(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
02 April 2020 (02.04.2020)

WIPO | PCT

(10) International Publication Number
WO 2020/068880 A1

(51) International Patent Classification:
C12Q 1/6869 (2018.01) *G06F 19/20* (2011.01)
C12Q 1/6886 (2018.01) *G06F 19/22* (2011.01)

(21) International Application Number:
PCT/US2019/052801

(22) International Filing Date:
24 September 2019 (24.09.2019)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
62/735,349 24 September 2018 (24.09.2018) US

(71) Applicant: **TEMPUS LABS, INC.** [US/US]; 600 West Chicago Ave, Suite 510, Chicago, IL 60654 (US).

(72) Inventors: **SHAH, Kaanan**; 111 E Chestnut St. Apt 14C, Chicago, IL 60611 (US). **HAFEZ, Ashraf**; 118 N. Beverly St., Wheaton, IL 60187 (US). **IGARTUA, Catherine**; 1808 S. Michigan Avenue, Unit 11, Chicago, IL 60616 (US). **MICHUDA, Jackson**; 1250 N. Dearborn 7d, Chicago, IL 60610 (US).

(74) Agent: **KISSLING, Heather, R.** et al.; Marshall, Gerstein & Borun LLP, 233 S. Wacker Drive, 6300 Willis Tower, Chicago, IL 60606-6357 (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, JO, 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,

(54) **Title:** METHODS OF NORMALIZING AND CORRECTING RNA EXPRESSION DATA

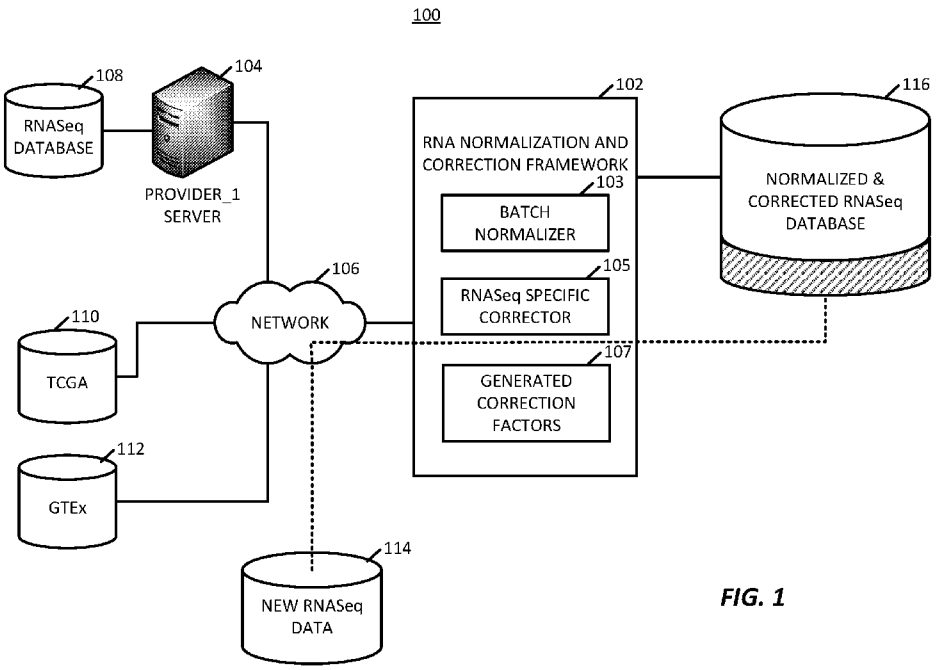


FIG. 1

(57) **Abstract:** A platform to perform normalization and correction on gene expression datasets and combines different datasets into a standard dataset using a framework configured to continuously incorporate new gene expression data. The framework determines a series of conversion factors that are used to on-board new gene expression datasets, such as unpaired datasets, where these conversion factors are able to correct for variations in data type, variations in gene expressions, and variations in collection systems.

[Continued on next page]

WO 2020/068880 A1

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

Published:

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

— *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))*

METHODS OF NORMALIZING AND CORRECTING RNA EXPRESSION DATA

Cross-Reference to Related Applications

[0001] This application claims benefit of priority to and claims under 35 U.S.C. §119(e)(1) the benefit of the filing date of U.S. provisional application serial number 62/735,349 filed September 24, 2018, the entire disclosure of which is incorporated herein by reference.

Field of the Invention

[0002] The present disclosure relates to normalizing and correcting gene expression data and, more particularly, to normalizing and correcting gene expression data across varied gene expression databases.

Background

[0003] The background description provided herein is for the purpose of generally presenting the context of the disclosure. Work of the presently named inventors, to the extent it is described in this background section, as well as aspects of the description that may not otherwise qualify as prior art at the time of filing, are neither expressly nor impliedly admitted as prior art against the present disclosure.

[0004] Experiments examining gene expression are valuable in assessing patient response and projected responses to various treatments. There are relatively large databases of gene expression data, such as The Cancer Genome Atlas (TCGA) project database, the Genotype-Tissue Expression (GTEx) project database, and others. Unfortunately, gene expression data, in particular from RNA sequencing experiments, can be highly sensitive to biases in sample type, sample preparation, and sequencing protocol. The result is gene expression data across databases and data sets that cannot be readily compared, and certainly not if a relatively high level of specificity and sensitivity is required for data analysis. As such, there is a desire for techniques to combine data across gene expression datasets to provide functionally useful and comparable gene expression data.

[0005] For gene expression data in the form of RNA sequencing data (referred to herein as "RNA seq" or "RNAseq" data), for example, main sources of bias are varied. Biases arise from tissue type (e.g., fresh frozen (FF) or formalin fixed, paraffin embedded (FFPE)), and RNA selection method (e.g., exon capture or poly-A RNA selection). For datasets sequenced using exome capture, for example, subtle differences between the different exome capture kits arise upon careful inspection. Examining these biases across multiple RNA seq datasets, it becomes clear that synchronizing RNA seq data is exceedingly challenging.

Summary of the Invention

[0006] The present application presents techniques for normalizing and correcting gene expression data across varied gene expression databases.

[0007] In exemplary embodiments, techniques are provided for normalizing RNA sequence data and for correcting RNA sequence data to establish a uniform gene expression database. The techniques further provide for on-boarding new gene expression data into the uniform gene expression database enriching the new gene expression data for better utilization with existing gene expression data.

[0008] Such techniques provide numerous advantages, including unifying actual gene expression data and parsing that data into different tumor profiles to allow for more accurate analysis of gene expression data, including, for example, greatly reducing database access speeds and data processing times. The present techniques can combine data across gene expression datasets to provide functionally useful and comparable gene expression data that have heretofore been unavailable.

[0009] In accordance with an example, a computer-implemented method includes: generating, from a comparison of a normalized RNA sequence dataset against a standard RNA sequence dataset, at least one conversion factor for applying to a next RNA sequence dataset; and correcting RNA sequence data of the next RNA sequence dataset using the at least one conversion factor.

