002 may receive a data set and, for each data point in the data set, determine one or more closest neighboring data points using the metric to be tested. For each point in the data set, the metric selection module 3002 may improve (e.g., increase) a metric score if the one or more closest neighboring data points to that data point share a similar or same shared characteristic(s) (e.g., similar or shared outcome). In this way, the metric selection module 3002 may generate a metric score for the graph. The metric selection module 3002 may similarly generate a metric score for each metric using the same data set. The metric selection module 3002 may select one or more metrics and generate a subset of metrics including the one or more selected metrics using the metric scores (e.g., based on highest data scores). The process is further described herein.

[0408] The metric-lens combination selection module 3004 may select one or more lens(es) to combine with one or more metric(s) of the subset of metrics. In various embodiments, the metric-lens combination selection module 3004 may select lenses from a set of lenses or receive one or more lenses to test from a user. For each metric-lens(es) combination, the metric-lens combination selection module 3004 may compute a reference map of the lens space using the data from the originally received data set and compute a category entropy of each subspace of each reference map. The metric-lens combination selection module 3004 may compute a metric-lens score based on a sum of category entropies of a particular graph. The metric-lens combination selection module 3004 may select one or more metric-lens(es) combinations based on the metric-lens score. The process is further described herein.

[0409] The lens parameter selection module 3006 may select lens parameters (e.g., resolution, gain, or both). The choice of resolution may depend on the number of points in the space and the number of lenses. The process is further described with regard to FIG. 38.

[0410] The group identification module 3008 identifies groups within a graph (e.g., a topological graph) generated using at least one of the subset of metric-lens combination and at least one selected resolution. The group identification module 3008 may identify one or more groups that share the same or similar outcomes (or distribution of outcomes).

[0411] The group score module 3010 may generate a group score for each group identified by the group identification module 3008. The group score may be based in part on entropy of the group. The graph score module 3012 generates a graph score based on the entropy of each group within that graph. A graph selection module 3014 may select one or more graphs to provide or identify for the user based, at least in part, on the graph score. The visualization module 3016 may generate one or more visualizations of graphs selected by the graph selection module 3014. These processes are further described herein.

[0412] FIG. 31 is a flowchart for outcome auto analysis in some embodiments. In step 3102, the input module 314 may receive data including or from a large data set. Systems and methods described herein may be utilized in big data analysis. Big data is a term that refers to data sets that are large and/or complex such that traditional data processing of the prior art may be inadequate or limited. Massive data sets may include hundreds of thousands, millions, or even billions (or more) data points and/or any number of characteristics per data point. There may be significant hardware, service, and/or financial limitations that must be considered when attempting to analyze large data sets (e.g., up to an including massive data sets as described above). In some embodiments, the data, data set, or both may be large (e.g., massive) data sets.

[0413] In various embodiments, the data received in step 3102 may include one or more shared characteristics in columns related to each row or data point. In one example, the one or more shared characteristics may include one or more outcomes. The outcomes may, in some embodiments, be categorical. If the data is medical data, the shared characteristics may include, for example, patient outcomes after treatment(s), services, or the like. Shared characteristics, such as outcomes, may be continuous or discrete. A discrete outcome may individually separate and distinct (e.g., the data may be divided in part on those who survived and those who died). Alternately, the outcomes may be measured as part of a continuum. In some embodiments, the metric selection module 3002 may perform discretization of continuous features of outcomes from one or more outcome columns of the data to generate outcome categories that are discrete.

[0414] In various embodiments, the method described herein may search through existing metrics and lenses to produce one or more graphs that separate the outcome groups. The separations may not necessarily be based on geometry. One existing example method includes steps for choosing or selecting one or more metrics, choose or select one or more lenses, and choose or select one or more scale parameters. Those metrics that are chosen or selected may make the outcome (e.g., shared characteristic(s)) column(s) “continuous” with respect to the metric. Those lenses (e.g., metric-lens(es) combination(s)) that are chosen or selected may assist in localizing outcome values. Those scale parameters that are chosen or selected may both localize and separate different outcome values without making too many singletons (e.g., nodes of a graph of visualization with a single data point as a member or few data points as a member). These processes are further described below.

[0415] The data may be related to any science, profession, service, or the like (e.g. the data may be medical data, financial data, scientific data, energy data, or the like).

[0416] In step 3104, the metric selection module 3002 selects a subset of metrics. The subset may include any number of metrics from a set of metrics or pseudo-metrics (e.g., does not satisfy the triangle inequality) from a set of metrics. Metrics that are selected may be used with received data to generate “continuous” results with respect to the shared characteristics (e.g., outcome column(s)). For example, points that are similar in the metric may have similar or the same outcome characteristics (e.g., outcome categories).

[0417] In various embodiments, the metric selection module 3002 builds neighborhoods for each point. For each point, the metric selection module 3002 may count the number of points whose nearest neighbor has the same shared characteristic category as the particular point to generate a metric score for each metric. Different metric scores may be compared to select those metrics that are “continuous” with respect to the shared characteristics. The process is further described with regard to FIG. 32.

[0418] In step 3106, the metric-lens combination selection module 3002 selects a subset of metric-lens(es) combinations based in part on the subset of metrics. In various embodiments, metric-lens(es) combinations may be selected to test (e.g., any number of metric-lens(es) combinations may be tested). The metric-lens combination selection module 3002 may select the metric-lens(es) combinations that localize outcome values (e.g., relative to other metric-lens(es) combinations). The process is further described with regard to FIG. 35.

[0419] In step 3108, the lens parameter selection module 3006 may select a resolution. In various embodiments, the lens parameter selection module 3006 chooses resolution and gain. The choice of resolution may depend on the number of points in the space and the number of lenses. The process is further described with regard to FIG. 38.

[0420] In step 3110, the processing module 212 generates a topological graph using at least some of the metric-lens(es) combinations from the subset of metric-lens combinations and at least one of the resolutions from subset of resolutions. The process is further described with regard to FIG. 38.

[0421] In step 3112, the group identification module 3008 characterizes nodes of each group based on outcome(s) (e.g., shared characteristic(s)). In various embodiments, the group identification module 3008 colors or otherwise denotes each node of the topological graph based on outcome(s) of member points of that node or distribution of outcome(s) of member points of that node. The process is further described with regard to FIG. 38.

[0422] In step 3114, the group identification module 3008 may identify groups of nodes that share the same or similar outcome of each graph. This process may utilize autogrouping described herein to determine edge weight between nodes, identify a partition that includes groupings of nodes based on distribution of outcome(s) of data points, or both. Autogrouping is described in U.S. Nonprovisional application No. 15/166,207 entitled “Outcome Analysis for Graph Generation” which is incorporated by reference herein.

[0423] In step 3116, the group score module 3010 scores each graph based on identified groups. In various embodiments, the group score module 3010 may score each group in a graph based on entropy of groups, number of data points, number of nodes in a group, number of outcome categories (e.g., discrete outcome groups of the outcomes identified using the originally received data), or any combination. The graph score module 3012 may generate a graph score for each graph based on the group scores of that graph. These processes are further described with regard to FIG. 38.

[0424] In step 3118, the graph selection module 3014 may select one or more graphs based on the graph scores. The graph selection module 3014 may select a graph in any number of ways. For example, the graph selection module 3014 may select one or more graphs based on the highest scores, may select the top graph based on score, may select all graphs with graph scores above a predetermined threshold, may select a predetermined number of graphs in order

of the highest score, or may not select any graph below a predetermined threshold. The process is further described with regard to FIG. 38.

[0425] In step 3120, the graph visualization module 3016 generates visualizations for each selected graph and provides the visualizations to a user. The graph visualization module 3016 may provide the visualizations in any order including, for example, beginning with a visualization of the graph with the highest graph score. In some embodiments, for each graph, the graph visualization module 3016 displays information related to generating the particular graph, including for example metric information indicating the metric used, metric-lens information indicating the lens(es) used, resolution information indicating the resolution and gain used, graph score, or any combination. The process is further described with regard to FIG. 38.

[0426] FIG. 32 is a flowchart for selection of a subset of metrics in some embodiments. In various embodiments, the metric selection module 3002 may receive or retrieve a set of metrics. The metrics may be of any type. Example metrics include, but are not limited to Variance Normalized Euclidean (“VNE”), Euclidean, Manhattan, Chebyshev, Interquartile Range Normalized Euclidean, Angle, Cosine, Pearson Correlation, Absolute Correlation, or the like. The set of metrics may be a subset of another set of metrics. For example, metrics may be initially chosen based on the type of data received from the original data set. Metrics may be initially chosen in any number of ways or alternately, a set of metrics may be tested as described with regard to FIG. 32 regardless of the underlying data.

[0427] In step 3202, for each metric, the metric selection module 3002 uses that particular metric to identify one or more of the closest points for the points in the data set. The metric selection module 3002 may not build a graph. For example, assume metrics VNE, Euclidean, and Manhattan are to be tested using the method described regarding FIG. 32. For each point in the data set, the metric selection module 3002 may identify one or more closest points for each metric. For example, for each point in the data set, the metric selection module 3002 may identify the closest points within the data set using the first metric (e.g., VNE) to create a first metric dataset. The process may be repeated for the other metrics. For example, for each point in the data set, the metric selection module 3002 may identify the closest points within the data set using the second metric (e.g., Euclidean) to create a second metric dataset. As follows, for each point in the data set, the metric selection module 3002 may identify the

closest points within the data set using the third (e.g., Manhattan) to create a third metric dataset.

[0428] In step 3204, for each point in each graph, the metric selection module 3002 calculates a point score for that graph based on the nearest neighbor to that point which has the same or similar outcome (e.g., same or similar shared characteristic) to that particular point. In some embodiments, for each point in the first graph, the metric selection module 3002 may find the nearest point. If the nearest point shares the same or similar outcome as the particular point, then the metric selection module 3002 may add a value to a point score or otherwise change the value of a point score. If the nearest point does not share the same or similar outcome to the particular point, the metric selection module 3002 may not add a value to a point score, may not change the value of the point score, or change the value of a point score to indicate the dissimilarity in the point score for the graph.

[0429] For example, after the metric selection module 3002 generates a graph using data and a first metric, the metric selection module 3002 may select a particular point in the graph. The metric selection module 3002 may then identify the closest point to the particular point in the graph and determine if the two points have the same or similar outcome. If they have the same outcome, the metric selection module 3002 may change the value of the point score for the graph. If they do not have the same outcome, the metric selection module 3002 may not change the value of the point score. The metric selection module 3002 will repeat this process for each point in the graph (e.g., determining that point's nearest neighboring point, determining if the two points have the same or similar outcome, and changing the value of the point score if they have similar values).

[0430] The metric selection module 3002 may repeat this process for each graph. It will be appreciated that each graph may be associated with a point score based on the point scores of points and neighboring points in that particular graph. It will be appreciated that the metric selection module 3002 may generate metric scores using nearest neighboring points and shared outcomes in any number of ways.

[0431] In some embodiments, for each point of a graph, the metric selection module 3002 may identify any number of nearest neighboring points. The metric selection module 3002 may generate or change the metric score based on the outcomes associated with the nearest neighboring points (e.g., an average outcome of the nearest neighboring points, a majority of the nearest neighboring points, or the like).

[0432] In step 3206, the metric selection module 3002 may calculate metric scores for each graph using the point scores. In some embodiments, the metric scores are calculated when a point score is changed, a point score is added to the metric score, or a point score is subtracted from the metric score in step 3204. The point score for any number of points may change the metric score in any number of ways. The point score may be the metric score.

[0433] In step 3208, the metric selection module 3002 may optionally rank or compare each graph for each metric using the metric scores. For example, each graph may be associated with a different metric score and the graphs may be ordered or ranked from highest metric score to lowest metric score.

[0434] In step 3210, the metric selection module 3002 selects one or more metrics using the metric scores. For example, the metric selection module 3002 may identify the graph associated with a metric as having the highest metric score and the metric selection module 3002 may select the associated metric as a member of the selected subset of metrics. The metric selection module 3002 may select a percentage of the metric scores indicating consistent more outcomes of neighboring points (e.g., the metric selection module 3002 may select metrics based on the top 20% of metric scores) or a predetermined number of metrics using the metric scores (e.g., the metric selection module 3002 selects four metrics associated with the top four graphs based on highest metric scores). It will be appreciated that the metric selection module 3002 may select metrics using the metric scores in any number of ways.

[0435] FIGS. 33A-33C depict example graphs including the same data points from Gaussian data and different metrics. Although only metrics for VNE, Manhattan, and Absolute Correlation are shown, it will be appreciated that any number of different metrics may be used to generate graphs to test the metrics. FIG. 33A depicts an example graph using a VNE metric. Each data point may be colored or otherwise associated with an outcome (e.g., from an outcome column of the data). FIG. 33A depicts groupings of data points with fairly consistent outcomes. This may represent a desirable graph. Metric score 3302 is high compared to the metric score 3306 of FIG. 33C. The metric score 3302 may be based on point scores for each point in the graph (e.g., each point in the graph contributing a point score to the metric score 3302 if the nearest neighbor to that point shares the same or similar outcome).

[0436] FIG. 33B depicts an example graph using a Manhattan metric. Like FIG. 33A, FIG. 33B depicts groupings of data points with fairly consistent outcomes. This may also represent

a desirable graph. Metric score 3304 is the same as metric score 3302 of FIG. 33A but is also high compared to the metric score 3306 of FIG. 33C.

[0437] FIG. 33C depicts an example graph using an Absolute Correlation metric. FIG. 33C depicts one large group with different outcomes that are intermixed. The metric score 3306 is low compared to metric scores 3302 and 3304.

[0438] In some embodiments, based on the graphs FIG. 33A-C, the metric scores 3302, 3304, and 3306, or both, the metric selection module 3002 may select metrics VNE and Manhattan for the subset of metrics. The metric selection module 3002 may not select the Absolute Correlation for the subset of metrics because the points, colored or otherwise identified by outcome, are intermixed (e.g., relative to other graphs).

[0439] FIGS. 34A-34C depict example graphs including the same data points from MNIST data and different metrics. FIG. 34A depicts an example graph using an Angle metric. Each data point may be colored or otherwise associated with an outcome (e.g., from an outcome column of the data). FIG. 34A depicts groupings of data points with fairly consistent outcomes although there are some relatively minor intermixed data. This may represent a desirable graph. Metric score 3402 is high compared to the metric score 3406 of FIG. 34C and metric score 3408 of FIG. 34D. The metric score 3402 may be based on point scores for each point in the graph (e.g., each point in the graph contributing a point score to the metric score 3402 if the nearest neighbor to that point shares the same or similar outcome).

[0440] FIG. 34B depicts an example graph using a Euclidean metric. Like FIG. 34A, FIG. 34B depicts groupings of data points with fairly consistent outcomes with some intermixing. This may also represent a desirable graph. Metric score 3404 is similar albeit a little lower than metric score 3402 of FIG. 34A but is also high compared to the metric score 3406 of FIG. 34C and metric score 3408 of FIG. 34D.

[0441] FIG. 34C depicts an example graph using a Norm. Correlation metric. FIG. 34C depicts a generally large shape but with somewhat consistent groupings of nodes that share outcomes. There are more intermixed nodes with different outcomes when compared to FIGS. 34A and 34B, however the metric score 3406 is still reasonably close to metric score 3402 of FIG. 34A and metric score 3404 of FIG. 34B. The metric score 3406 is high compared to metric score 3408 of FIG. 34D.

[0442] FIG. 34D depicts an example graph using a Chebyshev metric. FIG. 34D depicts a fairly large group with different outcomes that are more intermixed. The metric score 3408 is low compared to metric scores 3302 of FIG. 34A, metric score 3304 of FIG. 34B, and metric score 3306 of FIG. 34C.

[0443] In some embodiments, based on the graphs depicted in FIGS. 34A-34D, the metric scores 3402, 3404, 3406, and 3408, or both, the metric selection module 3002 may select metrics Angle, Euclidean, and potentially Norm. Correlation. In some embodiments, the metric selection module 3002 may be configured to only select the top 2 metrics or any limited number of metrics. In that case, the metric selection module 3002 may only select the Angle metric and the Euclidean metric. The metric selection module 3002 may not select the Chebyshev metric for the subset of metrics because the points, colored or otherwise identified by outcome, are intermixed (e.g., relative to other graphs).

[0444] FIG. 35 is a flowchart of selection of a subset of metric-lens combinations in some embodiments. In various embodiments, the metric-lens combination selection module 3004 identifies one or more potential lenses for each metric of the subset of metrics for testing. The metric-lens combination selection module 3004 may also test and score each metric in combination with one or more lenses to evaluate a metric-lens combination. It will be appreciated that the metric-lens combination selection module 3004 may select any different combination of metric and lens(es). For example, the metric-lens combination selection module 3004 may identify the same metric but with different lens combinations for testing.

[0445] In step 3502, for each metric of the subset of metrics, the metric-lens combination selection module 3004 identifies one or more lenses to evaluate in association with the metric. In various embodiments, the metric-lens combination selection module 3004 may identify any number of lenses that may be associated with a metric of the subset of metrics. It will be appreciated that some metric-lens combinations may not function or otherwise may not be desirable without further assessment. The metric-lens combination selection module 3004 may filter known nonfunctional or undesirable lenses before selecting metric-lens combinations. In some embodiments, the metric-lens combination selection module 3004 may combine known functional metric-lens combinations or potentially desirable metric-lens(es) combination for further evaluation. The metric-lens combination selection module 3004 may generate a set of metric-lens combinations.

[0446] In various embodiments, the metric-lens combination selection module 3004 may receive one or more lenses to combine with a metric and include in the subset of metric-lens combinations. For example, a user may input a preferred or desired lens to test with a metric of the subset of metrics. The metric-lens combination selection module 3004 may evaluate the provided lens in conjunction with one or more metrics or, alternately, the metric-lens combination selection module 3004 may include the provided lens in the subset of metric-lens combination.

[0447] In step 3504, for each metric-lens(es) combination, the metric-lens combination selection module 3004 computes a reference map of the lens space using data from the originally received data set (e.g., all or part of the data received in step 3102 of FIG. 31). The reference map is generated using both the metric and lens(es) of the metric-lens(es) combination. In various embodiments, the metric-lens combination selection module 3004 generates a separate reference map for each metric-lens(es) combination.

[0448] In step 3506, the metric-lens combination selection module 3004 computes category entropy for each subspace of each graph (e.g., each reference map). Category entropy may be, for example, entropy of outcome(s) (e.g., entropy of shared characteristic(s)). In various embodiments, the metric-lens combination selection module 3004 may divide a graph into subspaces. For example, the metric-lens combination selection module 3004 may divide the graph into subspaces of equal or variable size with each subspace being adjacent to other subspaces. In some embodiments, the metric-lens combination selection module 3004 may divide groupings of data points into separate subspaces. The metric-lens combination selection module 3004 may compute entropy for one or more subspaces based on shared characteristic(s) such as outcome(s) of the data points in the subspace(s).

[0449] In step 3508, the metric-lens combination selection module 3004 computes a metric-lens score of the metric-lens(es) combination by adding category entropy across all subspaces of a graph (e.g., a reference map). The metric-lens combination selection module 3004 may also compute the metric-lens score based, at least in part, on the number of points within each subspace. The metric-lens combination selection module 3004 may generate a metric-lens score for each reference map generated in step 3504. For example, for each graph, the metric-lens combination selection module 3004 may generate a metric-lens score based on the category entropy across any number of subspaces of that particular graph and based on the number of points within any number of subspaces. The metric-lens combination selection

module 3004 may optionally rank (e.g., order) the graphs based on metric lens scores (e.g., from highest to lowest or in any other method).

[0450] In step 3510, the metric-lens combination selection module 3004 selects a subset of metric-lens(es) combinations based on the metric lens score. In various embodiments, the metric-lens combination selection module 3004 may be configured to choose a predetermined number of metric-lens(es) combinations, choose metric-lens(es) combinations with a metric-lens score that meet a certain criteria (e.g., choose metric-lens(es) combinations with a metric-lens score over a predetermined threshold), choose metric-lens(es) that are in a top predetermined percentage, or any other method.

[0451] FIGS. 36A-36C depict example graphs including the same data points from Gaussian data and different metric-lens(es) combination. Although metrics for Euclidean (L2), Angle, and Manhattan (L1) are shown, it will be appreciated that any number of different metric-lens(es) combinations may be used to generate graphs to test the metric-lens(es) combinations. For example, Euclidean (L2) and neighborhood lens (e.g., neighborhood lens 1, neighborhood lens 2, or both) may be used. In another example, Chevyshev (L-Infinity) and neighborhood lens 1, neighborhood lens 2, multidimensional scaling (MDS) coordinates 1, MDS coordinates 2, metric Principal Component Analysis (PCA) coordinates 1, metric PCA coordinates 2, or any combination may be used. In a further example, variance normalized Euclidean metric and neighborhood lens (e.g., neighborhood lens 1, neighborhood lens 2, or both) may be used. Other examples include a cosine metric with PCA coord. 1, PCA coord. 2, L1 Centrality, Gaussian density, MDS coord. 1, MDS coord. 2, or any combination. Any other metric, lens(es), or combination may be used.

[0452] FIG. 36A depicts an example graph using a Euclidean (L2) metric with a neighborhood lens. Each data point may be colored or otherwise associated with an outcome (e.g., from an outcome column of the data). FIG. 36A depicts groupings of data points with fairly consistent outcomes with limited entropy. This may represent a desirable graph with a higher metric-lens score when compared to metric-lens scores associated with graphs of FIGS. 36B and 36C.

[0453] FIG. 36B depicts an example graph using an angle metric and MDS Coord. 2 lens. FIG. 36B has greater entropy for the graph (e.g., when entropy is added across subspaces of the graph), the entropy for the graph in FIG. 36B may be greater than the entropy for the graph of FIG. 36A.

[0454] FIG. 36C depicts an example graph using a Euclidean (L1) using a neighborhood lens. FIG. 36C depicts a line with that may have lower entropy when compared to the entropy of FIG. 36B.