[0010] In some examples, the computer-implemented method further includes: including the RNA sequence data of the next gene expression dataset into the standard gene expression dataset.

[0011] In some examples, the computer-implemented method includes: obtaining a gene expression dataset comprising the RNA sequence data for one or more genes, normalizing the RNA sequence data using gene length data, guanine-cytosine (GC) content data, and depth of sequencing data; and performing a correction on the RNA sequence data against the standard gene expression dataset by comparing the sequence data for at least one gene in the gene expression dataset to sequence data in the standard gene expression dataset.

[0012] In some examples, such normalization is performed by normalizing the gene length data for at least one gene to reduce systematic bias, normalizing the GC content data for the at least one gene to reduce systematic bias, and normalizing the depth of sequencing data for each sample.

[0013] In some examples, generating the at least one conversion factor includes: for a sample gene, obtaining sample data from the normalized dataset and obtaining sample data from the standard gene expression dataset; determining a statistical mapping between the sample data of the

normalized dataset and the sample data of the standard gene expression dataset; and determining the at least one conversion factor using the statistical mapping.

[0014] In some examples, determining the statistical mapping includes determining a linear mapping model between the sample data of the normalized dataset and the sample data of the standard gene expression dataset.

[0015] In some examples, the computer-implemented method includes: determining an intercept and a beta value for the linear mapping model; and determining the at least one conversion factor using the statistical mapping from the intercept and the beta value.

[0016] In accordance with another example, a computing device comprising one or more memories and one or more processors is configured to: generate, from a normalization of an RNA sequence data against a standard RNA sequence dataset, at least one conversion factor for applying to a next RNA sequence dataset; and correct RNA sequence data of the next RNA sequence dataset using the at least one conversion factor.

[0017] In some examples, the computing device is configured to include the corrected RNA sequence data of the next RNA sequence dataset into the standard RNA sequence dataset.

[0018] In some examples, the computing device is configured to: obtain a gene expression dataset comprising the RNA sequence data for one or more genes, the RNA sequence data including gene length data, guanine-cytosine (GC) content data, and/or depth of sequencing data; and normalize the RNA sequence data to remove systematic known biases.

[0019] In some examples, the computing device is configured to: normalize the gene length data for the one or more genes to reduce systematic bias; normalize the GC content data for the one or more genes to reduce systematic bias; and normalize the depth of sequencing data for the RNA sequence data.

[0020] In some examples, the computing device is configured to: for a sample gene, obtain sample data from a normalized RNA sequence dataset and obtaining sample data from the standard RNA sequence dataset; determine a statistical mapping between the sample data of the normalized RNA sequence dataset and the sample data of the standard RNA sequence dataset; and determine the at least one conversion factor using the statistical mapping.

[0021] In some examples, the computing device is configured to: determine an intercept and a beta value for the linear mapping model; and determine the at least one conversion factor using the statistical mapping from the intercept and the beta value.

[0022] In accordance with another example, a computer-implemented method includes: generating, from a normalization of gene expression data against another gene expression dataset,

at least one conversion factor for applying to a next gene expression dataset; and correcting gene sequence data of the next gene expression dataset using the at least one conversion factor.

[0023] In accordance with an example, a computer-implemented method comprises: receiving, at one or more processors, a gene expression dataset; identifying within the gene expression dataset, using a regression technique implemented by the one or more processors, gene expression data having multiple modal expression peaks; for the gene expression data, normalizing, using the one or more processors, a spacing between each of the multiple model expression peaks to form a normalized gene expression data; and storing the normalized gene expression data in a normalized gene expression dataset.

[0024] In accordance with another example, a computer-implemented method comprises: receiving, at one or more processors, a RNA sequence dataset; identifying within the gene expression dataset, using a regression technique implemented by the one or more processors, a plurality of RNA expression data each having a bimodal distribution comprising two expression peaks; for each of the plurality of RNA expression data, normalizing, using the one or more processors, a spacing between the two expression peaks such that each of the plurality of RNA expression data has the same spacing between the two expression peaks; and storing the normalized RNA expression data in a normalized RNA sequence dataset.

Brief Description of the Drawings

[0025] The figures described below depict various aspects of the system and methods disclosed herein. It should be understood that each figure depicts an example of aspects of the present systems and methods.

[0026] FIG. 1 is a schematic illustration of an example computer processing system for normalizing and correcting RNA expression data, in accordance with an example.

[0027] FIG. 2 illustrates an example computing device for implementing the systems of FIG. 1 and the processes of FIGS. 3-6, in accordance with an example.

[0028] FIG. 3 is a block diagram of an example process for updating a standard gene expression dataset with new gene expression data using one or more conversion factors, in accordance with an example.

[0029] FIG. 4 is a block diagram of an example process for performing RNA seq normalization, as may be implemented by the process of FIG. 5, in accordance with an example.

[0030] FIG. 5 is a block diagram of an example process for generating conversion factors and performing RNA seq correction on unpaired normalized RNA datasets, produced by the process of FIG. 4, in accordance with an example.

[0031] FIG. 6 is a block diagram of an example process for performing correction on a paired dataset formed from multiple normalized gene expression datasets, in accordance with an example.

[0032] FIG. 7 is a schematic illustration of an RNA normalization and correction framework interfacing to provide normalized and corrected RNA seq data to a multi-model RNA seq, imaging features machine learning framework, and/or other framework, in accordance with an example.

[0033] FIG. 8 illustrates an example workflow for implementing a two-step normalization process and correction process on an external gene expression dataset, in accordance with an example.

[0034] FIG. 9 is a plot of a linear mapping performed on gene expression data to identifier outliers to clean from the gene expression dataset, in accordance with an example.

[0035] FIG. 10 is a plot of a linear mapping performed on an external gene expression data to identify outliers to clean from the external gene expression dataset, in accordance with an example.

[0036] FIG. 11A is a plot of an uncorrected exemplary gene expression dataset. FIG. 11B is a plot of the exemplary gene expression dataset normalized and corrected, in accordance with an example.

[0037] FIG. 12 is a schematic illustration of an multimodal normalization framework for normalization of gene expression data, in accordance with an example.

[0038] FIG. 13 is a plot of RNA expression data for ESR1, where that data exhibits a bimodal distribution prior to normalization, in accordance with an example.

[0039] FIG. 14 is a block diagram of an example process for performing correction a multimodal gene expression data, such as the bimodal data of FIG. 13, in accordance with an example.

[0040] FIG. 15 is a plot of the RNA expression data for ESR1 from FIG. 13, after an intra-peak normalization of the bimodal distribution, in accordance with an example.

[0041] FIG. 15 is a plot of the normalized RNA expression data for ESR1 from FIG. 15, after a shift of the normalized peaks is performed, in accordance with an example.

[0042] FIG. 17A is a plot of bimodal RNA expression data for multiple different genes (AR, PGR, ESR1, ERBB2), prior to normalization, and FIG. 17B is a plot of that RNA expression data after normalization showing aligned peaks for the bimodal distributions of the data, in accordance with an example.