[0455] In some embodiments, based on the graphs FIGS. 36A-36C, the metric-lens scores of the graphs, or both, the metric-lens combination selection module 3004 may select the Euclidean (L2) and neighborhood lens for the subset of metric-lens(es). The metric-lens combination selection module 3004 may select the Euclidean (L1) and neighborhood lens for the subset of metric-lens(es) as well. The metric-lens combination selection module 3004 may not select the Angle metric with the MDS Coord. 2 lens based on a comparison of the entropy of the graph to entropy of graphs depicted in FIG. 36A and 36B.

[0456] FIGS. 37A-37C depict example graphs including the same data points from MNIST data and different metric-lens(es) combinations. FIG. 37A depicts an example graph using a Cosine metric and neighborhood lens. Each data point may be colored or otherwise associated with an outcome (e.g., from an outcome column of the data). FIG. 37A depicts groupings of data points with some mixed outcomes. The entropy associated with the graph in FIG. 37A, and therefore the metric-lens score associated with the graph, may be lower than the entropy and/or metric-lens scores of the graphs depicted in FIGS. 37B and 37C.

[0457] FIG. 37B depicts an example graph using a Euclidean metric and neighborhood lens. FIG. 37B is similar to that in FIG. 37A and may have similar or greater entropy for the graph (e.g., when entropy is added across subspaces of the graph) than the entropy for the graph of FIG. 37A.

[0458] FIG. 37C depicts an example graph using a Cosine using a PCA lens. FIG. 37C depicts a mixture of outcomes and a higher entropy when compared to the entropy of FIGS. 37A and 37B.

[0459] In some embodiments, based on the graphs FIG. 37A-C, the metric-lens scores of the graphs, or both, the metric-lens combination selection module 3004 may select the Cosine and neighborhood lens for the subset of metric-lens(es). The metric-lens combination selection module 3004 may select the Euclidean (L1) and neighborhood lens for the subset of metric-lens(es) as well. The metric-lens combination selection module 3004 may not select the Cosine with the PCA lens based on a comparison of the entropy of the graph to entropy of graphs depicted in FIG. 37A and 37B.

[0460] FIG. 38 is a flowchart for identifying one or more graphs based on a graph score using the metric-lens combinations of the subset of metric-lens combinations. In step 3802, for each metric-lens(es) combination of the subset of metric-lens combinations, the lens parameter selection module 3006 or the processing module 312 constructs a TDA graph. The graph may not be visualized in some embodiments. In various embodiments, the lens parameter selection module 3006 selects a subset of resolutions to be used for TDA generation.

[0461] The lens parameter selection module 3006 choice of resolution may depend on the number of points in the space (here denoted by N) and the number of lenses (here denoted by L_n in the metric-lens(es) combination). Since resolutions may be multiplicative with the lenses (that is, if there are two lenses with resolution 10, there may be 100 buckets), the per-lens resolutions may be adjusted by taking the L_n -th root in some embodiments.

[0462] In some embodiments, different lens parameters are given by this formulas:

For gain evaluate {2., 3., 4.}

For resolution, evaluate for each j in [0, NUM_RES-1] (where NUM_RES is the number of resolutions to be considered).

$$res = \text{Math.pow}((\text{Math.pow}(\text{Math.max}(gain * N / (L_n * 100.0), 10), L_n) + \text{Math.pow}((\text{Math.sqrt}(N)/4.0) * j, L_n)), 1/(\text{double})L_n)$$

[0463] This equation may be written as:

$$res = \left(\left[\max \left(\frac{gain * N}{L_n * 100}, 10 \right) \right]^{L_n} + \left(\frac{\sqrt{N}}{4} * j \right)^{L_n} \right)^{\frac{1}{L_n}}$$

[0464] The last resolution value (starting with 0) may be recalled and if res is not at least three greater than the last one resolution, the res may be skipped. In some embodiments, NUM_RES=20 is enough values to effectively investigate each metric-lens(es) combination. The number of parameter combinations for a metric-lens(es) combination may be denoted by PCML. For example, this may be $20 * 3 * 2$ (uniformized and not). In some embodiments, the formula covers a range of possible gains. It will be appreciated that are many other possible ways of finding and selecting resolutions.

[0465] In constructing the TDA graph, as discussed regarding FIG. 3, the input module 314 may receive the original data receives data S. The input module 314 may generate reference space R using metric-lens(es) combination of the subset of metric-lens combinations. The

analysis module 320 may generate a map $\text{ref}()$ from S into R . The map $\text{ref}()$ from S into R may be called the “reference map.” In one example, a reference of map from S is to a reference metric space R . The map can be described by one or more lenses (i.e., real valued functions on S). The resolution module 218 generates a cover of R based on the resolution from the lens parameter selection module 3006. The cover of R may be a finite collection of open sets (in the metric of R) such that every point in R lies in at least one of these sets. In various examples, R is k -dimensional Euclidean space, where k is the number of filter functions. In this example, R is a box in k -dimensional Euclidean space given by the product of the intervals $[\text{min}_k, \text{max}_k]$, where min_k is the minimum value of the k -th filter function on S , and max_k is the maximum value.

[0466] The analysis module 320 may cluster each $S(d)$ based on the metric, filter, and the space S . In some embodiments, a dynamic single-linkage clustering algorithm may be used to partition $S(d)$. It will be appreciated that any number of clustering algorithms may be used with embodiments discussed herein. For example, the clustering scheme may be k -means clustering for some k , single linkage clustering, average linkage clustering, or any method specified by the user.

[0467] The visualization engine 322 may identify nodes which are associated with a subset of the partition elements of all of the $S(d)$ for generating an interactive visualization. For example, suppose that $S = \{1, 2, 3, 4\}$, and the cover is $C1, C2, C3$. Then if $\text{ref_tags}(1) = \{1, 2, 3\}$ and $\text{ref_tags}(2) = \{2, 3\}$, and $\text{ref_tags}(3) = \{3\}$, and finally $\text{ref_tags}(4) = \{1, 3\}$, then $S(1)$ in this example is $\{1, 4\}$, $S(2) = \{1, 2\}$, and $S(3) = \{1, 2, 3, 4\}$. If 1 and 2 are close enough to be clustered, and 3 and 4 are, but nothing else, then the clustering for $S(1)$ may be $\{1\} \{3\}$, and for $S(2)$ it may be $\{1, 2\}$, and for $S(3)$ it may be $\{1, 2\}, \{3, 4\}$. So the generated graph has, in this example, at most four nodes, given by the sets $\{1\}, \{4\}, \{1, 2\}$, and $\{3, 4\}$ (note that $\{1, 2\}$ appears in two different clusterings). Of the sets of points that are used, two nodes intersect provided that the associated node sets have a non-empty intersection (although this could easily be modified to allow users to require that the intersection is “large enough” either in absolute or relative terms).

[0468] Nodes may be eliminated for any number of reasons. For example, a node may be eliminated as having too few points and/or not being connected to anything else. In some embodiments, the criteria for the elimination of nodes (if any) may be under user control or have application-specific requirements imposed on it. For example, if the points are

consumers, for instance, clusters with too few people in area codes served by a company could be eliminated. If a cluster was found with “enough” customers, however, this might indicate that expansion into area codes of the other consumers in the cluster could be warranted.

[0469] The visualization engine 322 may join clusters to identify edges (e.g., connecting lines between nodes). Once the nodes are constructed, the intersections (e.g., edges) may be computed “all at once,” by computing, for each point, the set of node sets (not `ref_tags`, this time). That is, for each s in S , `node_id_set(s)` may be computed, which is an `int[]`. In some embodiments, if the cover is well behaved, then this operation is linear in the size of the set S , and we then iterate over each pair in `node_id_set(s)`. There may be an edge between two `node_id`’s if they both belong to the same `node_id_set()` value, and the number of points in the intersection is precisely the number of different `node_id` sets in which that pair is seen. This means that, except for the clustering step (which is often quadratic in the size of the sets $S(d)$, but whose size may be controlled by the choice of cover), all of the other steps in the graph construction algorithm may be linear in the size of S , and may be computed quite efficiently.

[0470] In step 3804, the group identification module 3008 may identify groups of nodes of each TDA graph that share a similar outcome (e.g., share a shared characteristic). In some embodiments, the nodes may be colored or otherwise noted as belonging to or associated with one or more outcomes. As discussed herein, the outcomes may be identified in one or more columns in data S . Each node may be associated with one or more outcomes in any number of ways.

[0471] In some embodiments, if all data points belonging to a node share the same or similar outcome, the node may be colored or otherwise noted as belonging or being associated with that same or similar outcome. In some embodiments, the group identification module 3008 may determine that if there are a majority of data points of a node (e.g., or an average, median, or mean), then the node will be colored or otherwise noted as belonging or being associated with that same or similar outcome.

[0472] Groups of nodes include nodes that share the same or similar outcomes (or groups of similar distributions of outcomes). Groups of nodes that share the same or similar outcomes may be found in any number of ways.

[0473] For each group of each TDA graph, the group score module 3010 computes the outcome entropy of the outcome within that group. In various embodiments, each graph may

comprise any number of groups of nodes with shared similar or same outcomes. Each group may also be associated with an outcome entropy computed by the group score module 3010.

[0474] In step 3812, the graph score module 3012 scores each TDA graph based on the number of groups in that particular TDA graph and the computed outcome entropy of each group in that particular TDA graph. In one example, the graph score module 3012 calculates the TDA graph score using the following:

$$\left(\sum_{groups\ g} entropy(g) * \#pts(g) + \frac{N}{50 * \#pts(g)} \right) * \begin{cases} \text{if } \#groups < \\ \#cat, \text{ then } \frac{\#cats}{\#groups}, \text{ else } 1 \end{cases}$$

[0475] Groups g represent the number of groups in a particular TDA graph. Entropy (g) is the entropy for the group (g) in that particular TDA graph which is multiplied by the number of points in the group (g). The number of nodes divided by a constant (e.g., 50) multiplied by the number of points in the group is added. It will be appreciated that any constant may be used (e.g., between 10 and 90). The score may be multiplied by the number of outcome categories over the number of groups if the number of groups is less than the number of outcome categories (to penalize the TDA graph score) or multiply by one, otherwise. It will be appreciated that each graph may be associated with a TDA graph score.

[0476] In various embodiments, the graph selection module 3014 may optionally rank or order all or some of the graphs using the TDA graph score.

[0477] In some embodiments, the graph selection module 3014 may select one or more of the graphs, metrics, metric-lens(es) combinations, resolutions, or any combination using the TDA graph score. For example, the graph selection module 3014 may select a predetermined number of TDA graphs, select TDA graphs with a TDA graph score that meet a certain criteria (e.g., choose TDA graphs with a TDA graph score over a predetermined threshold), select TDA graphs that are in a top predetermined percentage, or any other method.

[0478] In various embodiments, the outcome visualization module 3016 may generate visualizations of one or more selected TDA graphs (e.g., graphs selected by the graph selection module 3014).

[0479] For example, the outcome visualization module 3016 may generate visualizations of one or more selected TDA graphs including visualizations of graphs depicted in FIGS. 39A-39D and FIGS. 40A-40D.

[0480] FIGS. 39A-39B depict visualizations of selected TDA graphs of Gaussian data. FIG. 39A may depict a visualization of a graph that is most preferable when compared to others, FIG. 39B may depict a visualization of a graph that is the next most preferable when compared to others.

[0481] FIG. 39A depicts a visualization of a graph using a Chebyshev (L-Infinity) metric and a neighborhood lens 2 (resolution 61, gain of 4.0). As can be shown, the groupings of nodes in the visualization are grouped by outcome and the groupings show consistent outcomes.

[0482] FIG. 39B depicts a visualization of a graph using a variance normalized Euclidean metric and a neighborhood lens 1 (resolution 57, gain of 3.0). This visualization is similar to FIG. 39A. The groupings of nodes in the visualization are grouped by outcome and the groupings show consistent outcomes.

[0483] The above-described functions and components can be comprised of instructions that are stored on a storage medium (e.g., a computer readable storage medium). The instructions can be retrieved and executed by a processor. Some examples of instructions are software, program code, and firmware. Some examples of storage medium are memory devices, tape, disks, integrated circuits, and servers. The instructions are operational when executed by the processor (e.g., a data processing device) to direct the processor to operate in accord with embodiments of the present invention. Those skilled in the art are familiar with instructions, processor(s), and storage medium.

[0484] The present invention has been described above with reference to exemplary embodiments. It will be apparent to those skilled in the art that various modifications may be made and other embodiments can be used without departing from the broader scope of the invention. Therefore, these and other variations upon the exemplary embodiments are intended to be covered by the present invention.

CLAIMS

1. A non-transitory computer readable medium including executable instructions, the instructions being executable by a processor to perform a method, the method comprising:
 - receiving a first set of data identifying entities and performance information for analysis;
 - receiving a second set of data identifying entities and performance information associated with known or suspected past fraud or abuse;
 - receiving metric and lens selections;
 - performing metric and lens functions, based on the metric and lens selections, on first and second set of data to map performance information from the first and second sets to a reference space;
 - generating cover of reference space and cluster mapped performance information to identify nodes in a graph, each node including one or more entities as members, each node being connected to another node if they share at least one common entity as members;
 - identifying nodes that include at least one member from the second set of data;
 - determining entities that are members of the identified nodes that are from the first set of data; and
 - generating a first report listing the determined entities that are members of the identified nodes that are from the first set of data as possibly involved in fraud or waste.
2. The non-transitory computer readable medium of claim 1, wherein the method further comprises determining other entities that are member of other nodes that are connected to the identified nodes by an edge, the other entities being from the first set of data, and adding one or more of the other entities to the first report.
3. The non-transitory computer readable medium of claim 1, wherein the entities of the first set of data are providers of health services and the performance information is health care information including charges for treatment and treatments provided to patients.
4. The non-transitory computer readable medium of claim 1, wherein the entities of the first set of data are consumers of services and the performance information is access information and resource utilization of networks and network resources.

5. The non-transitory computer readable medium of claim 1, wherein the method further comprises applying one or more functions on at least some performance data of the first set of data and including those determined entities that are members of the identified nodes that are from the first set of data in the first report based, in part, on at least one value calculated as a result of the one or more functions.
6. The non-transitory computer readable medium of claim 1, wherein the one or more functions include an L1 function or L infinity function.
7. The non-transitory computer readable medium of claim 1, wherein the method further comprises:
- receiving a new entity with new performance information associated with that new entity;
 - determining distances between new performance information of new entity and performance information of entities of first and second sets of data;
 - comparing distances between new performance information for new entity and the distances between entities of each node;
 - determining location of new entity in the TDA graph based on the comparison; and
 - generating a second report if the new entity is determined to be in a node that has at least one member from the second set of data.
8. The non-transitory computer readable medium of claim 1, wherein the method further comprises:
- receiving a new entity with new performance information associated with that new entity;
 - determining distances between new performance information of new entity and performance information of entities of first and second sets of data;
 - comparing distances between new performance information for new entity and the distances between entities of each node;
 - determining location of new entity in the TDA graph based on the comparison; and
 - generating a second report if the new entity is determined to be in a node that is linked by an edge to a node that has at least one member from the second set of data.
9. The non-transitory computer readable medium of claim 1, wherein the method further comprises:

removing performance information from the first set of data if the performance information is older than a predetermined date leaving remaining performance information;

performing the metric and lens functions on first and second set of data to map remaining performance information from the first set of data and the performance information from the second set of data to the reference space; and

generating cover of the reference space and cluster mapped performance information to identify nodes in the graph.

10. A method comprising:

receiving a first set of data identifying entities and performance information for analysis;

receiving a second set of data identifying entities and performance information associated with known or suspected past fraud or abuse;

receiving metric and lens selections;

performing metric and lens functions, based on the metric and lens selections, on first and second set of data to map performance information from the first and second sets to a reference space;

generating cover of reference space and cluster mapped performance information to identify nodes in a graph, each node including one or more entities as members, each node being connected to another node if they share at least one common entity as members;

identifying nodes that include at least one member from the second set of data;

determining entities that are members of the identified nodes that are from the first set of data; and

generating a first report listing the determined entities that are members of the identified nodes that are from the first set of data as possibly involved in fraud or waste.

11. The method of claim 10, further comprising determining other entities that are member of other nodes that are connected to the identified nodes by an edge, the other entities being from the first set of data, and adding one or more of the other entities to the first report.

12. The method of claim 10, wherein the entities of the first set of data are providers of health services and the performance information is health care information including charges for treatment and treatments provided to patients.

13. The method of claim 10, wherein the entities of the first set of data are consumers of services and the performance information is access information and resource utilization of networks and network resources.
14. The method of claim 10, further comprising applying one or more functions on at least some performance data of the first set of data and including those determined entities that are members of the identified nodes that are from the first set of data in the first report based, in part, on at least one value calculated as a result of the one or more functions.
15. The method of claim 14, wherein the one or more functions include an L1 function or L infinity function.
16. The method of claim 10, further comprising:
- receiving a new entity with new performance information associated with that new entity;
 - determining distances between new performance information of new entity and performance information of entities of first and second sets of data;
 - comparing distances between new performance information for new entity and the distances between entities of each node;
 - determining location of new entity in the TDA graph based on the comparison; and
 - generating a second report if the new entity is determined to be in a node that has at least one member from the second set of data.
17. The method of claim 10, further comprising:
- receiving a new entity with new performance information associated with that new entity;
 - determining distances between new performance information of new entity and performance information of entities of first and second sets of data;
 - comparing distances between new performance information for new entity and the distances between entities of each node;
 - determining location of new entity in the TDA graph based on the comparison; and
 - generating a second report if the new entity is determined to be in a node that is linked by an edge to a node that has at least one member from the second set of data.
18. The method of claim 10, further comprising:

removing performance information from the first set of data if the performance information is older than a predetermined date leaving remaining performance information;

performing the metric and lens functions on first and second set of data to map remaining performance information from the first set of data and the performance information from the second set of data to the reference space; and

generating cover of the reference space and cluster mapped performance information to identify nodes in the graph.

19. A system comprising:

a processor; and

a memory including instructions to configure the processor to:

receive a first set of data identifying entities and performance information for analysis;

receive a second set of data identifying entities and performance information associated with known or suspected past fraud or abuse;

receive metric and lens selections;

perform metric and lens functions, based on the metric and lens selections, on first and second set of data to map performance information from the first and second sets to a reference space;

generate cover of reference space and cluster mapped performance information to identify nodes in a graph, each node including one or more entities as members, each node being connected to another node if they share at least one common entity as members;

identify nodes that include at least one member from the second set of data;

determine entities that are members of the identified nodes that are from the first set of data; and

generate a first report listing the determined entities that are members of the identified nodes that are from the first set of data as possibly involved in fraud or waste.