[0043] FIG. 18 is a plot of the RNA expression data for TP53 which exhibits a uni-modal distribution, showing a bimodal peak normalization process in progress, in accordance with an example.

[0044] FIG. 19A is a plot of uni-modal RNA expression data for multiple different genes (BRCA1, BRCA2, PIK3CA), prior to normalization, and FIG. 19B is a plot of that uni-modal RNA expression data

after normalization showing alignment resulting a bimodal peak normalization process, in accordance with an example.

[0045] FIG. 20A is a plot of Uniform Manifold Approximation and Projection (UMAP) of ESR1 gene expression data prior to normalization, and FIG. 20B is a plot of UMAP for that RNA expression data after normalization, in accordance with an example.

[0046] FIG. 21 displays a UMAP plot of a cohort of patient specimens with one cluster of triple negative breast cancer specimens.

Detailed Description

[0047] The present application presents a platform for performing normalization and correction on gene expression datasets to allow for combining of different datasets into a standardized dataset, such as a previously normalized dataset, that may continuously incorporate new data. The present techniques generate a series of conversion factors that are used to on-board new gene expression datasets, such as unpaired datasets, where these conversion factors are able to correct for variations in data type, variations in gene expressions, and variations in collection systems. For example, conversion factors are able to correct against data collection bias, variations in laboratory data generation processes, variations in data sample size, and other factors that can cause incongruity between datasets. The techniques may correct older datasets for inclusion into new dataset. For example, existing, stable datasets, such as the TCGA (https://portal.gdc.cancer.gov/) or GTEx (https://gtexportal.org/home/), may be corrected to match new datasets. Examples of RNA seq datasets include RNAseq data from FFPE tissue, RNAseq data from fresh frozen tissue, or from other tissue from which RNA seq data may be extracted. Datasets may come from laboratories (such as Tempus Labs, Inc., Chicago, IL), from individual research institutions (such as the Michigan Center for Translational Pathology, Ann Arbor, MI), from public data repositories such as TCGA and GTEx, or from other sources.

[0048] The present techniques include platforms for normalization of gene expression data, such as RNA sequence data or array-based technologies data, and comparison of gene expression data to a standard gene expression dataset. The present techniques include platforms for generating one or more conversion factors by comparing gene expression data to such standard gene expression datasets. The present techniques include correcting gene expression data, such as RNA sequence data, of subsequent gene expression datasets using these one or more conversion factors, thereby allowing subsequent gene expression datasets to be integrated into the standard gene expression dataset.

[0049] In some examples, the present techniques include obtaining a gene expression dataset having RNA sequence data for one or more genes, where that RNA sequence data includes gene

length data, guanine-cytosine (GC) content data, and depth of sequencing data. In other examples, other types of gene expression datasets from array-based technologies, such as RNA microarrays, may be obtained. The techniques may include performing normalization of the RNA sequence data or other gene expression datasets. The normalization may include normalizing the gene length data for at least one gene to reduce systematic bias, normalizing the GC content data for the at least one gene to reduce systematic bias, and normalizing the depth of sequencing data for each sample, for example. The normalized dataset may be compared against the standard gene expression dataset by comparing the sequence data for at least one gene in the gene expression dataset to sequence data in the standard gene expression dataset to generate at least one conversion factor.

[0050] FIG. 1 illustrates a system 100 for normalizing and correcting gene expression data, such as RNA seq data. A normalization and correction framework 102 is coupled to receive gene expression data from a multitude of different sources through a communication network 106. The framework 102, for example, may be coupled to a health care provider computing system 104, such as a research institution computing system, lab computing system, hospital computing system, physician group computing system, etc., that makes available stored gene expression data in the form of RNA sequencing dataset 108. Other gene expression network-accessible datasets are also coupled to the network 106, including the Cancer Genome Atlas (TCGA) dataset 110 and the Genotype-Tissue Expression (GTEx) dataset 112, both examples of established gene expression datasets that can be normalized and corrected to be incorporated into a already-normalized and corrected growing database of gene expression data.

[0051] The framework 102 includes a batch normalizer 103 configured to perform gene expression batch normalization processes in accordance with examples herein, processes that adjust for known biases within the dataset including, but not limited to, GC content biases, gene length biases, and sequencing depth biases. In the example of FIG. 1, the framework 102 is further configured to perform gene expression correction processes in accordance with examples herein using a RNA seq corrector 105. As discussed herein, the processes of the normalizer 103 and the corrector 105 are used by the framework 102 to normalize gene expression data and generate one or more correction factors (107), which are stored in the framework 102 and applied by the framework 102 to convert new gene expression datasets, such as dataset 114. Applying these correction factors to the new dataset 114, for example, the framework 102 is able to normalize, correct, and convert that dataset 114 into a format for integration into an existing normalized, corrected gene expression dataset 116, as shown.

[0052] The framework 102 may be implemented on a computing device such as a computer, tablet or other mobile computing device, or server. The framework 102 may be implemented by any number of processors, controllers or other electronic components for processing or facilitating the

RNA sequencing data analyses. In some examples, the system 100 is implemented in a broader system that includes processing and hardware for imaging feature analysis, such as analyzing features in medical imaging data, immune infiltration data analysis, DNA sequencing data analysis, organoid development analysis, and/or other modality analyses.

[0053] An example computing device 200 for implementing the framework 102 is illustrated in FIG. 2. As illustrated, the framework 102 may be implemented on the computing device 200 and in particular on one or more processing units 201, which may represent Central Processing Units (CPUs), and/or on one or more or Graphical Processing Units (GPUs), including clusters of CPUs and/or GPUs. The framework 102 may be configured to perform processes of the techniques herein, such as those described with reference to FIGS. 3-9. Features and functions described for the framework 102 may be stored on and implemented from one or more non-transitory computer-readable media 203 of the computing device 200. The computer-readable media 203 may include, for example, an operating system 205 and the framework 102. More generally, the computer-readable media may store the batch normalizer 103 for executing batch normalization process instructions, a gene expression corrector (e.g., the RNASeq specific inspector 105) for executing gene expression process instructions, and the generated correction factors 107. The computing device 200 may be a distributed computing system, such as an Amazon Web Services cloud computing solution.

[0054] The computing device 200 includes a network interface 210 communicatively coupled to the network 106, for communicating to and/or from a portable personal computer, smart phone, electronic document, tablet, and/or desktop personal computer, or other computing devices. The computing device further includes an I/O interface 212 connected to devices, such as digital displays 214, user input devices 216, etc. The computing device 200 may be connected to gene expression databases 108 through network 106, as well as the normalized and corrected gene expression database 116. In some examples. A database 218 within the computer device 200 may be used to store gene expression data, including new gene expression data for normalization and correction, normalized and corrected gene expression data, or other data. A graphic user interface (GUI) generator 220 is provided for generating digital reports, user interfaces, etc. for allowing users to interact with the normalized and corrected gene expression databases.

[0055] The functions of the framework 102 may be implemented across distributed devices 202, 204, etc. connected to one another through a communication link. In other examples, functionality of the system 100 may be distributed across any number of devices, including the portable personal computer, smart phone, electronic document, tablet, and desktop personal computer devices shown. The server 200 may be communicatively coupled to the network 106 and another network 206. The networks 106/206 may be public networks such as the Internet, a private network such as

that of research institution or a corporation, or any combination thereof. Networks can include, local area network (LAN), wide area network (WAN), cellular, satellite, or other network infrastructure, whether wireless or wired. The networks can utilize communications protocols, including packet-based and/or datagram-based protocols such as Internet protocol (IP), transmission control protocol (TCP), user datagram protocol (UDP), or other types of protocols. Moreover, the networks can include a number of devices that facilitate network communications and/or form a hardware basis for the networks, such as switches, routers, gateways, access points (such as a wireless access point as shown), firewalls, base stations, repeaters, backbone devices, etc.

[0056] The computer-readable media may include executable computer-readable code stored thereon for programming a computer (e.g., comprising a processor(s) and GPU(s)) to the techniques herein. Examples of such computer-readable storage media include a hard disk, a CD-ROM, digital versatile disks (DVDs), an optical storage device, a magnetic storage device, a ROM (Read Only Memory), a PROM (Programmable Read Only Memory), an EPROM (Erasable Programmable Read Only Memory), an EEPROM (Electrically Erasable Programmable Read Only Memory) and a Flash memory. More generally, the processing units of the computing device 200 may represent a CPU-type processing unit, a GPU-type processing unit, a field-programmable gate array (FPGA), another class of digital signal processor (DSP), or other hardware logic components that can be driven by a CPU.

[0057] FIG. 3 illustrates a process 300 that may be executed by the system 100. Gene expression data is obtained from a gene expression database or data source, at process 302. In the example of RNA seq data, the dataset may be obtained from a high throughput sequencer, such as Illumina HiSeq, Illumina NextSeq, Illumina NovaSeq, or other high throughput sequencing machines. The framework normalizes the newly obtained gene expression dataset, at process 304, to eliminate biases caused by, for example, GC content, gene length, and sequencing depth. Conversion factors are generated by comparing the obtained gene expression dataset to a standard gene expression dataset using a statistical mapping model, at process 306. Examples of statistical mapping model include, but are not limited to, a standard linear model, a generalized linear model (using for example a gamma distribution of counts data), or non-parametric methods, such as data transformation into ranks. The generated conversion factors are stored by the framework, as a result. At process 308, conversion factors are applied to the new gene expression data, which is then integrated into the standard dataset, in this converted form, at process 308.

[0058] FIG. 4 illustrates an example normalization process 400 to be applied to received raw gene expression data. A gene expression dataset is obtained, e.g., from a network accessible database connected to the network 106. The selection may be manual, by an operator using a graphical user interface provided to a display. The selection may be automated, such as when pre-determined

search data is accessed by the system 100 and used to find corresponding data in the gene expression dataset (process 402). The gene expression dataset may contain RNA seq data, e.g., the TCGA, GTEx, or other database. The gene expression data may be array-based data, in other examples.

[0059] A gene information table comprising information such as gene name and starting and ending points (to calculate gene length) and gene GC content, is accessed and the resulting information is used to determine sample regions (process 404) for analyzing the gene expression datasets. A GC content normalization process 406 is performed using a first full quantile normalization process, e.g., a quantile normalization process like that of the R packages EDASeq and DESeq normalization processes (https://bioconductor.org/packages/release/bioc/html/DESeq.html) may be used. In an example, a 10 quantile bin normalization is performed. The GC content for the sampled data is then normalized for the gene expression dataset. Subsequently, a second, full quantile normalization (e.g., using 10 quantile bins) is performed on the gene lengths in the sample data, at process 408.

[0060] To correct for sequencing depth, a third normalization process 410 may be used that allows for correction for overall differences in sequencing depth across samples, without being overly influenced by outlier gene expression values in any given sample. In exemplary embodiments, at a process 412, a global reference is determined by calculating a geometric mean of expressions for each gene across all samples. In other examples the reference geometric mean is obtained from the gene information table based on the existing datasets (e.g., GTEx, TCGA, etc.).

[0061] The size factor is used to adjust the sample to match the global reference. In operation, a sample's expression values are compared to a global reference geometric mean (process 412), creating a set of expression ratios for each gene (i.e., sample expression to global reference expression). At a process 414, a size factor is determined as the median value of these calculated ratios. The sample is then adjusted by the single size factor correction in order to match to the global reference, e.g., by dividing gene expression value for each gene the sample's size factor, at a process 416.

[0062] In the illustrated example, after normalization, log transformation is performed on the RNA seq data for each gene, at a process 418. The entire GC normalized, gene length normalized, and sequence depth corrected RNA seq data is stored as normalized RNA seq data, at process 420.

[0063] Each of the normalizations for process 400 may be perform in sequential manner, where the output of one process provides input data to the next subsequent process. The particular ordering of the normalizations, however, is not important, as any of the three normalization processes may be performed in any order. Furthermore, alternative normalization methods can be

applied, including but not limited to, Fragments Per Kilobase Million (FPKM), Reads Per Kilobase Million (RPKM), Transcripts Per Kilobase Million (TPM), and 3rd quartile normalization.

[0064] In some examples, an objective of the present techniques is to combine RNA seq data across many different datasets, overcoming the technical differences in sample collection methods used by many labs today. As noted above, different sources of bias can affect RNA seq datasets, these include biases based on tissue type, e.g., fresh, frozen or formalin fixed, paraffin embedded (FFPE). Other biases arise from selection method, e.g., exon capture or poly-A RNA selection. Even for datasets sequenced using exome capture, subtle differences between different exome capture kits can affect datasets.

[0065] In order to correct for these biases, the system 100 may perform a correction after normalization for samples sequenced and obtained from external sources, e.g., network accessible databases, 108, 110, 112, and 114, for example. For each of these different databases a per-gene correction factor may be developed so that samples across datasets can be compared and analyzed for correction and integration into a normalized, corrected gene expression dataset.

[0066] FIG. 5 illustrates an example correction process 500 to be applied to the normalized RNA seq data produced by the process 400. For the illustrated example, in order to calculate the per gene correction factors, an equal number of samples, N, was obtained from two datasets, the normalized gene expression dataset (502) from the process 400 and the standard dataset (504), also normalized from the process 400. The two datasets 502/504 may be sampled an equal number of times, at processes 506 and 508, respectively. The sampling may be over random locations within the datasets, or based on a plurality of meta-data elements, for example, by cancer type, tissue type, age, gender, etc. Sampling can be done for all genes or may be confined to gene expression data within certain ranges of data, such as for example over certain genes or collections of genes identified in the datasets. Further, the total sample sizes for each dataset may vary, but generally should be at least 30 samples in size.