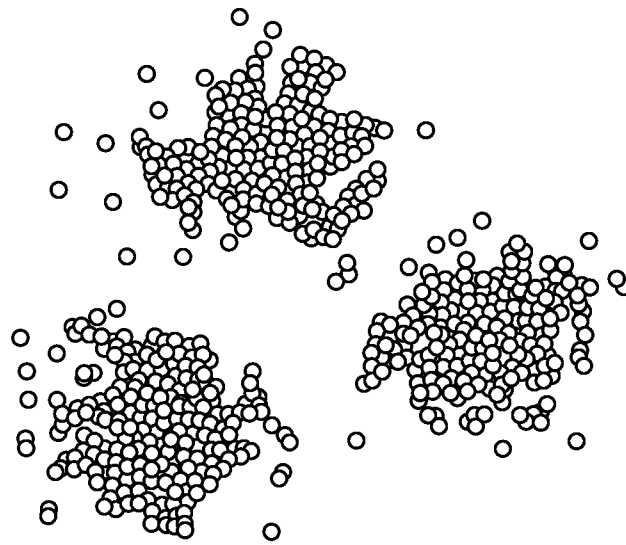


FIG. 1A

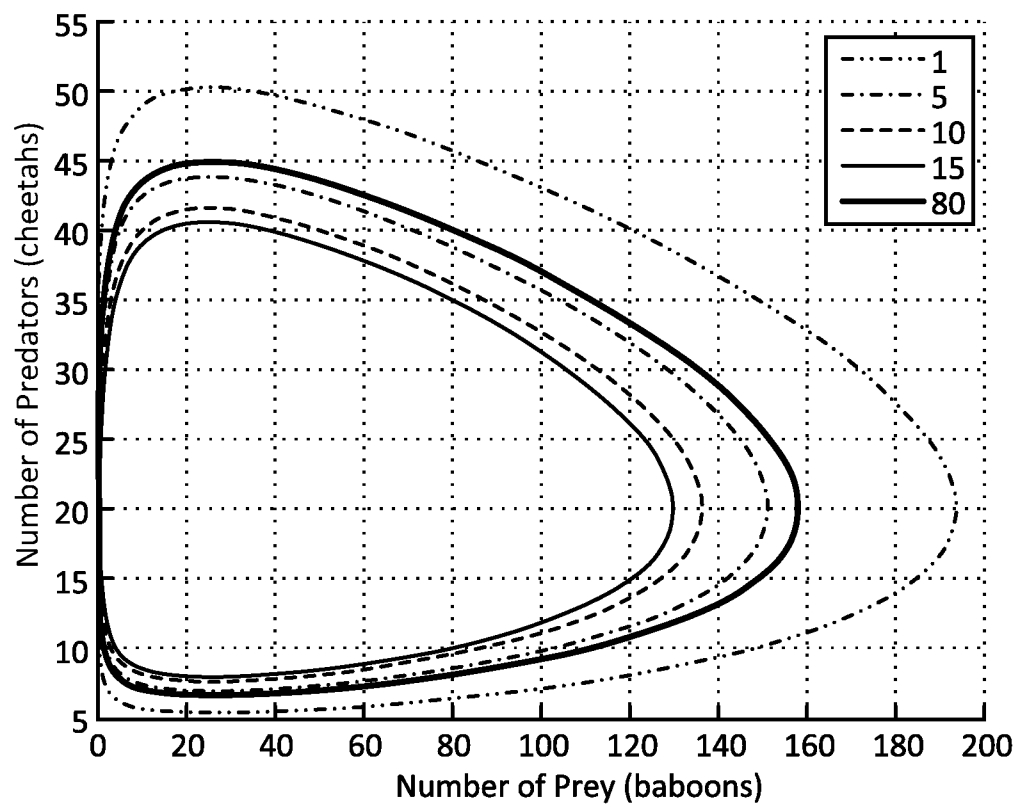


FIG. 1B



FIG. 1C

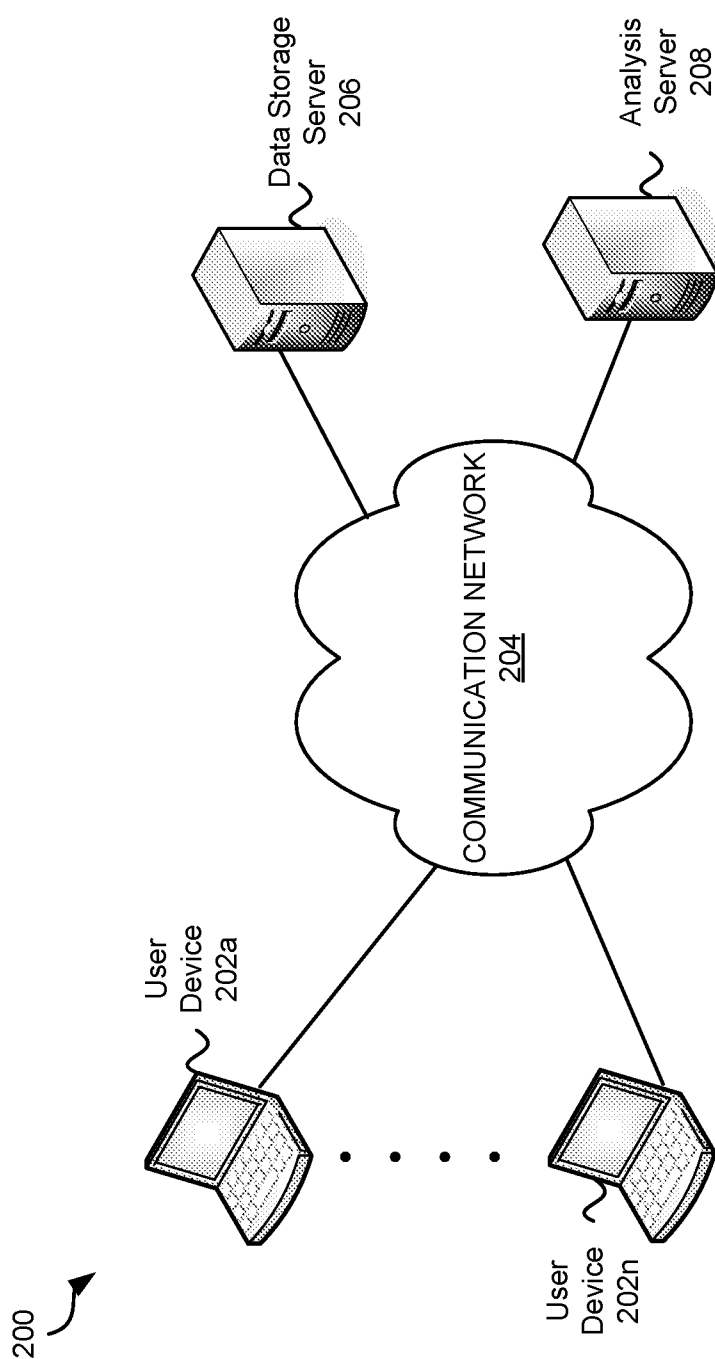


FIG. 2

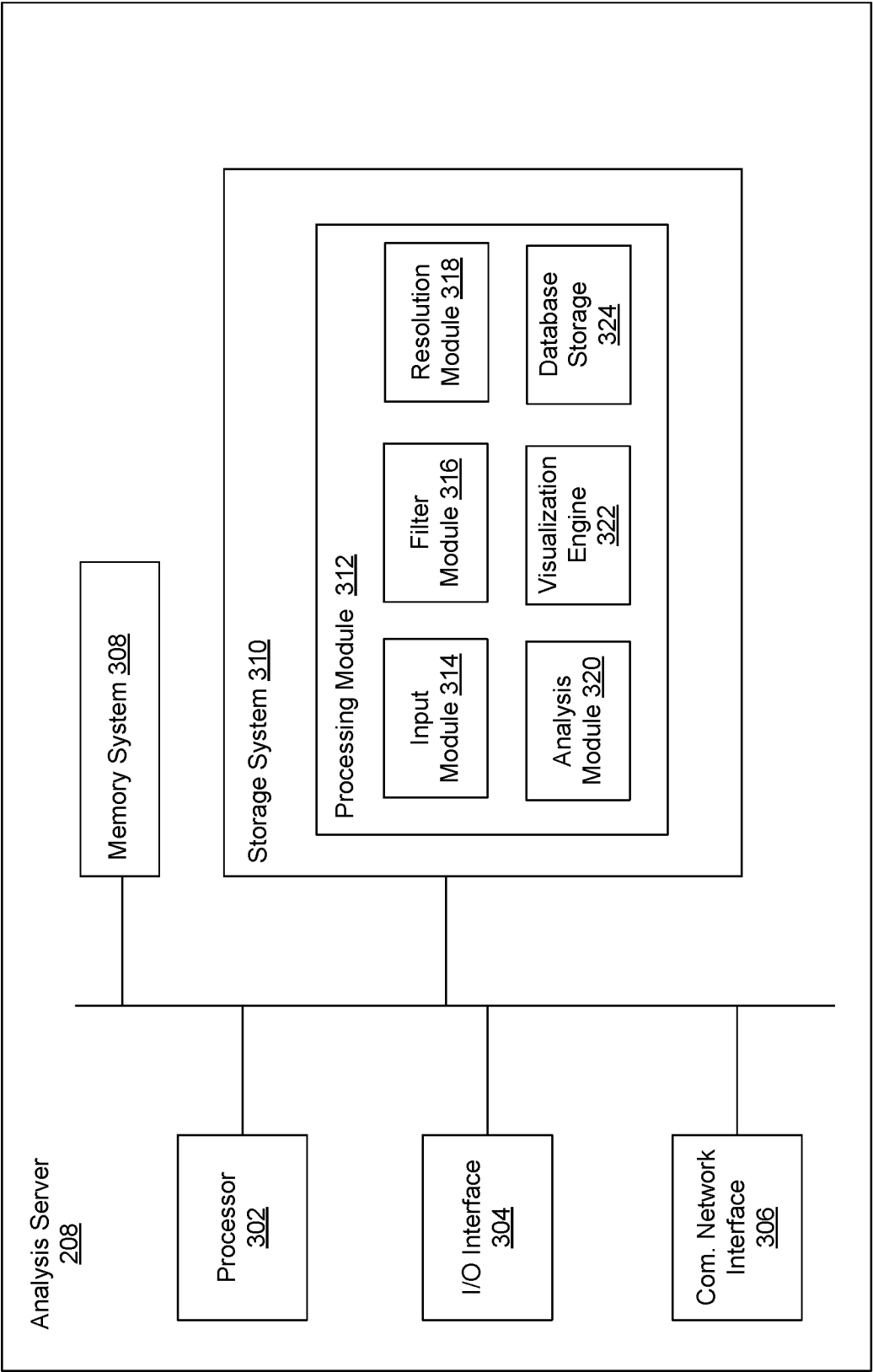


FIG. 3

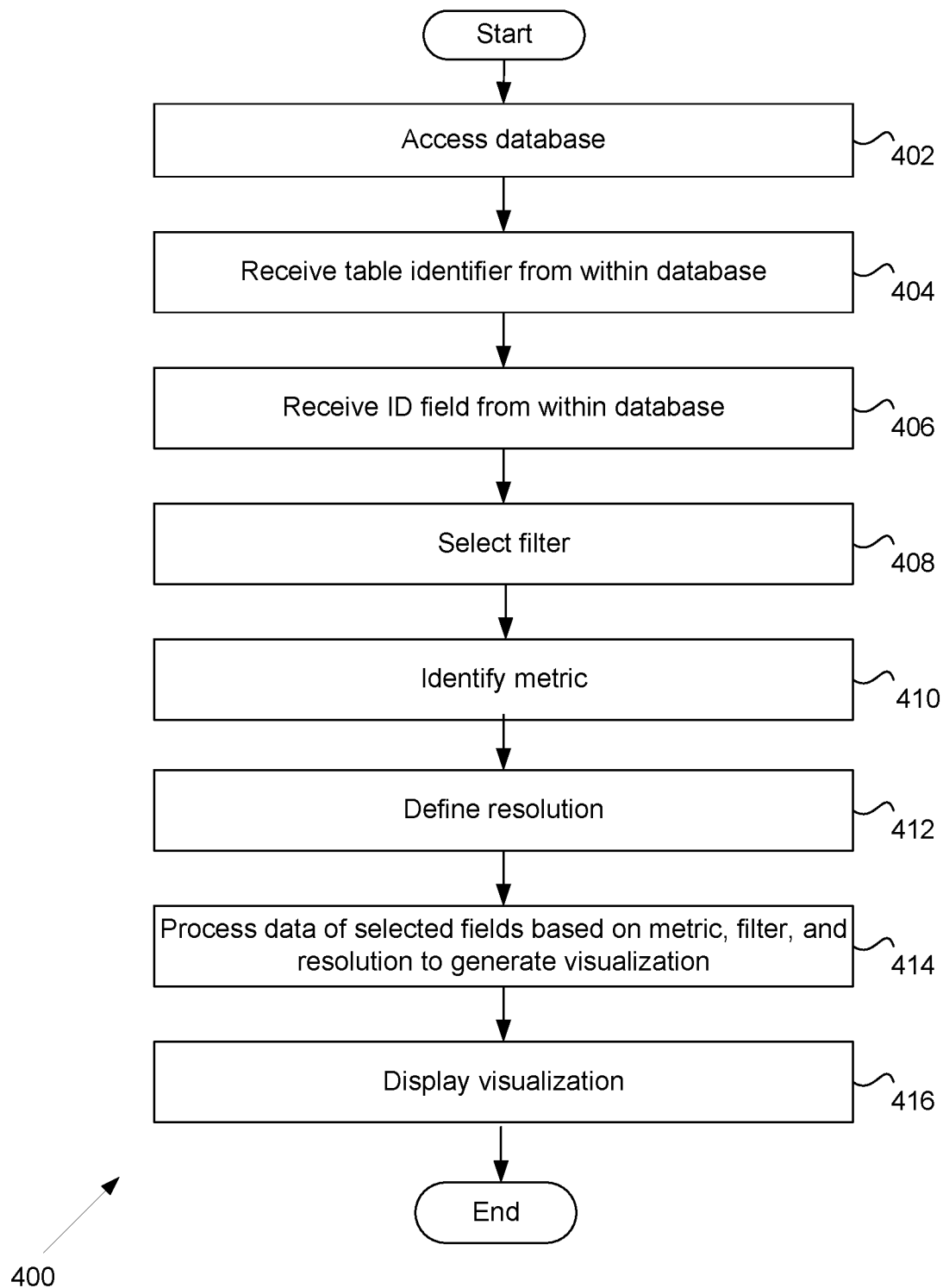


FIG. 4

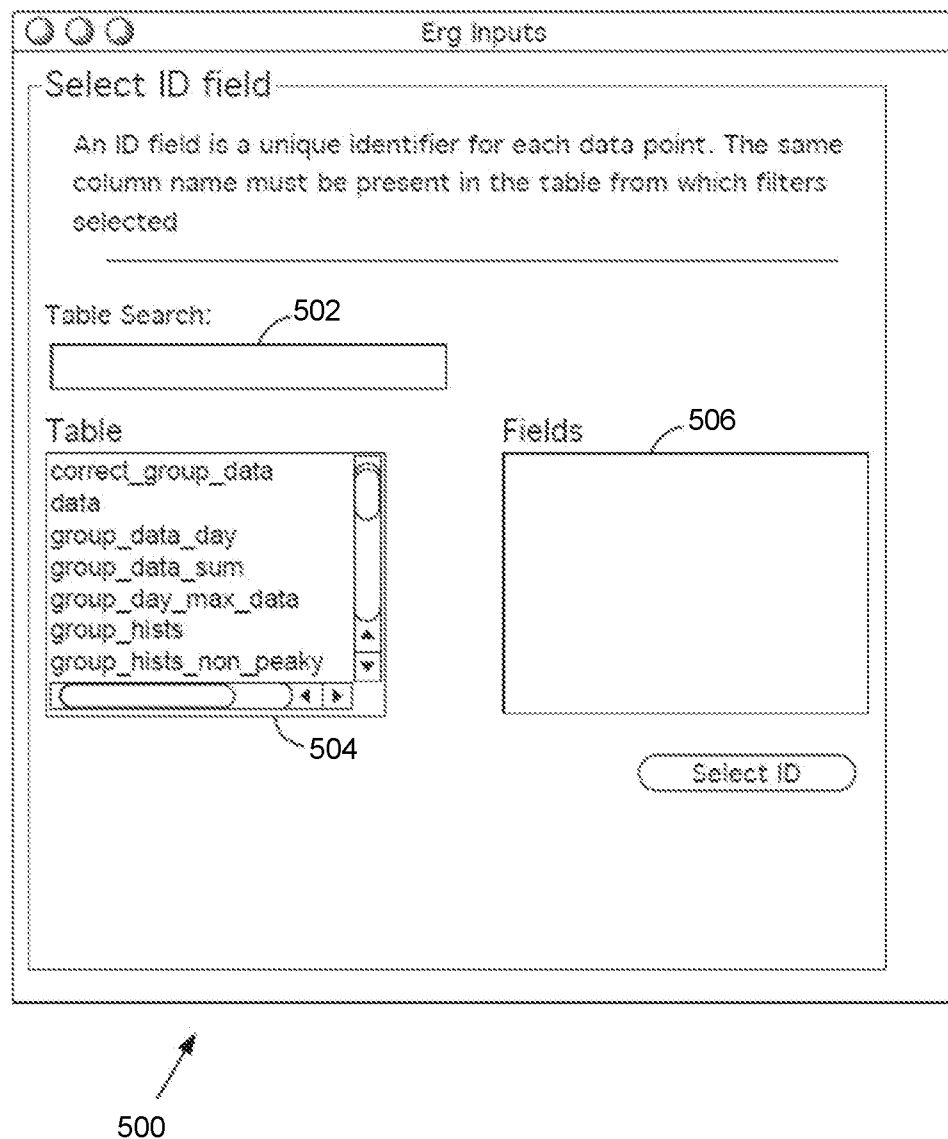


FIG. 5

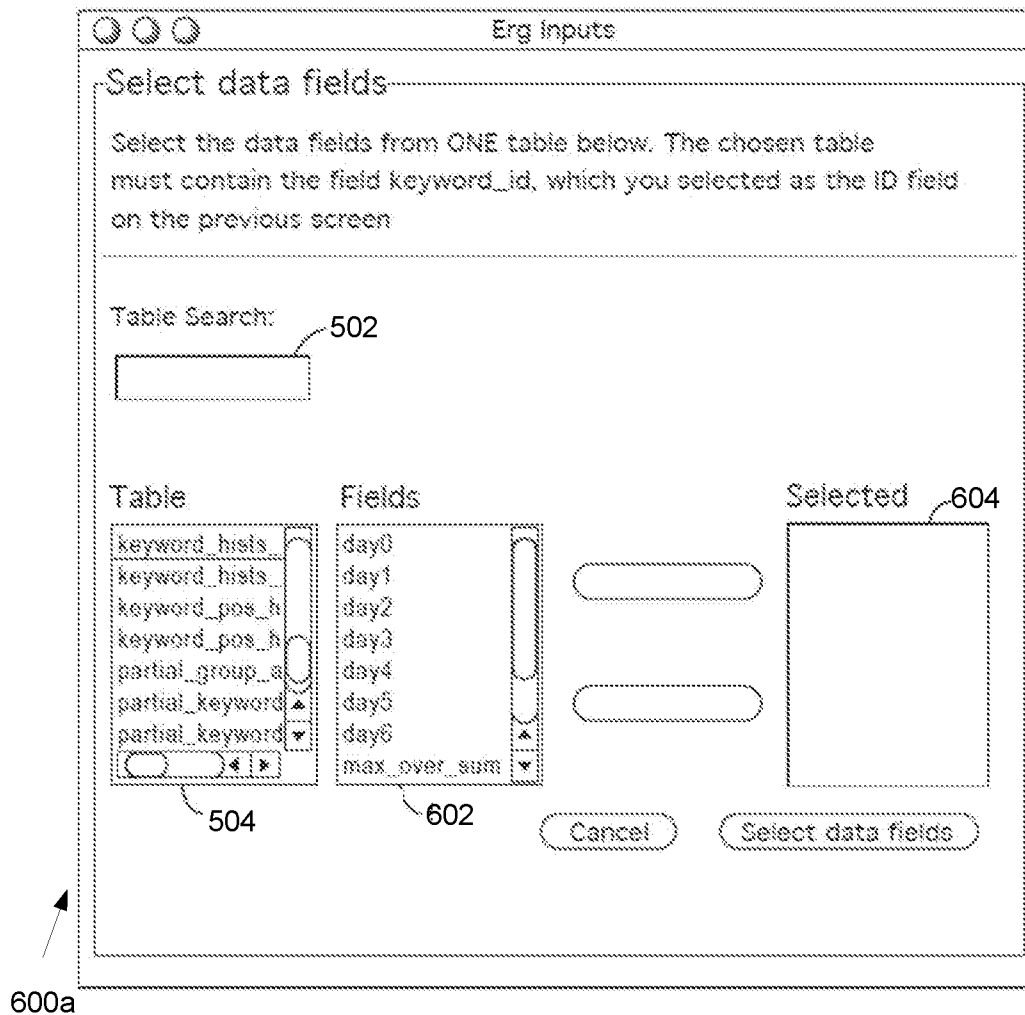
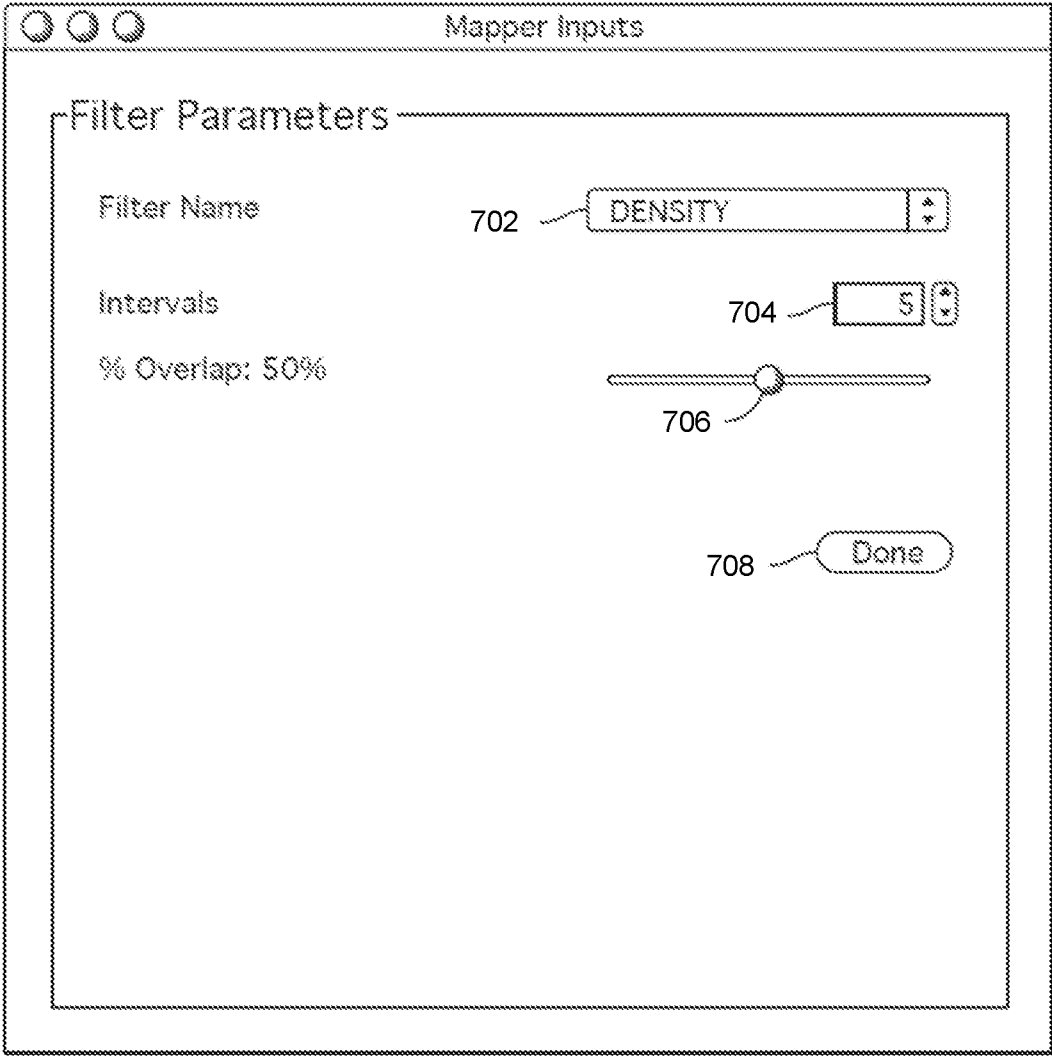


FIG. 6A

The image shows a graphical user interface window titled "Erg Inputs". Inside the window, there is a section titled "Add Filters/Adjust Filter Parameters". This section contains a dropdown menu (labeled 606) currently showing "EUCLIDEAN". Below the dropdown are three buttons: the first is labeled "Add Filter From Database" (labeled 608), the second is labeled "Add Geometric Filter" (labeled 610), and the third is an empty button. At the bottom of the window, there is a status bar that reads "Number of Filters: 0".

600b

FIG. 6B



700

FIG. 7

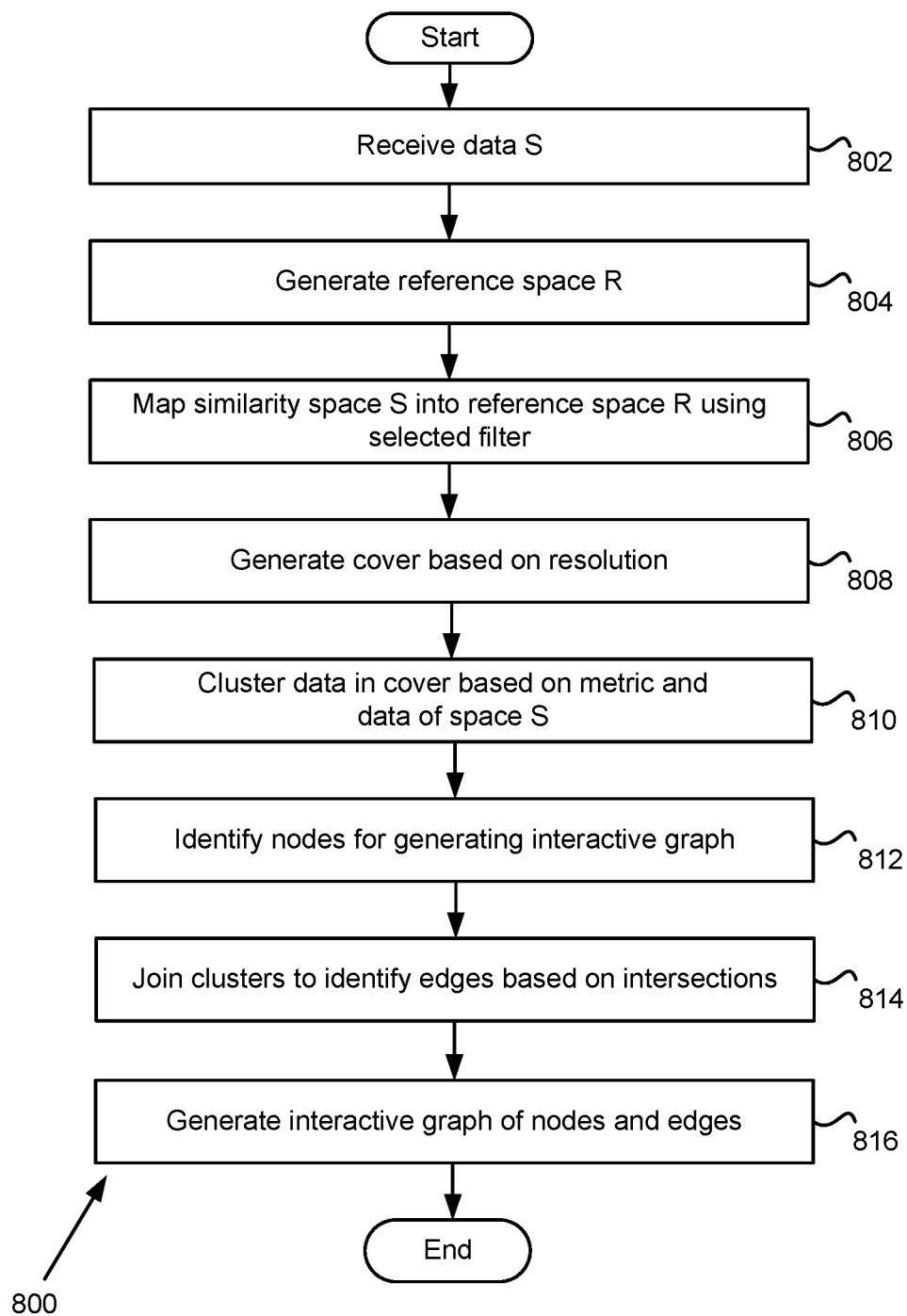
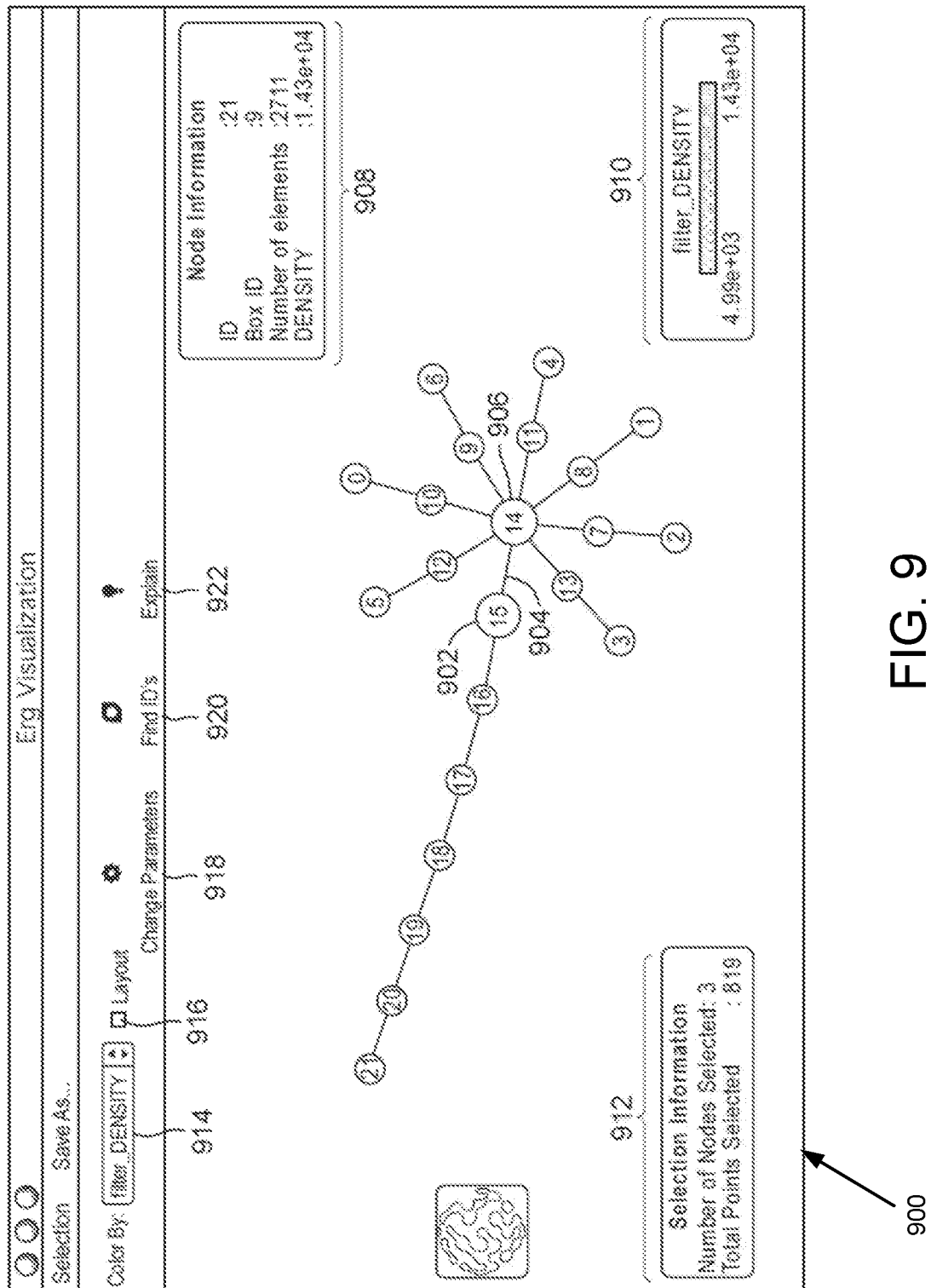


FIG. 8



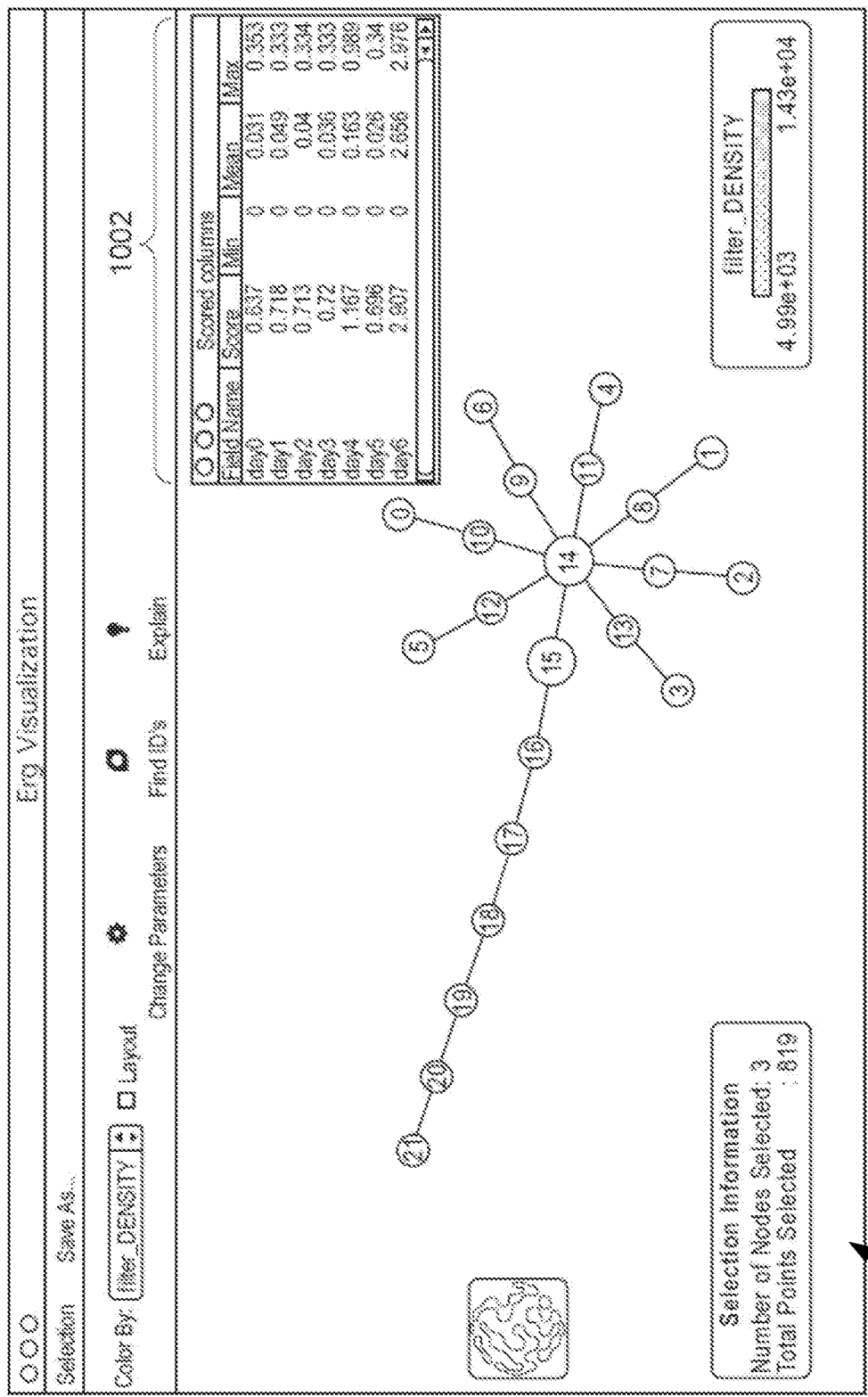
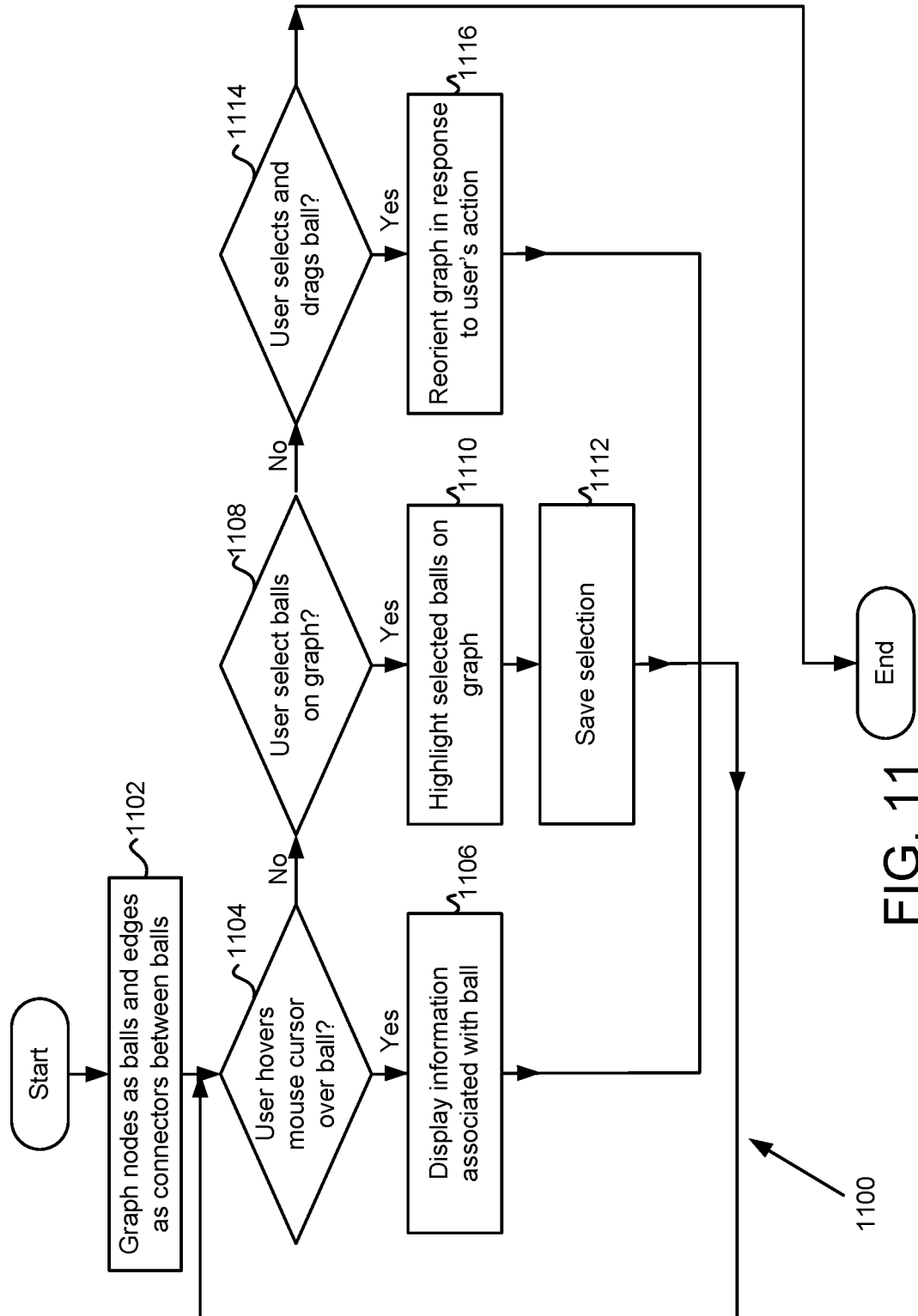


FIG. 10



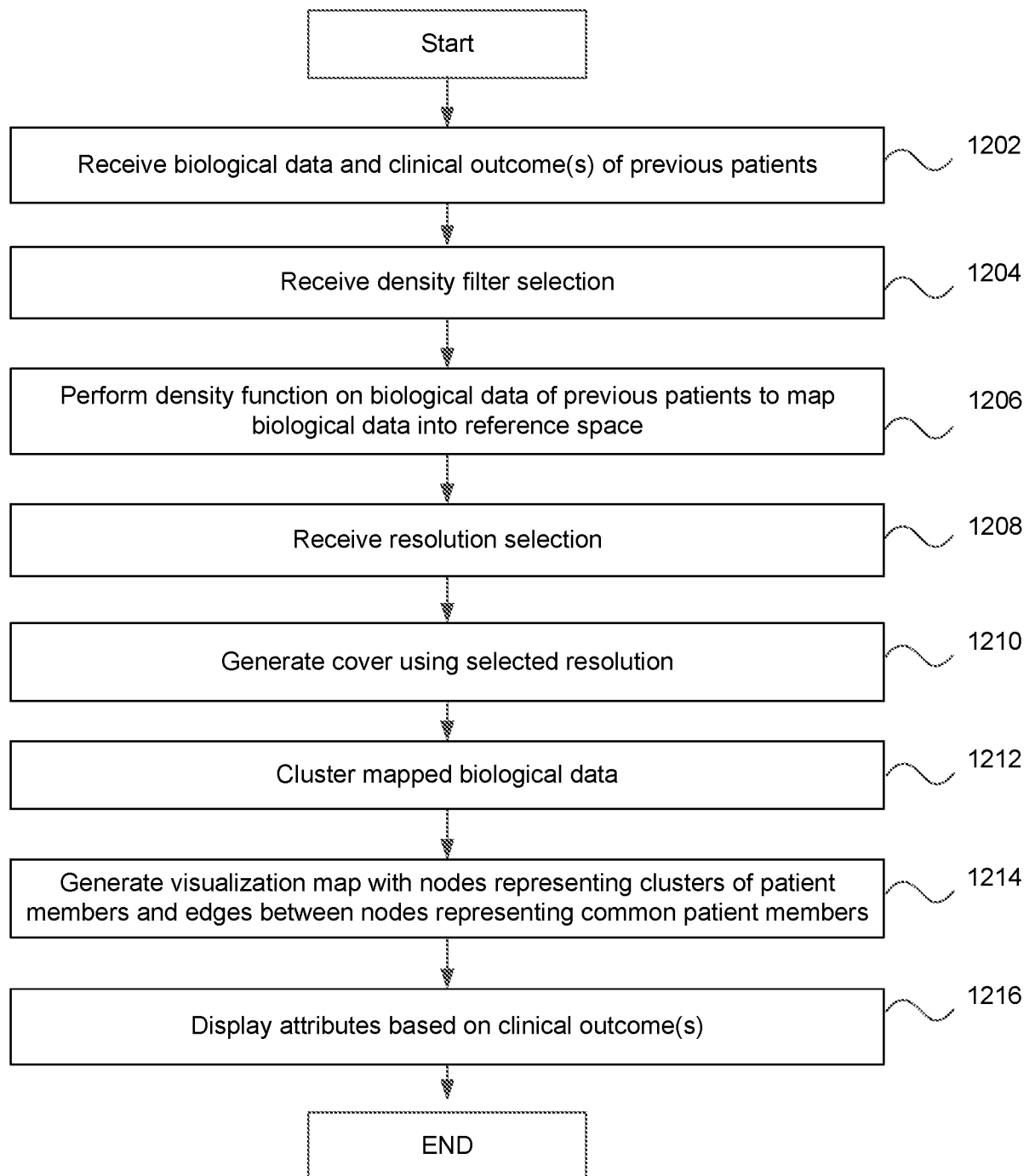


FIG. 12

1300

1308a 1308b 1308c 1308n	1302		1304a		1304b		1304y		1306
	Patient ID		Gene 1 Expression		Gene 2 Expression		Gene y Expression		Clinical Outcome
	P1		G1a		G2a		Gya		Outcome P1
	P2		G1b		G2b		Gyb		Outcome P2
	P3		G1c		G2c		Gyc		Outcome P3
	●		●		●		●		●
	●		●		●		●		●
	●		●		●		●		●
	Pn		G1n		G2n		Gyn		Outcome Pn