[0067] In the illustrated example, for each sampled dataset for which there is no paired data, for each gene, gene expression values were sorted (510 and 512) based on numerical values and used to estimate a statistical mapping/statistical transformation model (at process 514), in the form of a linear transformation model, for each gene. A linear transformation model is an example, as other techniques may be used to model the new (external) dataset to the standard (internal) dataset.

[0068] In exemplary embodiments, the linear transformation model (514) converts data from one type of data to another. The linear transformations are performed for each sample mapping from one dataset to the other, and the corresponding intercept and beta values for each linear transformation are stored (at process 516). The sampling is repeated, e.g., 10, 100, 1000, or 10000

times (e.g., through an iterative process), and the corresponding intercept and beta values are determined, and the mean intercept and mean beta values are computed for the linear transformations (518).

[0069] The mean beta and intercept values are then stored (at process 518) as conversion factors that may be used to correct the normalized external dataset from process 400. For example, a process 520 may subtract the mean intercept from the gene expression values in the normalized external dataset and divide the gene expression values by the mean beta for each gene. The mean intercept and mean beta comes from taking the average of X number of sampling iterations (through iteration feedback 521), for example 100 iterations, to estimate the model. At a process 522, any gene expression value after correction, that is below 0, is set to that minimum, e.g., 0, since gene expression values are constrained to be non-negative. The resulting normalized and corrected external dataset (524) is produced and stored by the system 100, either separately or stored as part of the dataset 116, for example.

[0070] FIG. 6 illustrates an example correction process 600 to be applied to the normalized RNA seq data produced by the process 400 for paired datasets. Two datasets 602 and 604 have been combined through a normalization process to form a paired dataset 606. An example of a paired dataset 606 would include, but is not limited to, data generated in the same manner as the standard dataset (602), and data, for the same set of samples, using a different data generation process (i.e. data generated using polyA-capture based RNA sequencing and exome-capture based RNA sequencing for the same set of samples). For the illustrated example, in order to calculate the per gene correction factors, a statistical mapping (610) would be created between the samples in the new RNA sequencing data (602) to the standard RNA sequencing data (602), using a model. The model parameters from the statistical mapping, e.g., the beta and intercept value, are obtained (612), stored as conversion factors, and used to correct the new RNA sequencing data, e.g., by subtracting the intercept value and dividing by the beta value. These conversion factors would be used to transform the new RNA sequencing dataset into the standard dataset and be deposited into the standard dataset database (614), in a similar manner to that of process 500, from minimum expression values, and a normalized and corrected dataset 616 is formed.

[0071] The normalized and corrected gene expression data may be provided as input data to any number of data analysis processes, data display processes, etc. The normalized and corrected gene expression data may be combined with additional types of data for such processes, as well. Examples of additional types of data that could be combined with the present application or be presented in addition to, include proteomics, metabolomics, metabonomics, epigenetics, microbiome, radiomics, and genomics data. Other examples may include non-molecular data such as clinical, epidemiological, demographic, etc. Proteomics data may comprise of protein expression,

protein modifications, and protein interactions obtained from high-throughput proteomic technologies such as mass spectrometry-based tech or microarrays. Metabolomic and metabonomic data may include small molecule metabolites, hormones, other signalling molecules, or metabolic responses obtained by mass spectrometry-based techniques, NMR spectrometry, etc. Epigenetic data may include changes in chromatin structure, such as histone modifications; transcript stability, such as DNA methylation status; nuclear organization; and small noncoding RNA species. These types of data may be obtained from high-performance liquid chromatography, bisulfite sequencing, CpG island microarrays, and chromatin immunoprecipitation-based methods. Microbiome and microbiota data may include and be obtained from direct observation methods, 16s rRNA sequencing, 18s sequencing, ITS gene sequencing, and molecular profiling such as metatranscriptomics, metaproteomics, metabolomics. Radiomics and digital imaging data may include and be obtained from PET, CT, histology slides and/or images, etc. Genomic data may include DNA sequencing data of coding and noncoding genomic regions of interest, and RNA sequencing data of coding and noncoding RNAs such as microRNA. Coding RNAs and gene expression data may also be obtained from single cell RNA sequencing and microarray. Noncoding RNAs may be obtained from RNA sequencing, polymerase chain reaction, and microarrays. Organoid culture assays may include healthy and disease state organoid cultures obtained from humans or animal model, such as a rodent.

[0072] FIG. 7 illustrates an example application of the gene expression normalization and correction framework 102 communicatively coupled to RNA seq analysis systems to make the standard gene expression dataset 116 available for further processing. In the illustrated example 700, the framework 102 can send the normalized and corrected dataset 116 to a RNA seq and imaging features machine learning framework 702. The framework 702 is a multi-modal framework capable of predicting immune infiltration based on integrating the normalized and corrected RNA seq data with digital imaging features. Framework 702 may be configured to predict immune infiltration in tumor samples, based on the combined data, by using a neural network framework that integrates the normalized and corrected RNA seq data neural network layer(s) with imaging feature neural network layer(s) to produce an integrated neural network output that can be used with a prediction function to produce an immune infiltration score for sample data.

[0073] In another example, the framework 102 may send the dataset 116 to another gene expression analyzer 704, providing automated processes for examining for example RNA seq data. Examples of the analyzer 704 include cancer type predictor systems, tissue/metastasis deconvolution systems, gene expression machine learning algorithms, patient report generators, and hormone receptor prediction systems. For example, the database 116, as a result of the

framework 102, can be applied to the framework 704 which may analyze the normalized and corrected RNA seq datasets for further processing.

[0074] In some examples, the database 116 is network accessible database communicatively coupled to (or part of) a network server for providing the dataset (or access thereto) to shared external sources, such as the additional data sources described herein.

[0075] In some examples, the database 116 may provide access to the dataset for user interaction through a user terminal (as shown), a patient report generator, clinician portable device, etc., e.g., through the network 106 or through a separate network 706.

[0076] While various examples herein are described in reference to gene expression data in the form of RNA seq data, it will be appreciated that the same techniques may be applied to transcript or isoform level expression data, in a similar manner.

[0077] An example workflow implementation of the present techniques includes receiving a biological sample, such as a tissue sample, and extracting RNA from the tissue sample, where the RNA is sequenced using a protocol, such as exome-capture RNA seq. RNA seq data may then be processes to go from raw sequence data to aligned reads and expression counts, for example, using the Kallisto pipeline technique (https://www.nature.com/articles/nbt.3519). Of course, any number of suitable pipelines can be used. These raw expression counts are then provided to the processes in FIGS. 3-6 to develop a continuously updated and updatable reference RNA seq dataset, which can then be provided to downstream gene expression analyzers like elements 702, 704, etc.

[0078] FIG. 8 provides an exemplary example workflow 800 that may be used to provide a corrected and normalized dataset. An RNA seq dataset 802 is accessed and quantified by a framework to generate a quantified output of RNA seq dataset 804.