FIG. 13

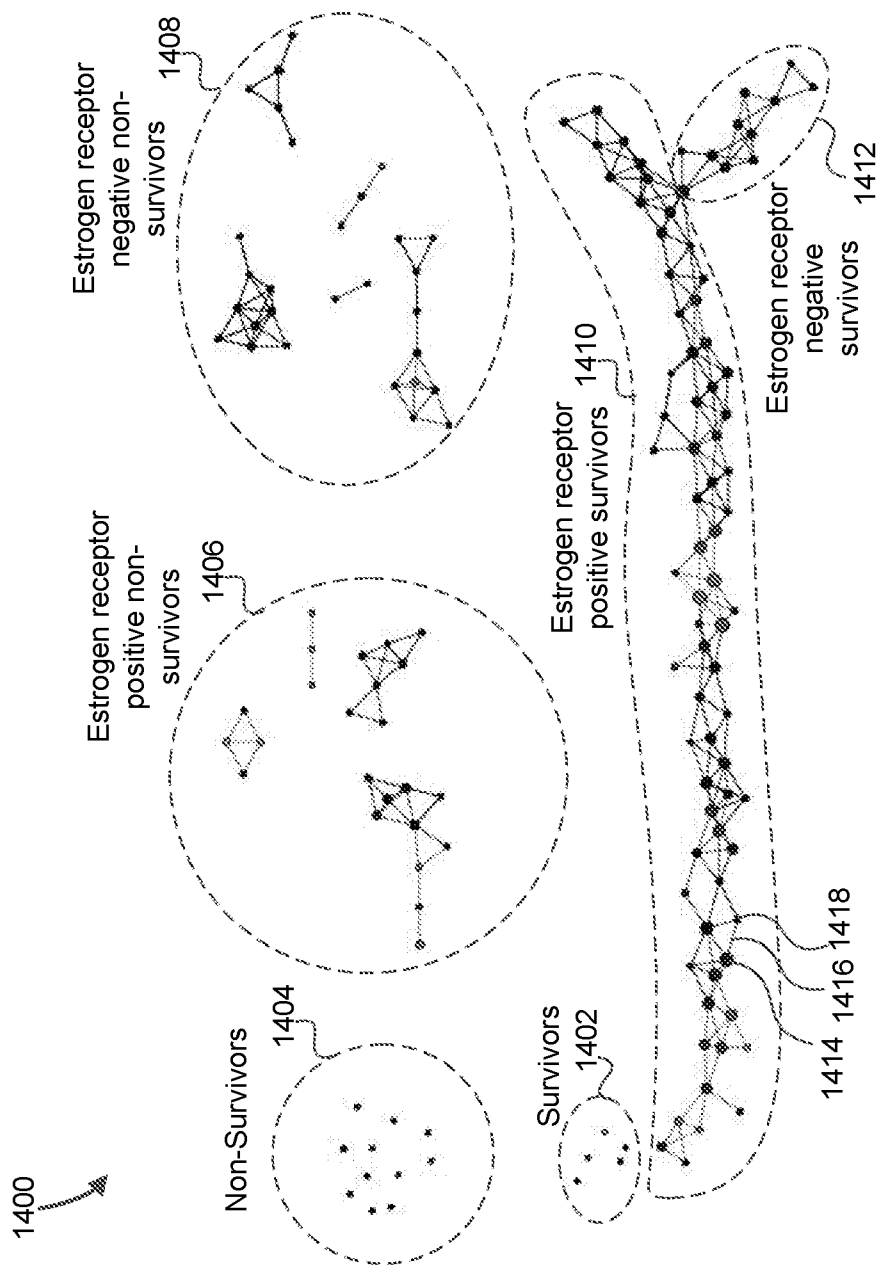


FIG. 14

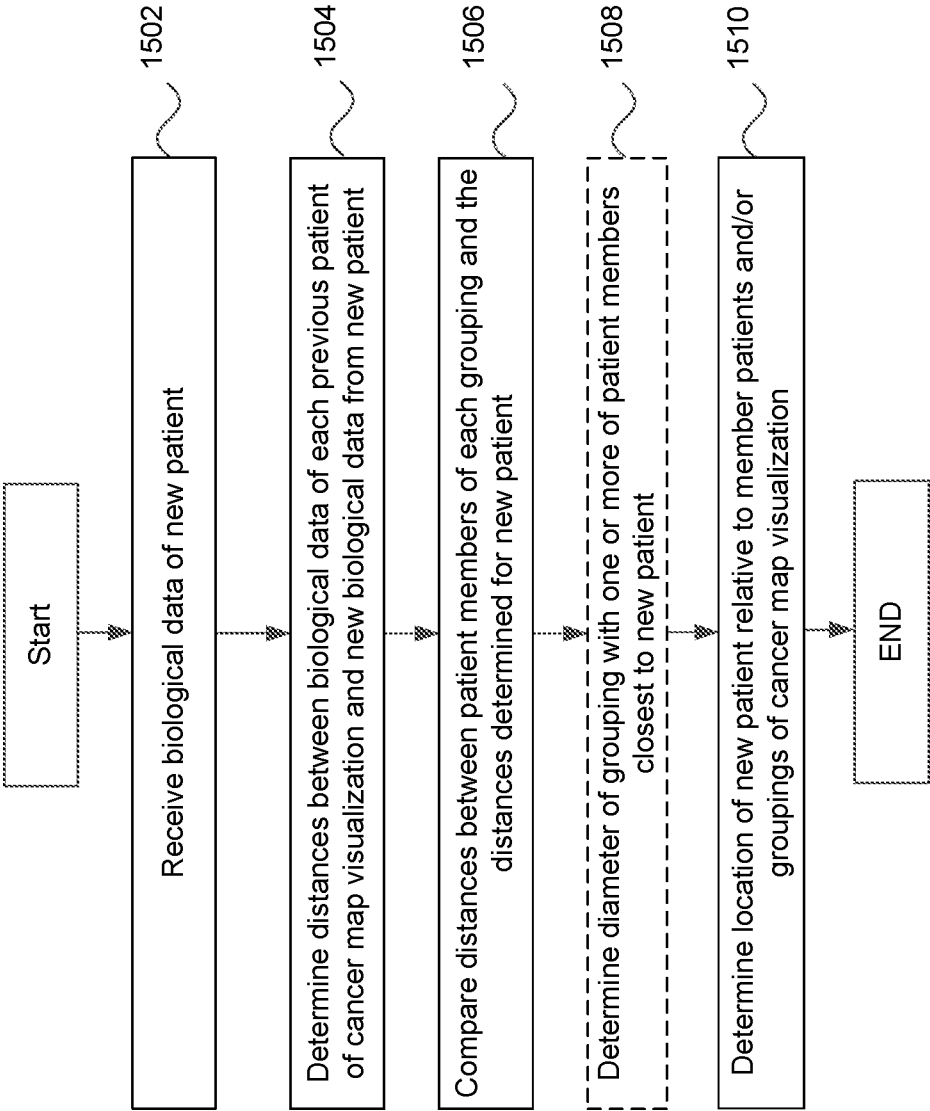


FIG. 15

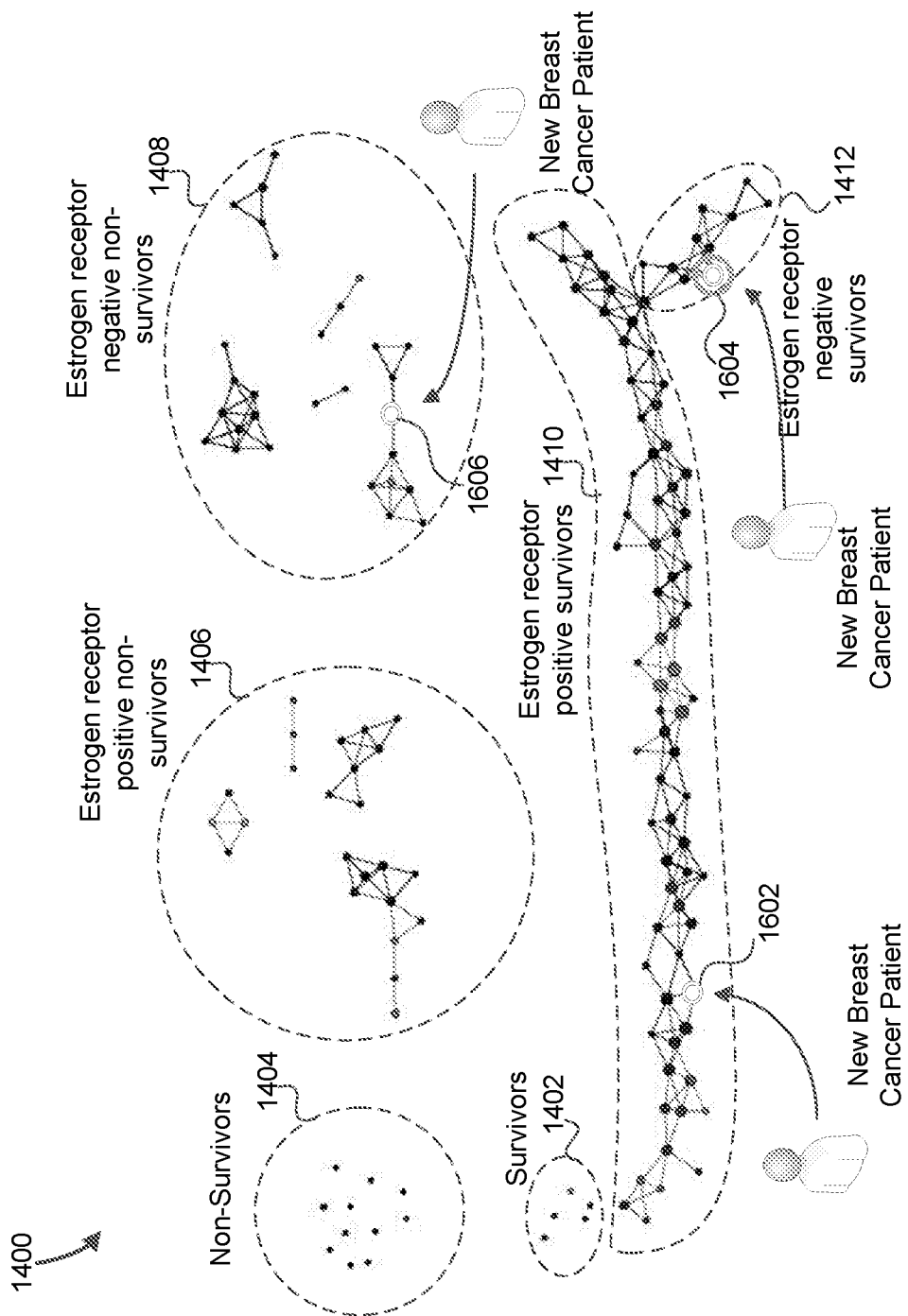


FIG. 16

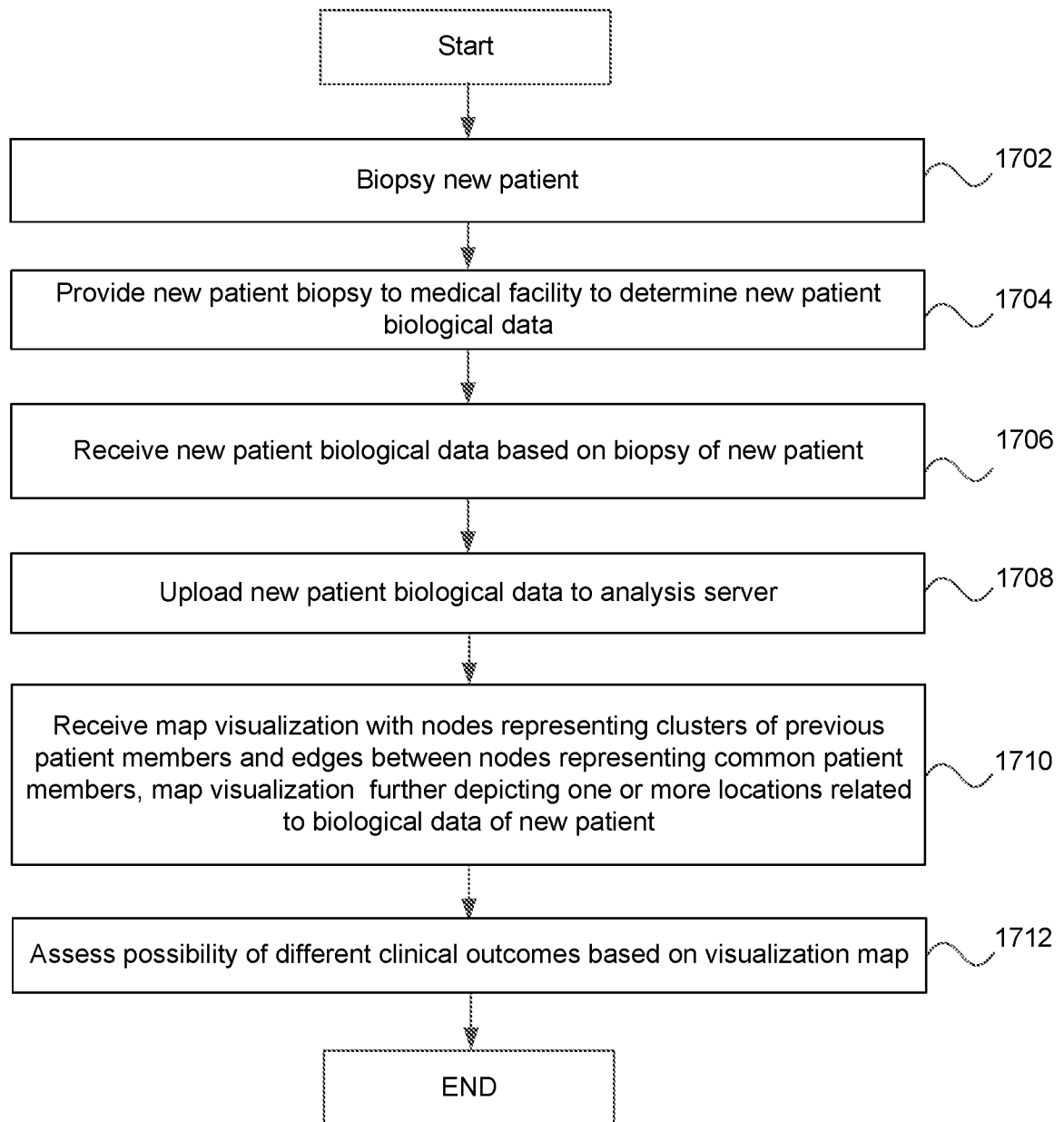


FIG. 17

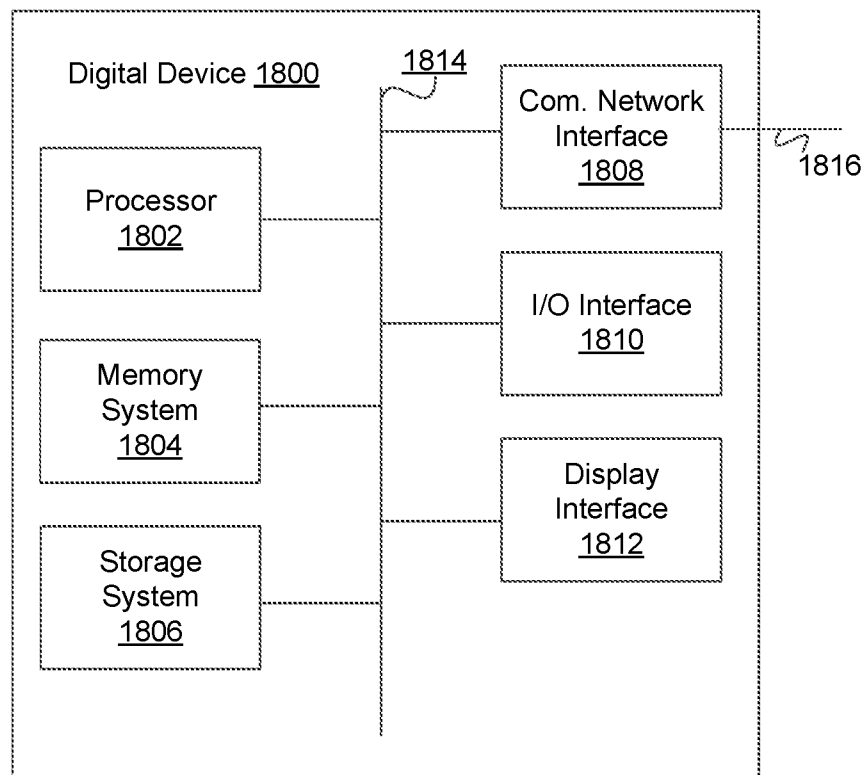


FIG. 18

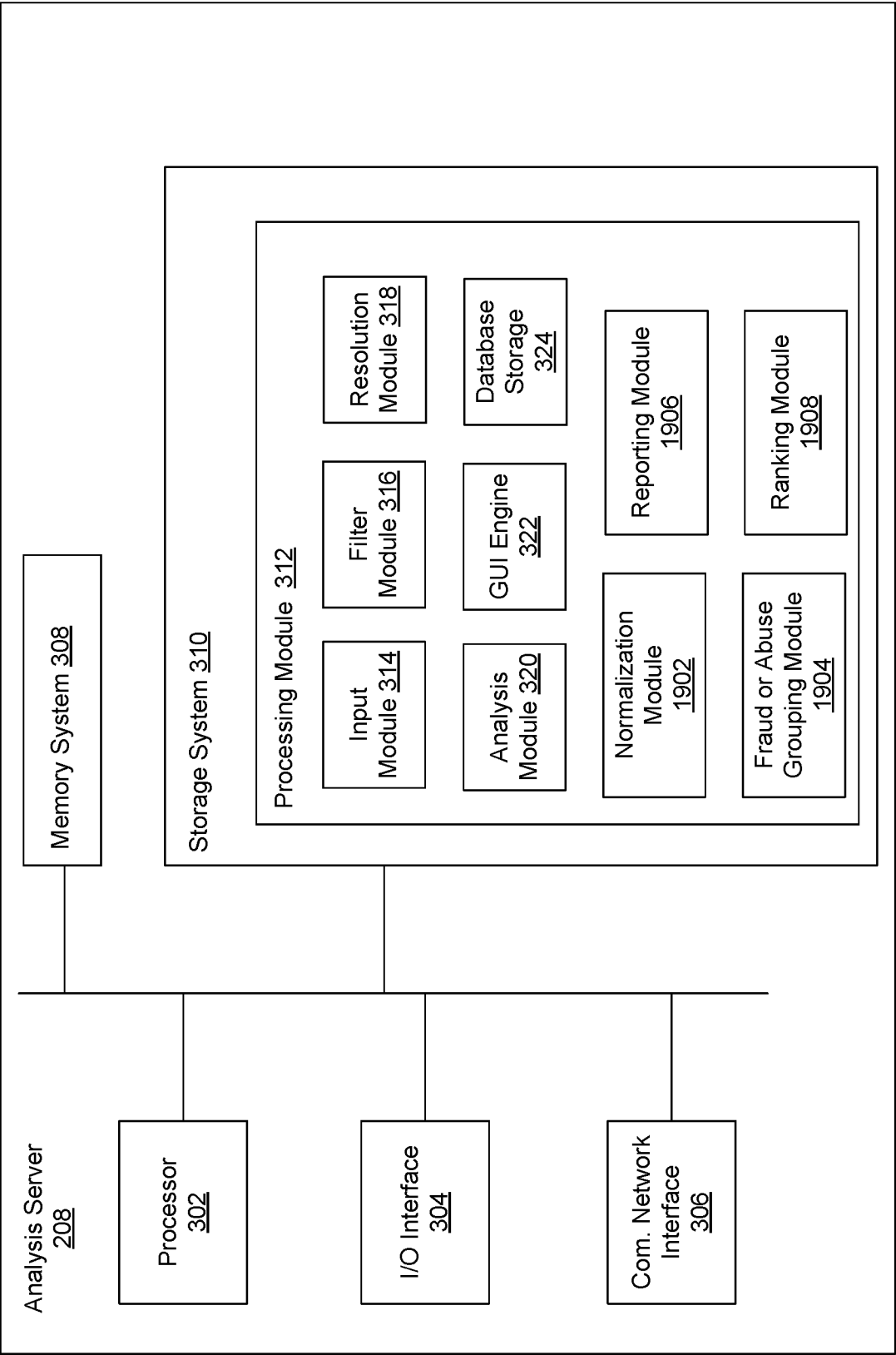


FIG. 19

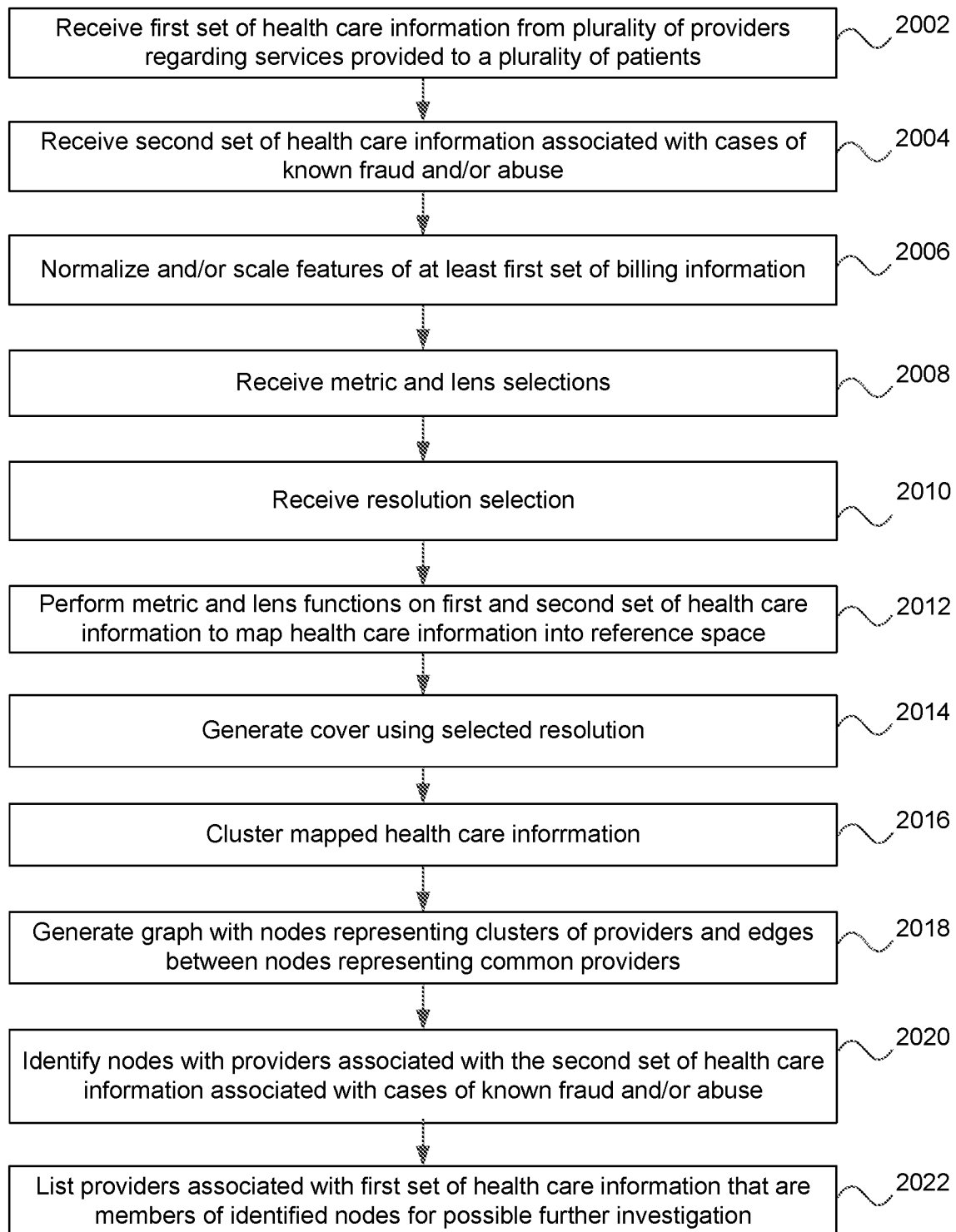
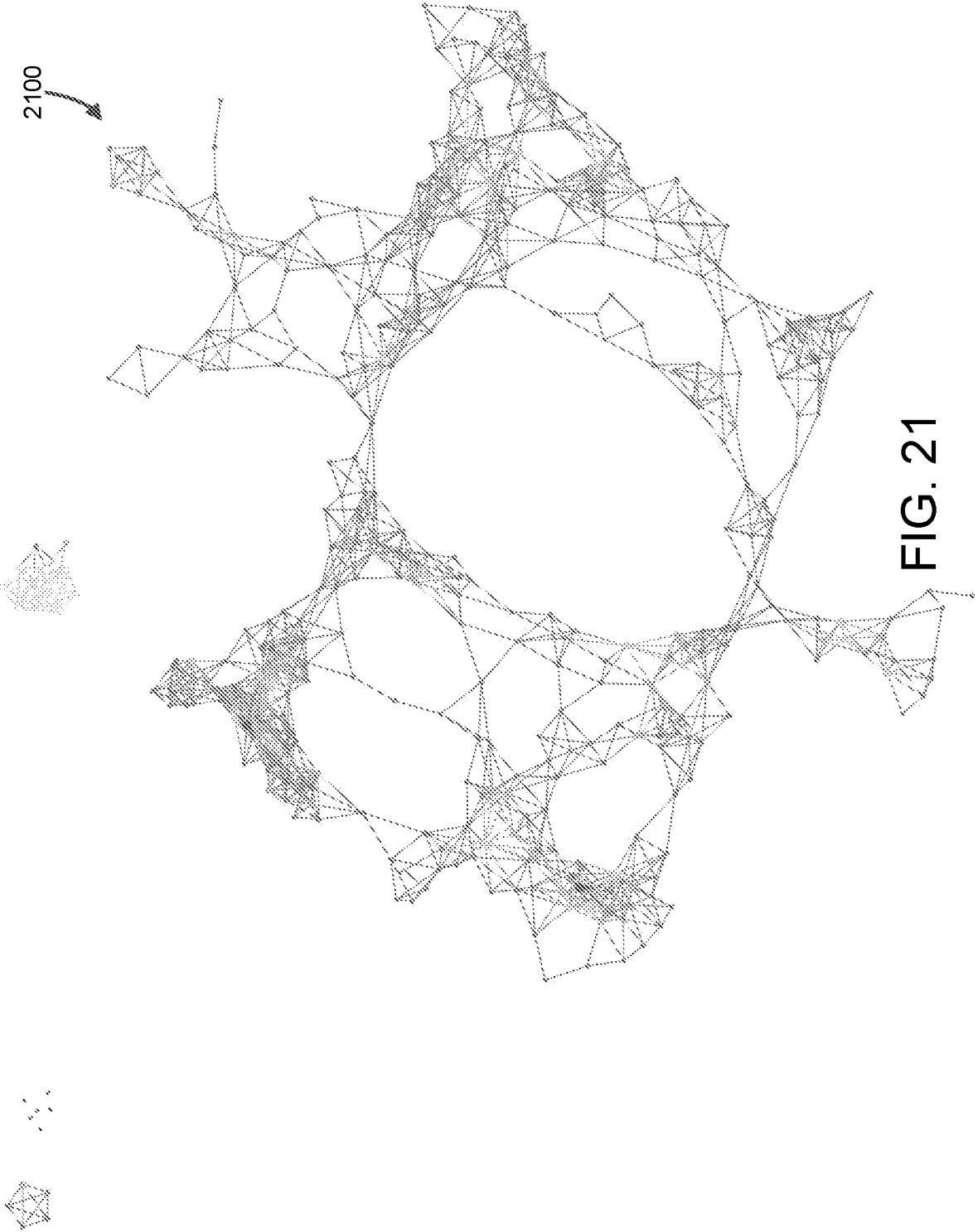


FIG. 20



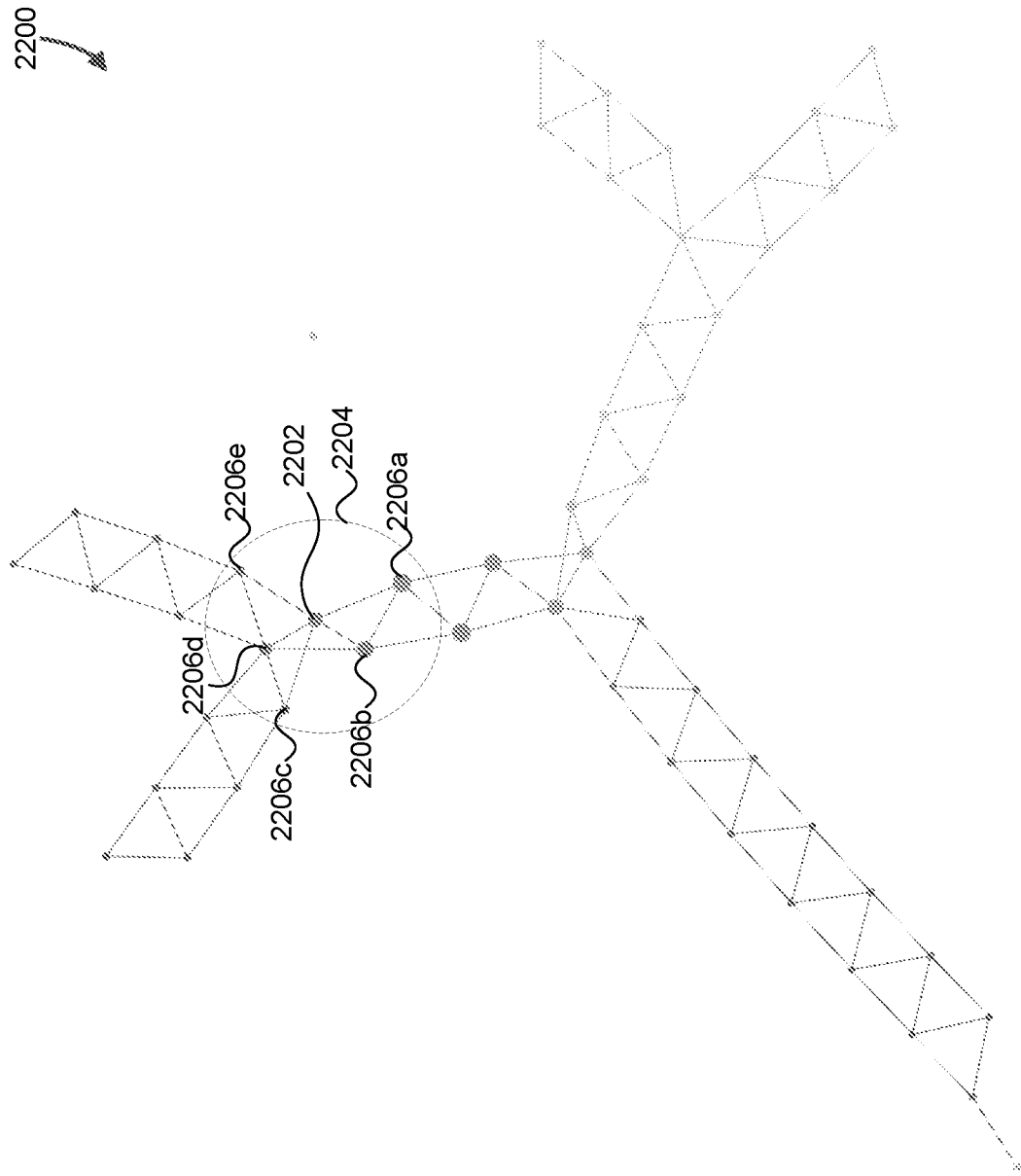


FIG. 22

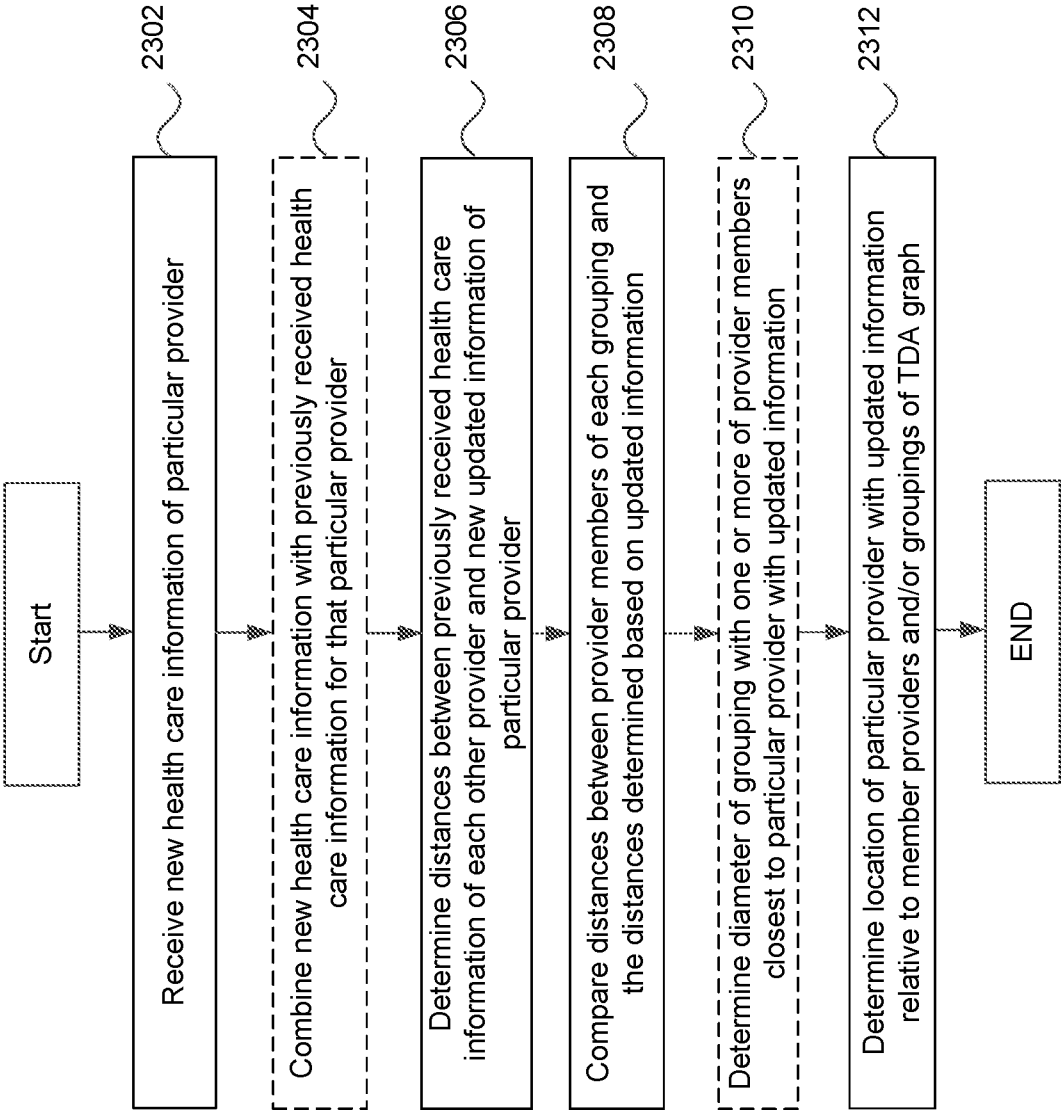


FIG. 23

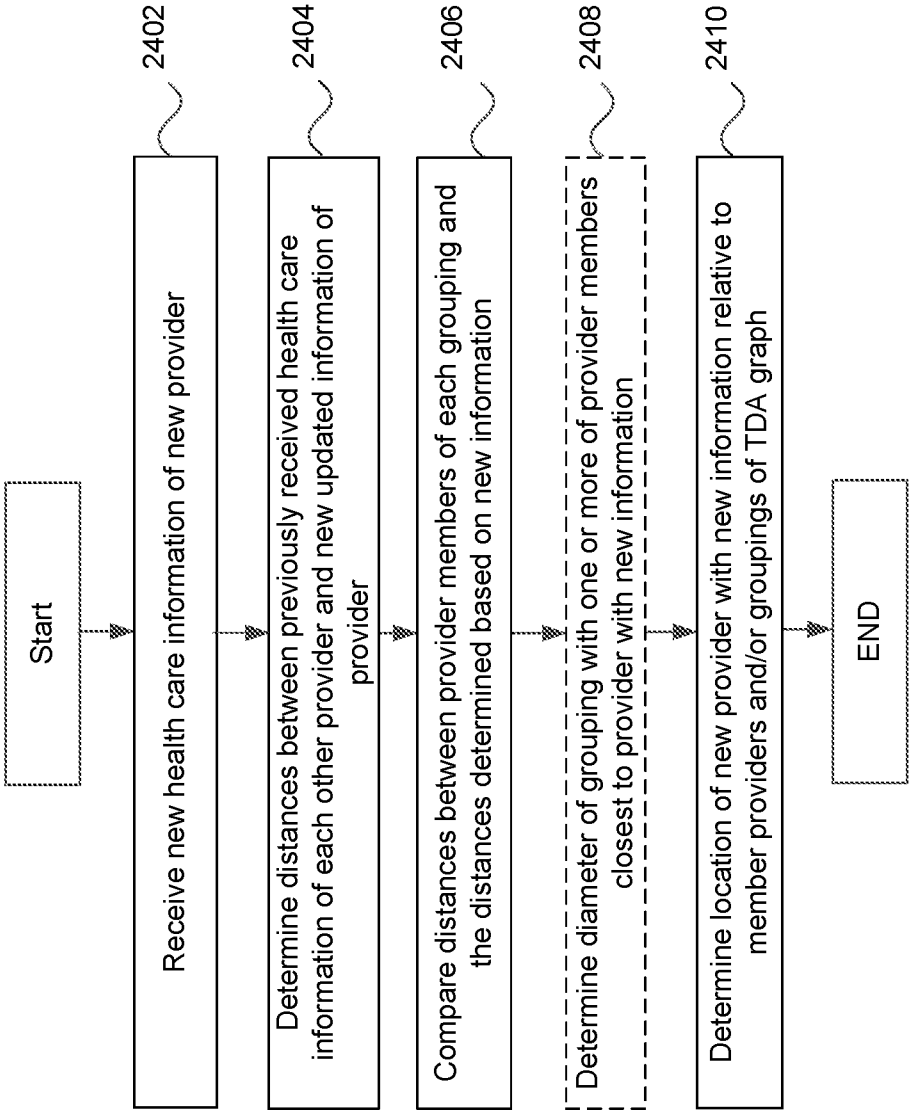


FIG. 24

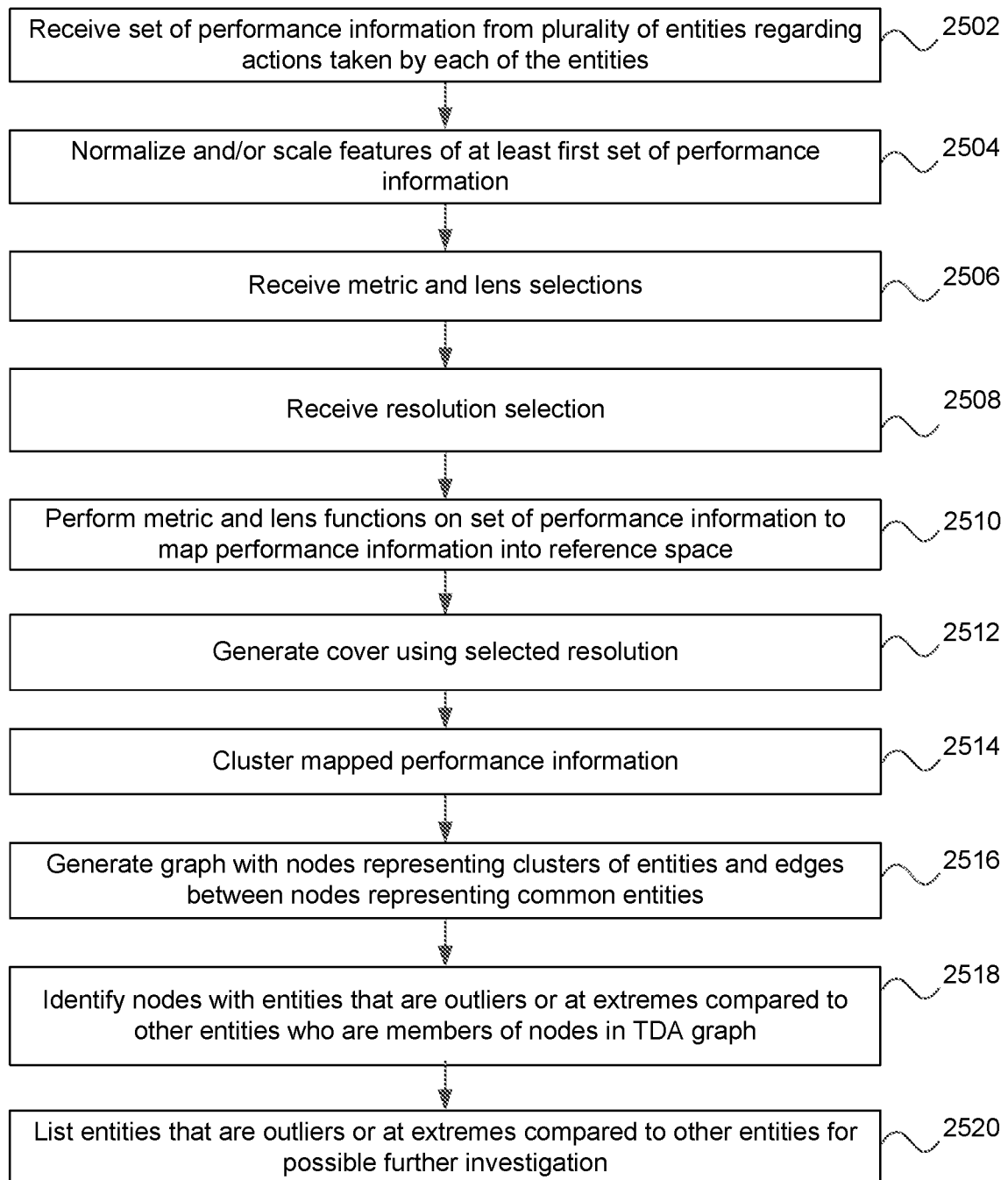


FIG. 25

mpin	eucl_dist_score	feature 1	value 1	feature 2	value 2
1	27.72991511	r_norm_Severity_Frequency_99285	0.784214956	r_norm_Severity_Frequency_99284	0.47051178
2	25.98777708	r_norm_Severity_Frequency_99284	0.650324796	r_norm_Severity_Frequency_99285	0.59131861
3	23.31137135	r_norm_Severity_Frequency_99284	0.741270776	r_norm_Severity_Frequency_99285	0.52811116
4	29.96139388	r_norm_Severity_Frequency_99285	0.654547607	r_norm_Severity_Frequency_99284	0.63140415
5	28.79851434	r_norm_Severity_Frequency_99285	0.835786812	r_norm_Severity_Frequency_99284	0.45001543
6	31.5322629	r_norm_Severity_Frequency_99285	0.741545562	r_norm_Severity_Frequency_99284	0.51378548
7	24.78414565	r_norm_Severity_Frequency_99284	0.656639157	r_norm_Severity_Frequency_99285	0.59947956
8	27.42171931	r_norm_Severity_Frequency_99285	0.746759895	r_norm_Severity_Frequency_99284	0.52852988
9	28.57590682	r_norm_Severity_Frequency_99284	0.741591271	r_norm_Severity_Frequency_99285	0.54401597