[0079] In an example, a bioinformatics pipeline may be used to process the RNA seq data to get a raw counts RNAseq dataset for normalization and correction. The bioinformatics pipeline may receive a FASTQ file and produce a raw RNA counts file. In one exemplary bioinformatics pipeline, RNA seq dataset is accessed and a quantification using pseudoalignment is performed. The pseudoalignment may be implemented using a transcriptome de Bruijn graph, for example. The quantification process may split a given read into k-mers (k = 31 in our case) and then map each k-mer to a node in an internal database. The intersection of the k-mers is then used to quantify transcript-level expression. The output may be a near-optimal quantification of the expression of 180,053 transcripts, for example.

[0080] In an example, at a process 806, the framework performs a sampling and quality control process on a RNA seq dataset, after the bioinformatics pipeline produces an output or before the

normalization steps described herein are carried out. For example, the framework may determine sequencing depth in the quantified RNA seq dataset. The framework may determine the number of expressed transcripts and the number of expressed genes. The framework may filter obvious outliers, e.g., by removing identified duplicates. In some examples, the framework filters transcripts that are off-target from a probe set .

[0081] In a series of next steps, a preliminary normalization (such as from processes 300 and 400) is performed on the RNA seq dataset. In the illustrated example, the normalization is an intra-dataset normalization, where the dataset is normalized against other data in that dataset, at a process 808. In some examples, an inter-dataset normalization is also is performed, that is, as discussed below through a normalization comparing gene expression data from different datasets. To achieve intra-dataset normalization, at the process 808, a preliminary (and temporary) normalized dataset is stored (at process 810) and, at least for the illustrated example, principal component analysis (PCA) and outlier detection is performed on that dataset, at a process 812. FIG. 9 illustrates an example, in which RNA seq data has been applied to linear mapping model and outlier gene expression data have been identified. For example, gene expression data that does not map to the x-axis (0 value) may be identified as an outlier and removed. In some examples, a threshold, cutoff value is used to identifier outliers, such as a value of ±0.1, ±0.01, ±0.005, etc. As shown in FIG. 9, outliers can be found in the data, but such outliers are resolved through the process described above. FIG. 10 illustrates an example identification of resolved outliers resulting from applying the process 800 to a dataset. A cleaned and intra-normalized RNA seq dataset 814) results, as shown in FIG. 8.

[0082] Next, the framework implementing the process 800 (at process 816) performs a normalization and correction on the dataset 814, e.g., by determining geometric mean expressions against a reference dataset, where these expressions are correction factors for the RNA seq data. For example, the conversion factor (e.g., an intercept and a beta value for the linear mapping model), may be generated by comparison to an internal reference dataset, such as a first RNA seq dataset, i.e., an already normalized gene expression dataset. The resulting cleaned and inter-normalized dataset 820 is corrected (822) against the internal dataset 820 and a final corrected and normalized RNA seq dataset is generated (824). That final dataset may then be combined into the reference dataset and/or used for further downstream processing, such as discussed in reference to FIG. 7. FIGS. 11A illustrates an example of a gene expression values in a second dataset (Dataset2) prior to application to the correction workflow 800. FIG. 11B illustrates the gene expression values of the second dataset (Dataset2) after correction and normalization, illustrating the updated dataset against a reference dataset (Dataset1). The x and y axes reflect a first and second principal component from the PCA analysis.

[0083] FIG. 12 illustrates another example system 900 for normalizing gene expression data, such as RNA seq data, and having a similar configuration to that of the system 100. A multimodal normalization framework 902 is coupled to receive gene expression data from different sources through the communication network 106, such as the health care provider computing system 104 that makes available stored gene expression data in the form of the RNA sequencing dataset 108. Other network-accessible gene expression datasets include the TCGA dataset 110 and the GTEx dataset 112. As with the normalization framework 102, the multimodal normalization framework 902 may be implemented on a computing device such as a computer, tablet or other mobile computing device, or server. The framework 902 may be implemented by any number of processors, controllers or other electronic components for processing or facilitating the RNA sequencing data analyses. In some examples, the system 900 is implemented in a broader system that includes processing and hardware for imaging feature analysis, such as analyzing features in medical imaging data, immune infiltration data analysis, DNA sequencing data analysis, organoid development analysis, and/or other modality analyses.

[0084] The multimodal normalization framework 902 includes a modal identifier 904 and a gene expression data normalizer 906. Gene expression datasets are provided to, or accessed by, the framework 902 for normalization processing. The modal identifier 904 is configured to receive the gene expression datasets and analyze gene expression data therein to determine if any of gene expression data exhibits more than one modal expression peak. Such analysis may be performed on each gene expression data within the received dataset. Multimodal gene expression data is gene expression data that exhibits multiple modals of expression within the same population, i.e., multiple expression distribution peaks. For example, FIG. 13 illustrates gene expression data for ESR1 exhibiting a bimodal distribution with two peaks, labeled at L and R. These expression peaks may result from two different factors, such as tumor type and tissue type, which each affect ESR1 expression in this example. More generally, multimodal gene expression data can exhibit expression peaks due to a number of different factors, including, but not limited to, tissue type, cancer type, purity of tumor within sample (for example with different peaks due to different purity levels, 10%, 20%, 30%, 40%, at least 50%, at least 60%, at least 70%, at least 80%, and at least 90%), cell type (immune, lymphocyte, red blood cell, cytotoxic T cells, B cells, NK cells, macrophages, etc.), and sex of subject. Cancer types may include, but are not limited to, epithelial ovarian carcinoma, colon cancer, esophageal cancer, melanoma, endometrial cancer, and breast cancer. Other factors include batch effects, such as, differences in bio-informatics pipelines used to generate the gene expression datasets, differences in sequencing machines, dates of collection of gene expression data, and contamination of tissue.

[0085] The modal identifier 904 is configured to apply a regression technique to identify the one or more modal expression peaks in the gene expression data. In an example, the modal identifier 904 is configured as a Decision Tree Regressor. For a bimodal distribution, for example, the modal identifier 904 may implement a 2-Leaf Decision Tree Regressor that performs an auto-encoding on the gene expression data to identify two distribution peaks that minimize the mean square error (MSE) within the distribution data. The resulting two distribution peaks then are the lower and upper peak points in the gene expression data.