FIG. 26

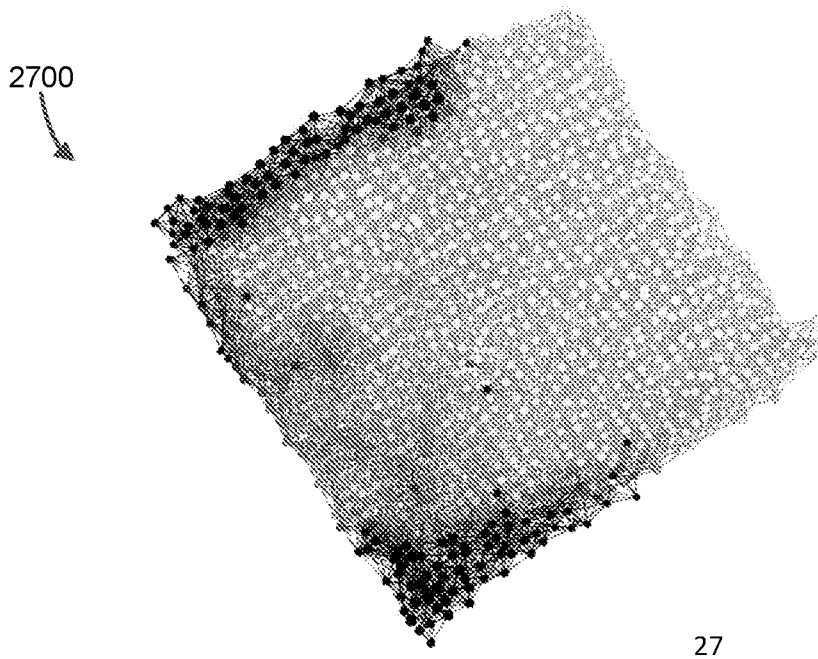


FIG. 27

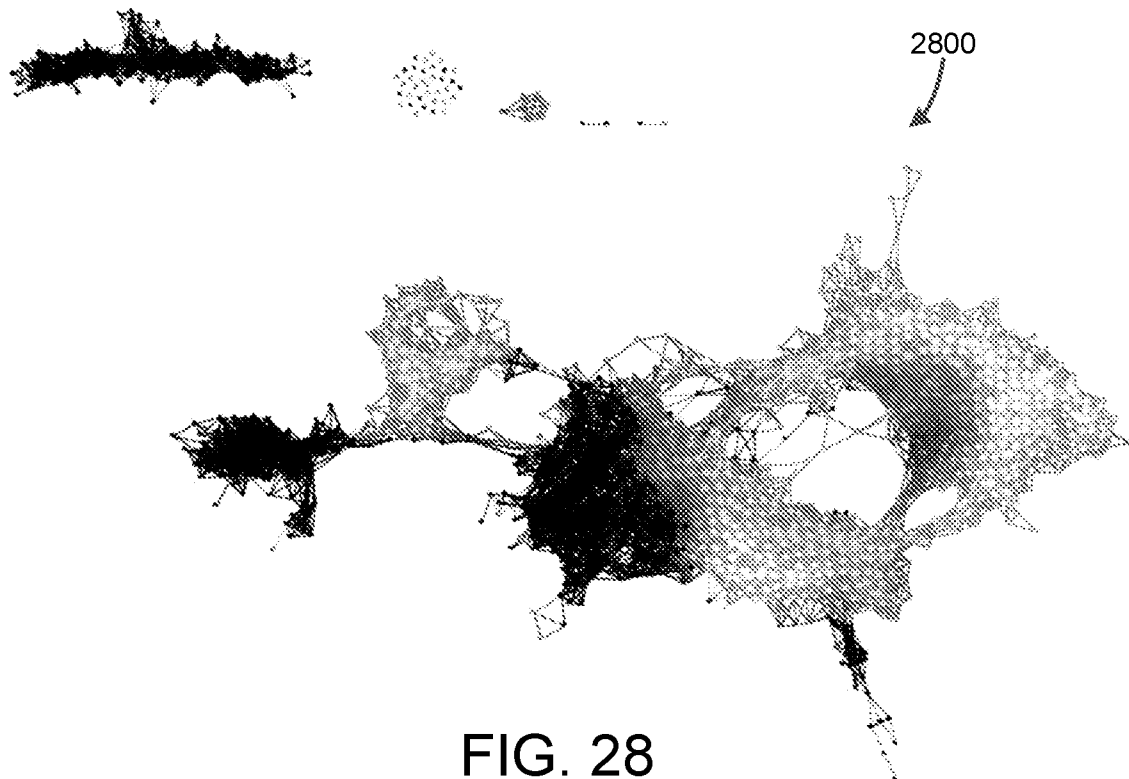


FIG. 28

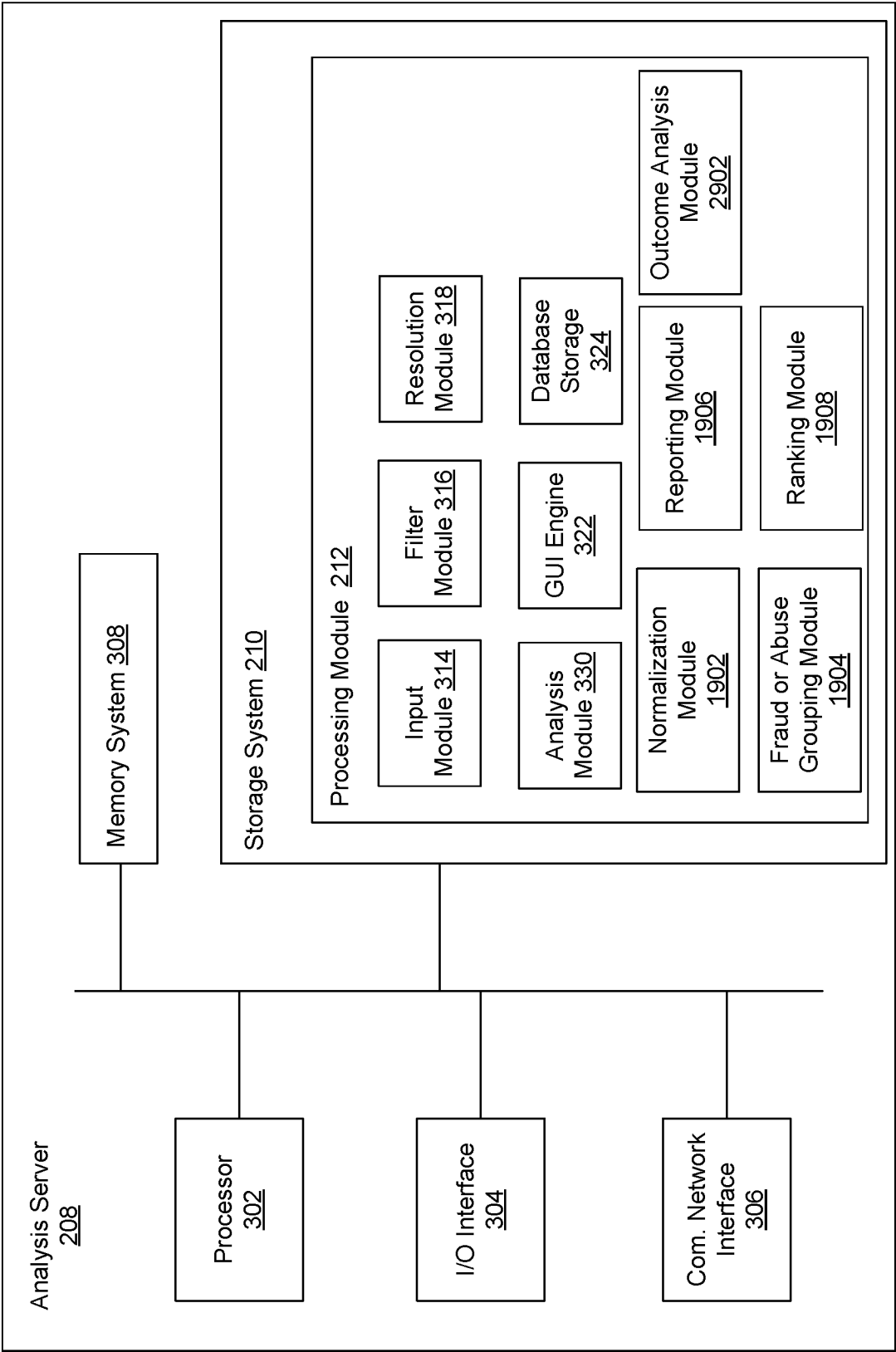


FIG. 29

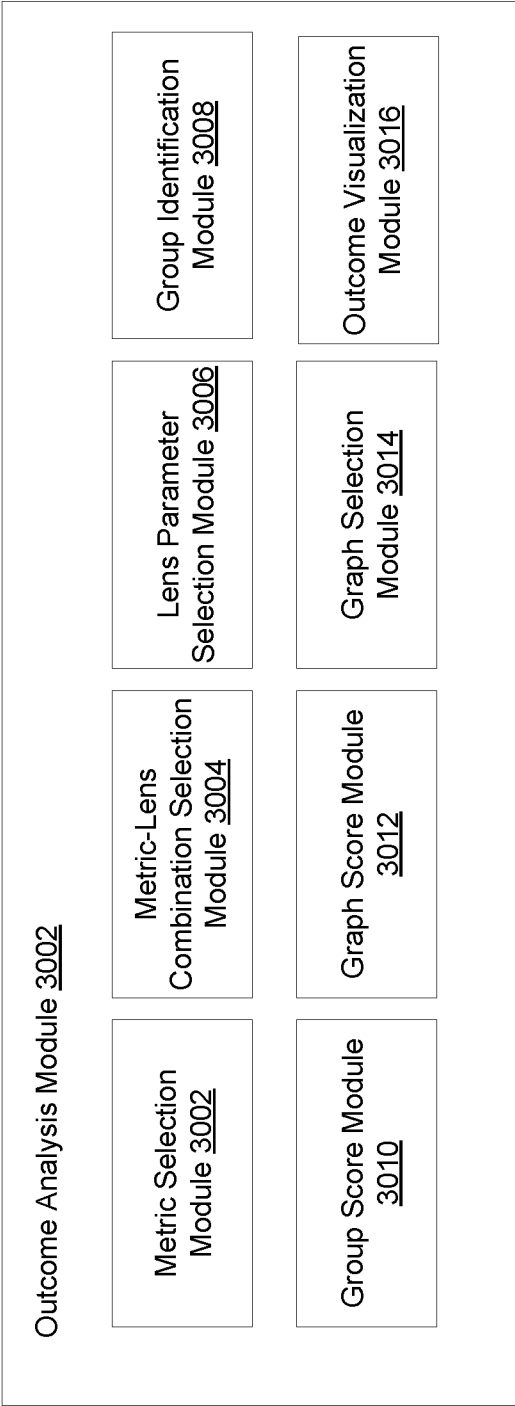


FIG. 30

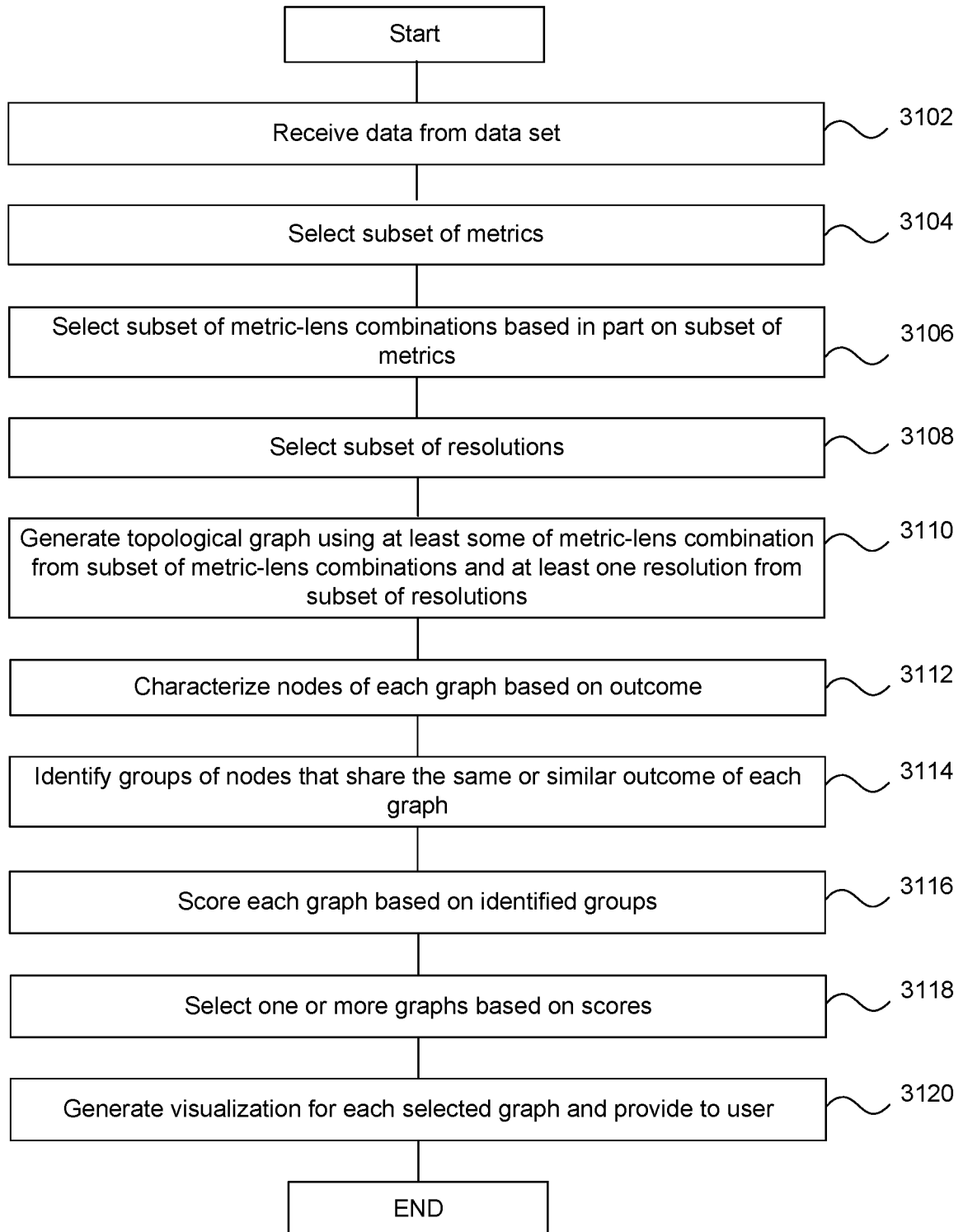


FIG. 31

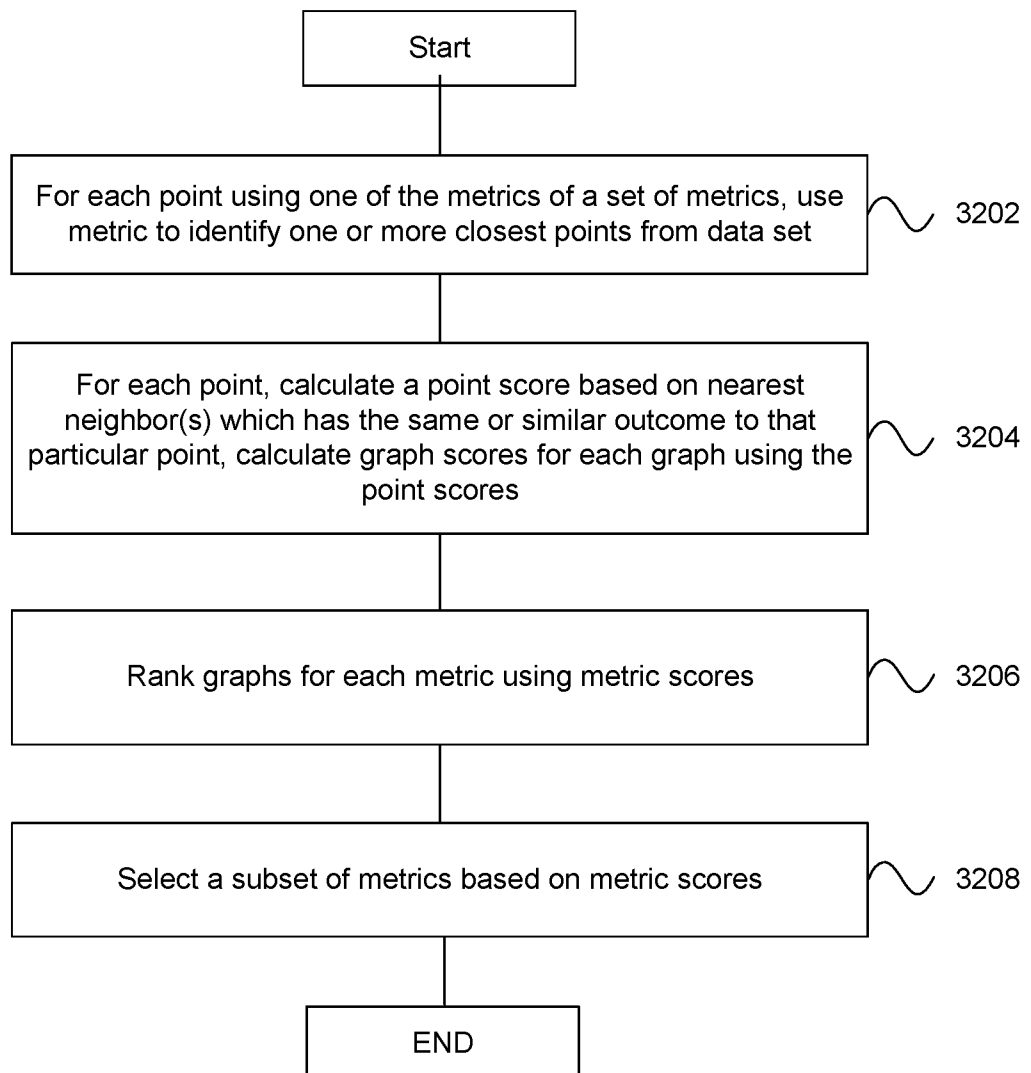


FIG. 32

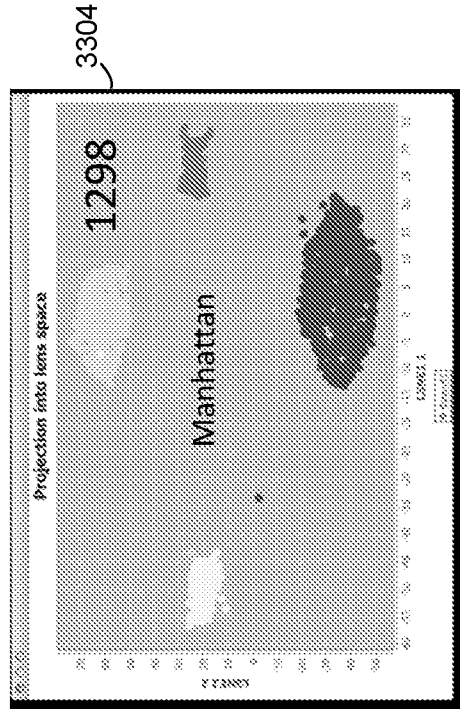


FIG. 33B

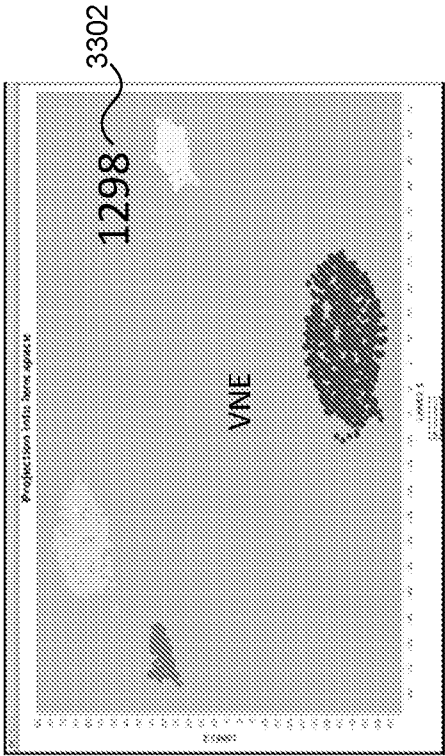


FIG. 33A

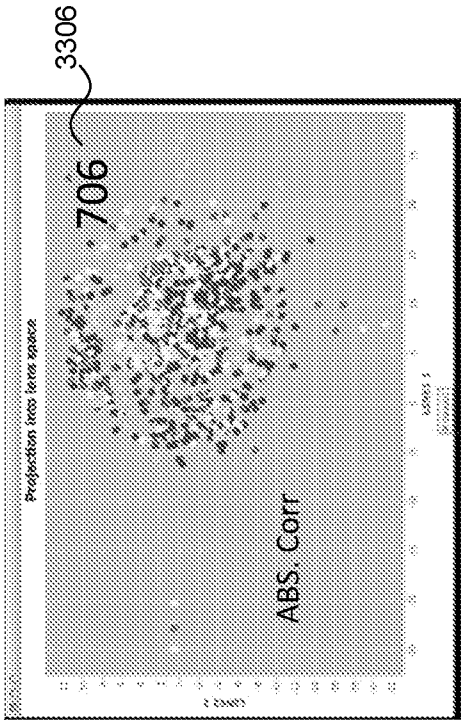


FIG. 33C

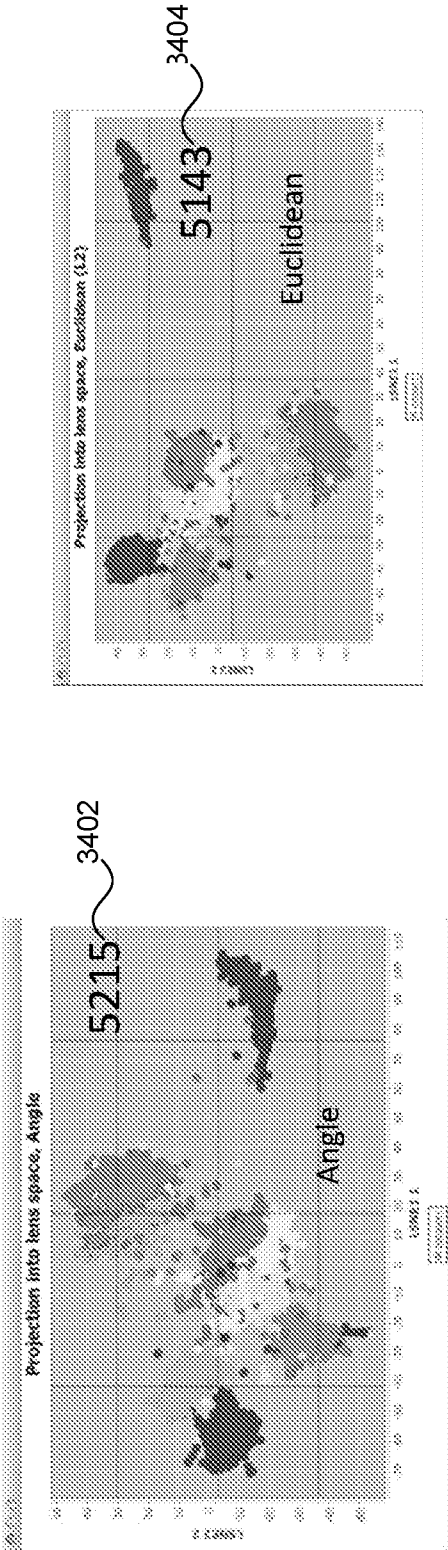


FIG. 34A

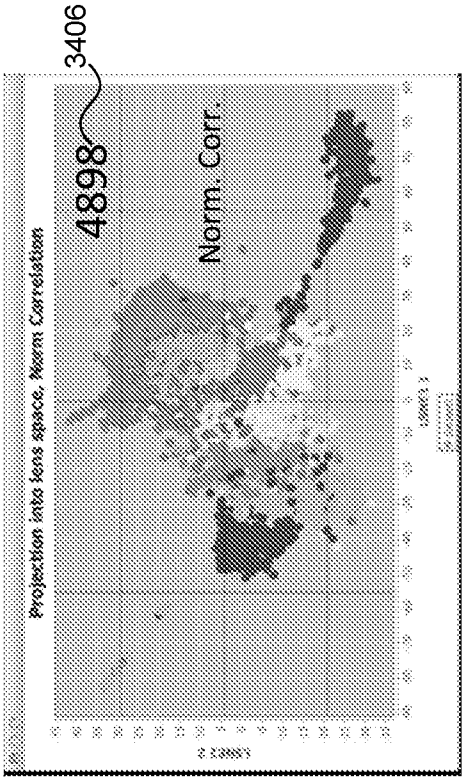


FIG. 34C

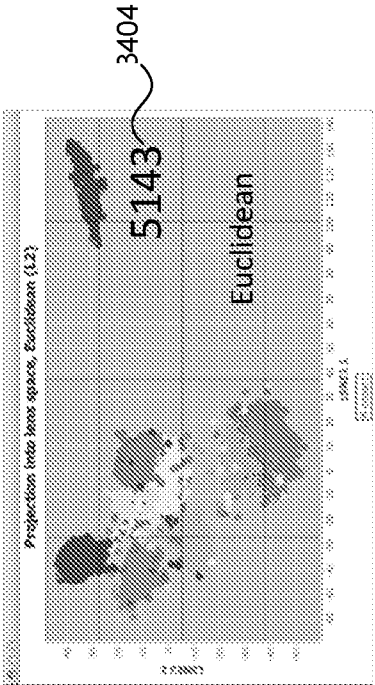


FIG. 34B

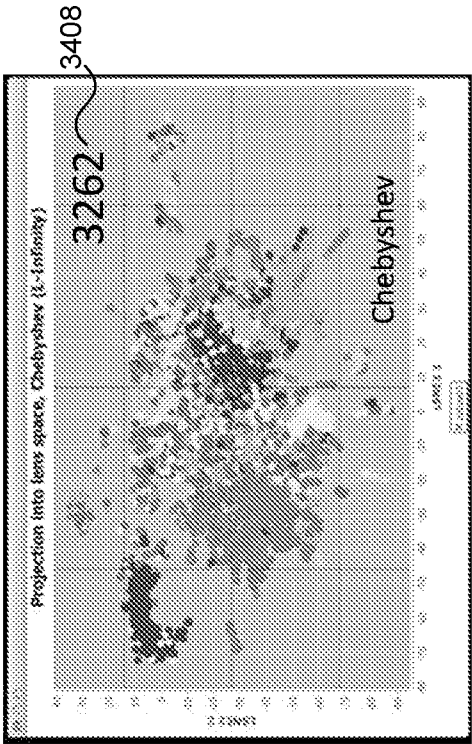


FIG. 34D

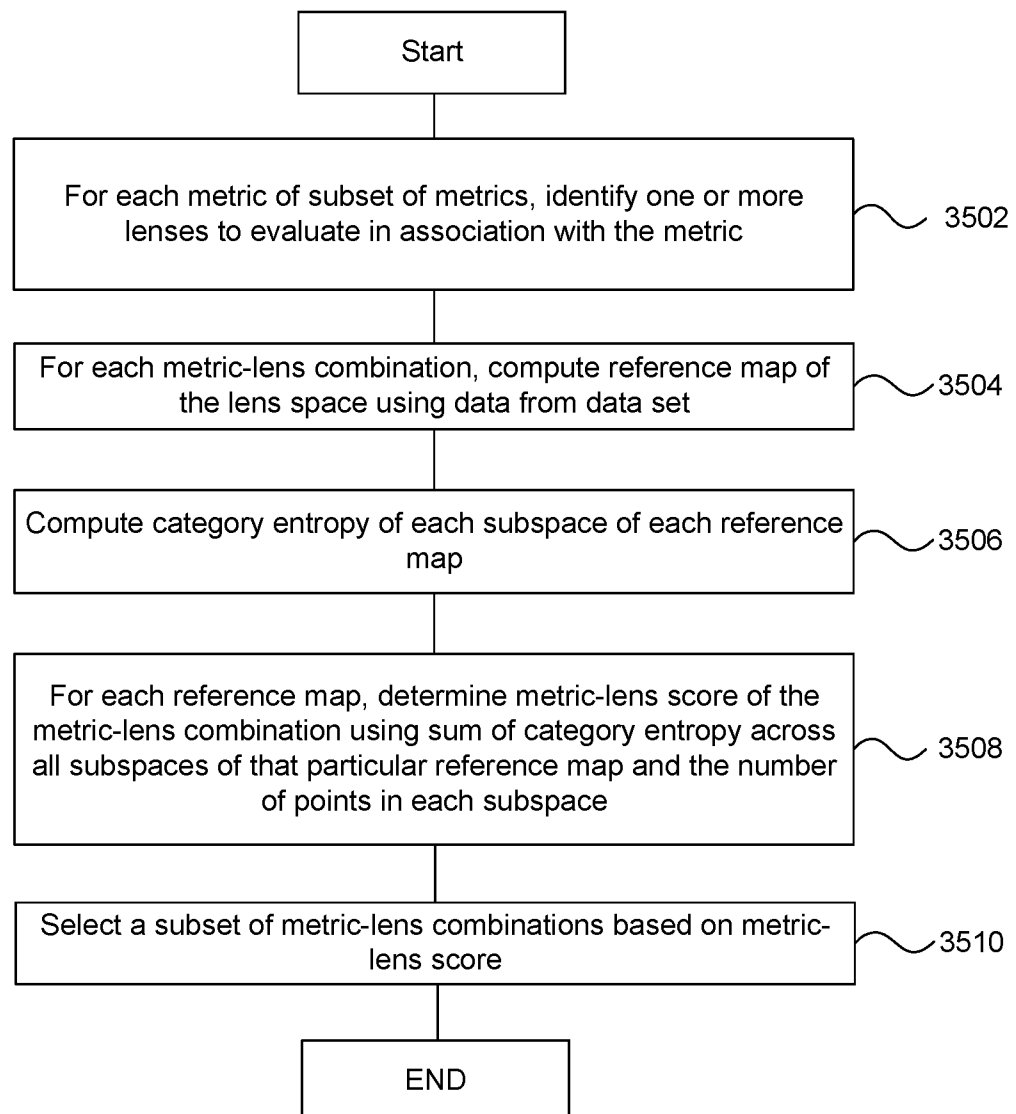


FIG. 35

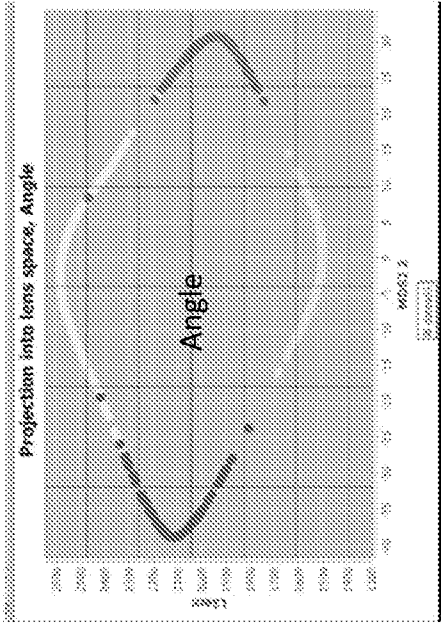


FIG. 36A

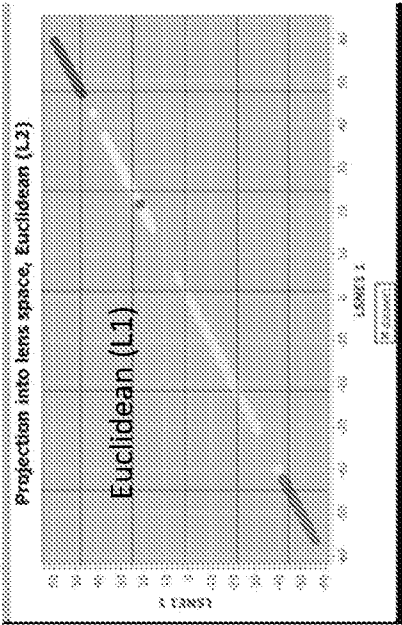


FIG. 36B

FIG. 36C

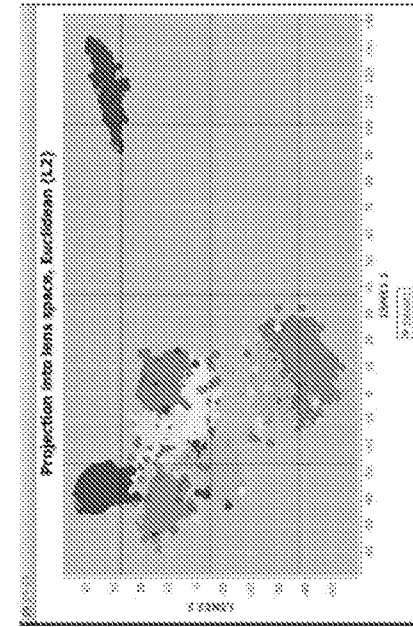


FIG. 37B

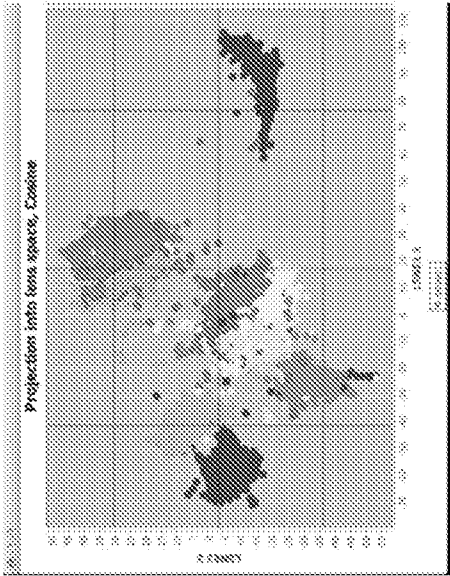


FIG. 37A

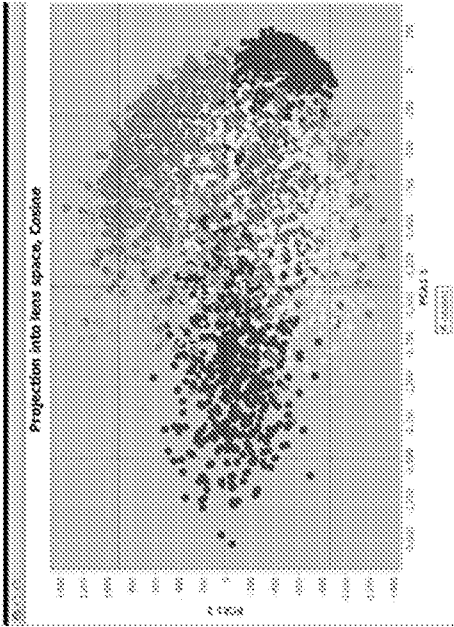


FIG. 37C

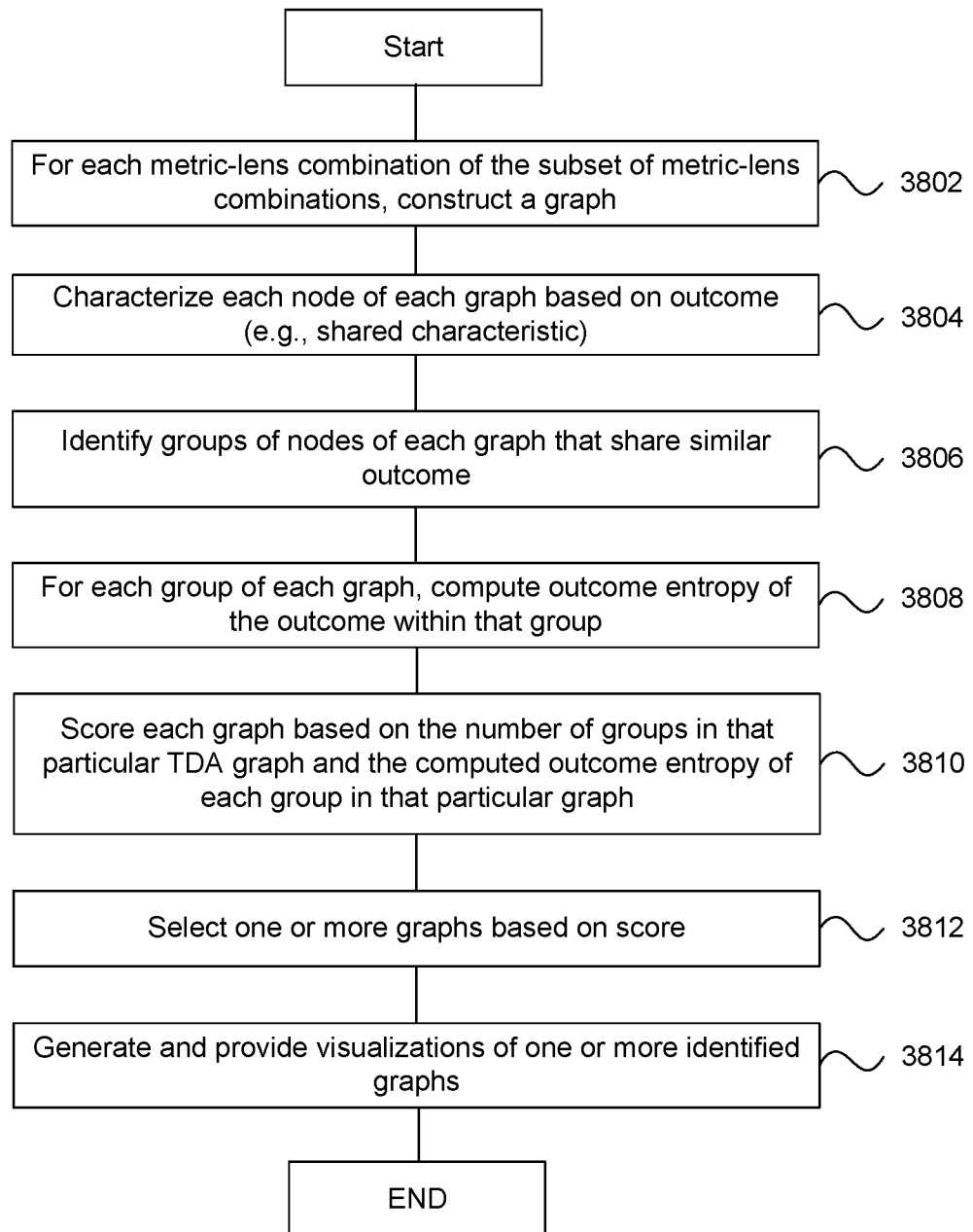


FIG. 38

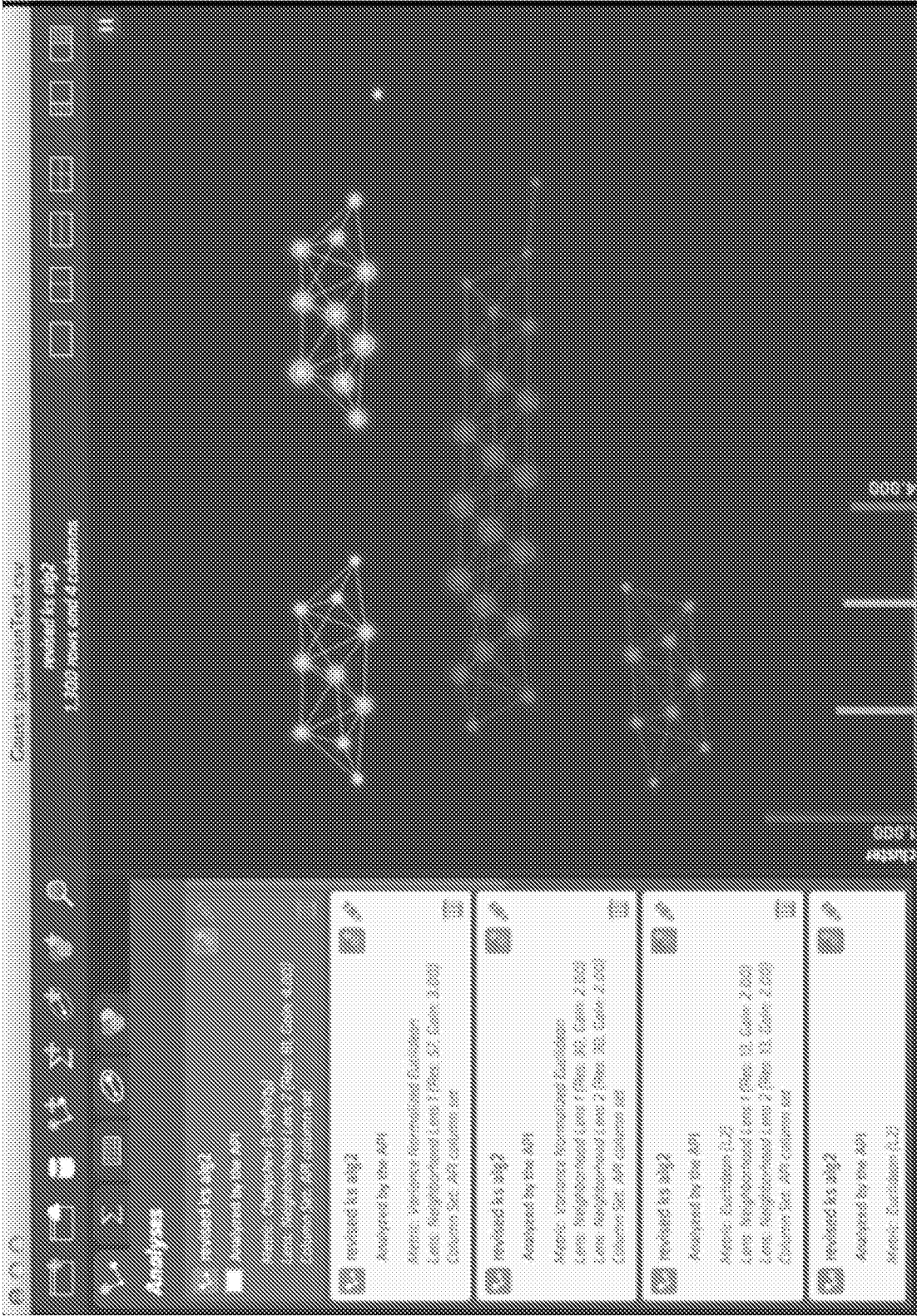


FIG. 39A

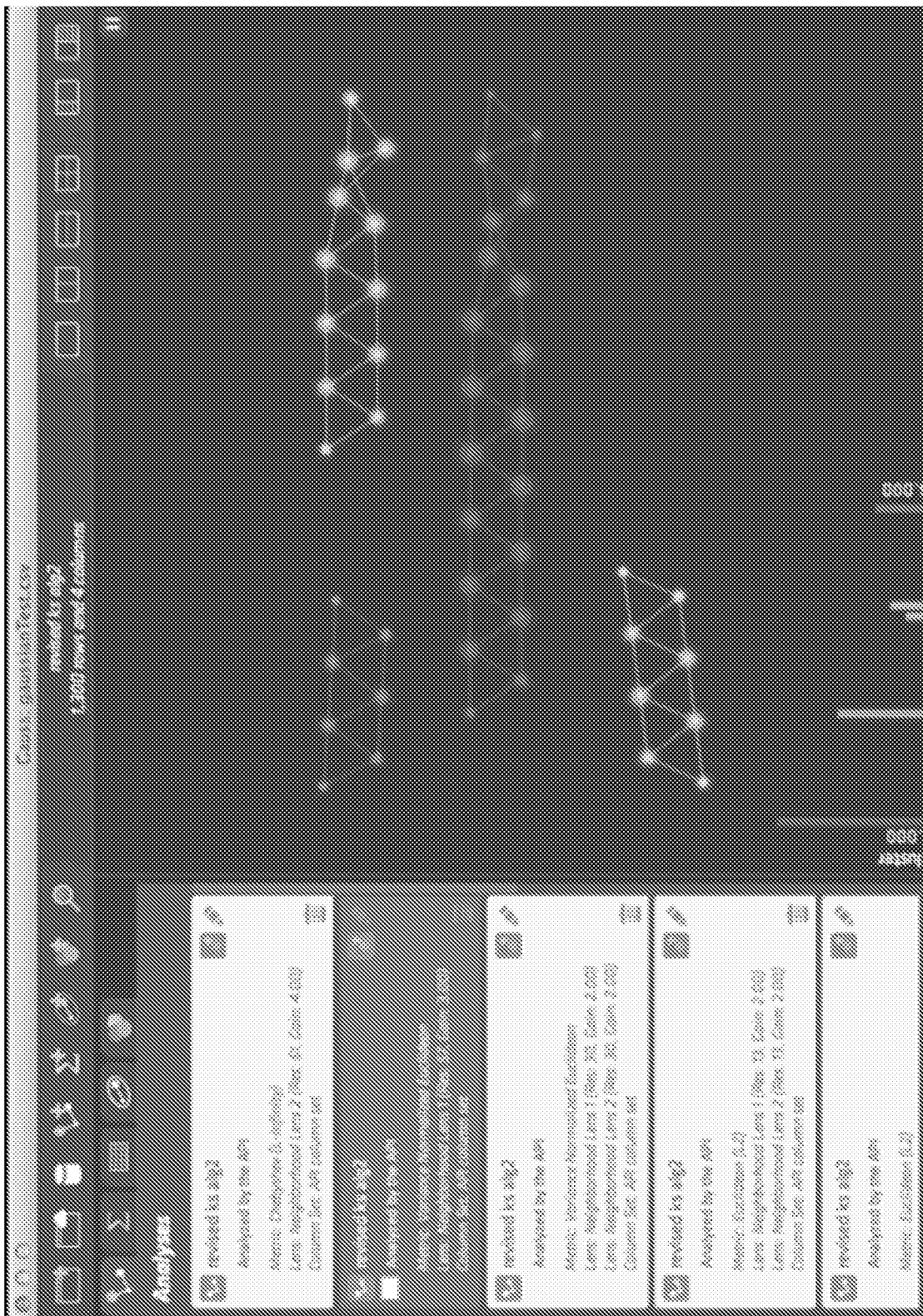


FIG. 39B

INTERNATIONAL SEARCH REPORT

International application No.

PCT/US16/66233