[0086] The gene expression data normalizer 906 receives the modal distribution peak data and gene expression data from the identifier 904 and performs a normalization on the gene expression data. FIG. 14 illustrates an example normalization process 1000 performed by the multimodal framework 902 on the gene expression data of FIG. 13. The initial gene expression data 1002, such as an RNA seq dataset, is received at the modal identifier 904, which identifies (at process 1004) one or more modal expression peaks in gene expression data within the dataset 1002. The gene expression data normalizer 906 normalizes the one or more modal expression peaks by applying a normalization rule that, in the illustrated example, normalizes a spacing distance between modal expression peaks. In the example of a bimodal distribution like that of FIG. 13, the normalizer sets a spacing distance of 1 between the identified peaks, resulting in the normalized distribution of FIG. 15. In the example of more than two distribution peaks, the normalizer 906 may set an equal spacing distance between each of the distribution peaks. In yet another example, when there are more than two distribution peaks, the normalizer 906 may establish a normalized spacing distance (e.g., a distance of 1), between the outermost peaks. Take for example, using a 2-Leaf Decision Tree Regressor approach, the normalizer 906 may be configured to optimize for the best point between auxiliary peaks (i.e., any of the peaks) to minimize overall mean-squared error in the distribution. In another example, a Decision Tree Regressor having enough leaves to match the number of peaks may be used, in which example, the normalizer 906 may be configured to perform a unit-norm between the outermost peaks, or configured to performed a unit-norm between inner-most peaks. In yet other examples of these multiple leaf Decision Tree Regressors, the normalizer 906 may be configured to proportionally space distance between detected peaks based on their individual proximity (e.g., with one far left peak and two right side peaks, the normalizer could be configured to place the left peak at -0.5, and the inner-most right peak at +0.5, and the outer most right peak at +1.0, etc). In an example, the normalizer determines the spacing distance by dividing the peak expression values (R and L) by a delta value between the R and L such that the distance between them is a normalization value, such as 1.0, resulting in normalized peaks R' and L' as shown in FIG. 15.

[0087] Optionally, in some examples, the process 1006 further performs a shift on the normalized spacing gene expression data to align the peaks around a reference baseline expression value, such as a zero (0) expression. An example shift applied to the normalized bimodal gene expression data of FIG. 15 is shown in FIG. 16, resulting in shifted peaks Ls and Rs centered around a zero reference value. As a result of the shifting, over expression and under expression can be identified more readily in the gene expression data.

[0088] The normalized gene expression data is then stored in a normalized gene expression dataset at process 1008. In some examples, the process 1008 may remove the un-normalized gene expression data from the dataset 1002 and replace that data with the normalized gene expression data. In some examples, the normalized gene expression data may be added to the dataset 1002. In yet other examples, the normalized gene expression data is added to a separate normalized gene expression dataset 908 (shown in FIG. 12).

[0089] This normalization may be applied across all gene expression data within the dataset 1002 to generate a normalized gene expression dataset that aligns each of the different gene expression data within the dataset. At a process 1010, the framework 902 determines if there is additional gene expression data within the dataset to be normalized, and if so the process 900 repeats applying the distribution peak spacing normalization rule (and optional shifting rule) to each subsequent gene expression data, until a completed normalized gene expression dataset 1012 (e.g., dataset 908) is formed.

[0090] FIG. 17A shows example gene expression data corresponding to four different genes (AR, PGR, ESR1, and ERBB2) prior to normalization. Each of the gene expression data exhibits bimodal distribution peaks, for example, resulting from different expression of the gene in different tissue. FIG. 17B illustrates the gene expression data after normalization applied by the process 1000 of the framework 902. As shown, the normalized gene expression data has the bimodal distribution peaks aligned, L and R, and centered around a zero reference expression.

[0091] The normalization of process 1000 may be applied to gene expression data exhibiting uni-modal expression distribution, such as shown in FIG. 18. With the modal identifier 904 configured as a bimodal peak identifier (e.g., a 2-Leaf Distribution Tree Regressor), the framework 902 identifies imposed "peaks" on the distribution as the locations on the distribution that minimize the mean-square error for the distribution. With these imposed "peaks" identified, the normalizer 906 may perform a transformation on the data to establish a normalized spacing between these peaks and another, linear transformation to shift the distribution. FIG. 19A illustrates uni-modal gene expression data for genes BRCA1, BRCA2, and PIK3CA prior to normalization by the process 1000, and FIG. 19B illustrates the corresponding normalized gene expression data after the process 1000.

[0092] By identifying and normalization multi-modal gene expression data within a dataset, such as within the RNA transcriptome, an gene sequence analyzer, such as the RNA Seq analyzer 704 in FIG. 7, is able to generate more accurate gene expression data for more accurate identification of population groups, suggested treatments, biomarker discovery, molecular sub-type clustering and identification, population clustering visualization, etc. For example, with a modal identifier configured as a bimodal identifier, a normalized gene expression dataset is formed where one of the expression factors may be isolated out from affecting analysis. FIG. 20A illustrates a Uniform Manifold Approximation and Projection (UMAP) plot of ESR1 gene expression data prior to normalization. The UMAP plot shows a large distance between two different clusters, A and B: one (A) that corresponds to expression data captured from a first tissue type, in this case liver tissue, and another (B) that corresponds to a second tissue type, breast tissue. The distances between the clusters demonstrates that any attempt to use the UMAP to identify ESR1 expression is tissue dependent. FIG. 20B illustrates another UMAP plot but after normalization, where tissue dependence, as shown, has been removed from the data. The UMAP visualization of the gene expression data is achieved computationally faster or more accurately with removed tissue using the normalization process. The same computational speed efficiency and tissue removal accuracy can be achieved in other visualization techniques, including principal component analysis (PCA) and t-Distributed Stochastic Neighbor Embedding (t-SNE). Indeed, the computationally efficiencies are considerable for visualizations such as UMAP, which is generally faster than t-SNE and generally is more accurate than PCA. With the normalization techniques herein, a RNA Seq analyzer can generate more accurate gene expression data reports, as a result. More generally, with the normalized data, the RNA Seq analyzer can more accurately identify samples based on expression values for ESR1, with the tissue dependence removed. Moreover, the RNA Seq analyzer can remove tissue dependence (or any other factor being considered against cancer in a bimodal analysis configuration) across all gene expression data. Thus, with the present techniques, gene expression data for different genes (and thus for different cancer types) can be normalized to be tissue independent, thereby allowing an RNA Seq analyzer to more quickly and more accurately identify cancer type for a subject irrespective of whether the tissue sample is from a primary site of cancer or a secondary malignant cancer site.

[0093] In some aspects, the GC bias length may be normalized in order to more effectively permit comparison of gene expression in a single sample. In some aspects, the read depth and gene length may be normalized to more effectively permit comparison of gene expression across multiple samples. The normalization may be performed on a set of paired-end RNA reads or a set of single-end RNA reads. The normalization may be performed on RNA-seq data or other RNA data that is generated using methods known in the art.

[0094] In one aspect, a normalized set of RNA may be utilized in connection with expression calling. Prior to normalization, samples may be biased by the depth of sequencing.

[0095] Comparison of transcriptome measures from among samples may be biased by depth of sequencing. Normalization permits comparison of expression levels of a single gene across samples. For instance, when calling overexpression of a gene, the overexpression may be made with respect to expression of other samples. As an example, sequencing of 20 breast cancer specimens at a depth of 20 million reads may result in 100 reads of the ESR1 estrogen receptor gene for each sequenced specimen. Sequencing of another 20 breast cancer specimens at patients at a depth of 40 million reads may result in 200 reads of the ESR1 estrogen receptor gene for each sequenced specimen. Normalizing the two data sets permits normalization of the read count across the two data sets.