A. CLASSIFICATION OF SUBJECT MATTER

IPC - G06F 21/54 (2017.01)

CPC - H04L 41/0631

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)

See Search History document

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

See Search History document

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

See Search History document

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2014/0278479 A1 (PALANTIR TECHNOLOGIES, INC.) 18 September 2014; abstract; paragraphs [0019]-[0022], [0025], [0032]-[0034], [0041], [0056], [0061], [0062], [0066], [0070]	1-5, 9-14, 18, 19
----		----
Y		6-8, 15-17
Y	US 2013/0101221 A1 (FUJIKI, Y et al.) 25 April 2013; abstract; paragraph [0073]	6, 15
Y	US 2014/0229768 A1 (BERNSTEIN, R et al.) 14 August 2014; abstract; paragraph [0014]	7, 8, 16, 17

☐ Further documents are listed in the continuation of Box C.☐ See patent family annex.

* 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

31 January 2017 (31.01.2017)

Date of mailing of the international search report

07 APR 2017

Name and mailing address of the ISA/

Mail Stop PCT, Attn: ISA/US, Commissioner for Patents
P.O. Box 1450, Alexandria, Virginia 22313-1450

Facsimile No. 571-273-8300

Authorized officer

Shane Thomas

PCT Helpdesk: 571-272-4300
PCT OSP: 571-272-7774