[0096] As another example, a normalized RNA data set may be utilized in connection with a tumor of unknown origin predictor model. The model may have to learn certain parameters for each gene. To apply those parameters to each gene among many different specimens, it is preferred that the gene expression value look the same across patients. If the model, for example, applies an estrogen level read depth by a factor of two, the model will be biased by the read depth. Where the tumor of unknown origin predictor model is formed as, for example, a linear model, each gene is provided with a weight by which the associated expression level is multiplied.

[0097] As another example, a normalized RNA data set may be utilized in connection with one or more methods to cluster samples in order, for instance, to identify disease subtypes. By comparing RNA expression levels among samples, clustering may be utilized to suggest those samples that are most similar to one another. In some embodiments, the normalization may be limited to normalizing read depth among samples. In other embodiments, the normalization may be limited to normalizing read depth and GC content. In other embodiments, the normalization may comprise normalization of read depth, GC content, and gene length. In an example, a set of normalized RNA transcriptomes may be matched with IHC staining information to identify cohorts of specimens with HER2+ status. For example, in a cohort of 400 specimens, 300 of the specimens may have an associated IHC stain and 100 do not. For the 100 that do not, an IHC prediction model may be used to predict the IHC status and then UMAP clustering may be utilized to cluster the specimens. Specimens with a normalized expression of ESR1 (for ER) or PGR (for PR) or ERBB2 (for HER2) above a pre-defined threshold may be stratified. In one embodiment the threshold is 2.5. Some specimens may have data available for ER, PR, and HER2 in which case the specimen is displayed in FIG. 21 as a circle. Other specimens may not have data available for ER, PR, or HER2 in which case the specimen may be displayed in FIG. 21 as an X mark.

[0098] As another example, a RNA normalization may be utilized to compare gene expression levels relative to each other within a sample. In some aspects, GC bias may be present in gene length. For example, if gene A is 100kb and gene B is 200kb, the same number of RNA molecules may exist for gene. However, gene B would have twice the counts of gene A because gene B's RNA molecule is twice the size. During PCR amplification in library prep, if a fragment has about 50% GC content it will have a first level of amplification. If, on the other hand, the GC content deviates significantly from 50% GC content, it will not amplify as well. For example, the GC content may deviate significantly if it has 80% content. A first gene with a first percentage GC content closer to 50% GC content and a second gene with a second percentage GC content that significantly deviates from the first gene content can have the same number of RNA molecules in the cell but the first GC content gene will have been amplified more than the second GC content gene during PCR amplification. RNA normalization of GC content may be utilized within a sample to compare the GC content of a first gene to the GC content of a second gene.

[0099] In another aspect, RNA normalization may be utilized in connection with a drug response model. In an exemplary drug response model, the model may multiply each gene expression value by a number the model has learned. The model may be trained on read depth normalized data and may be utilized to predict drug response using RNA expression information that has been normalized in a like fashion to the training RNA expression information. For instance, the drug response model may take the form $y = a1x1 + a2x2 + \ldots + anxn$, where a1, a2, …, an are weights and x1, x2, …, xn are genes. If $y < 1$ then the model may be set to not respond to the particular drug that is the focus of the model. If $y > 1$ the model may be set to respond to the particular drug that is the focus of the model.

[0100] In another aspect, RNA normalization may be utilized in connection with an assessment of pathway activity. For example, RNA expression data may be normalized as to GC content and length. For example, in the field of single sample gene set enrichment analysis, each gene's transcription levels may be normalized to adjust for GC bias in order to develop a ranked list of normalized gene expression values. The expression values of a pre-defined gene list, reflecting genes known to be associated with a pathway, may be examined in order to identify whether the genes in associated with the pathway are overexpressed, underexpressed, or a combination thereof that is relevant to the pathway. In this way, a set of normalized RNA data may be utilized to identify an activated pathway in the specimen.

[0101] In another aspect, RNA normalization may be utilized in connection with a comparison of expression levels of a given gene among a set of patients. For instance the read depth may be normalized in order to compare the expression levels of a BRAF mutation among patients.

[0102] In another aspect, RNA normalization may be utilized in connection with analysis of RNA expression information in order to identify potential sample swaps or input missing data. For example, a model y = a1x1 + a2x2 + … + anxn, where a1, a2, …, an are weights and x1, x2, …, xn are genes may be trained on a set of RNA expression information and the patient's gender. Read count and GC count may be normalized across the applicable RNA data set. By inputting the normalized RNA expression information of a new specimen, normalized in a like fashion to the training data set, it is possible to determine whether the specimen is from a male patient or a female patient. If the gender of the patient from whom the specimen was received was reported as male, but the gender analysis indicates the specimen came from a female person, the disparity would indicate a quality control process to confirm whether the specimen was the result of a sample swap, was taken from a patient who had a gender reassignment, or was from a patient whose gender was mis-identified in the patient's electronic health record.

[0103] Throughout this specification, plural instances may implement components, operations, or structures described as a single instance. Although individual operations of one or more methods are illustrated and described as separate operations, one or more of the individual operations may be performed concurrently, and nothing requires that the operations be performed in the order illustrated. Structures and functionality presented as separate components in example configurations may be implemented as a combined structure or component. Similarly, structures and functionality presented as a single component may be implemented as separate components or multiple components. These and other variations, modifications, additions, and improvements fall within the scope of the subject matter herein.

[0104] Additionally, certain embodiments are described herein as including logic or a number of routines, subroutines, applications, or instructions. These may constitute either software (e.g., code embodied on a machine-readable medium or in a transmission signal) or hardware. In hardware, the routines, etc., are tangible units capable of performing certain operations and may be configured or arranged in a certain manner. In example embodiments, one or more computer systems (e.g., a standalone, client or server computer system) or one or more hardware modules of a computer system (e.g., a processor or a group of processors) may be configured by software (e.g., an application or application portion) as a hardware module that operates to perform certain operations as described herein.

[0105] In various embodiments, a hardware module may be implemented mechanically or electronically. For example, a hardware module may comprise dedicated circuitry or logic that is permanently configured (e.g., as a special-purpose processor, such as a microcontroller, field programmable gate array (FPGA) or an application-specific integrated circuit (ASIC)) to perform certain operations. A hardware module may also comprise programmable logic or circuitry (e.g., as

encompassed within a processor or other programmable processor) that is temporarily configured by software to perform certain operations. It will be appreciated that the decision to implement a hardware module mechanically, in dedicated and permanently configured circuitry, or in temporarily configured circuitry (e.g., configured by software) may be driven by cost and time considerations.

[0106] Accordingly, the term "hardware module" should be understood to encompass a tangible entity, be that an entity that is physically constructed, permanently configured (e.g., hardwired), or temporarily configured (e.g., programmed) to operate in a certain manner or to perform certain operations described herein. Considering embodiments in which hardware modules are temporarily configured (e.g., programmed), each of the hardware modules need not be configured or instantiated at any one instance in time. For example, where the hardware modules comprise a processor configured using software, the processor may be configured as respective different hardware modules at different times. Software may accordingly configure a processor, for example, to constitute a particular hardware module at one instance of time and to constitute a different hardware module at a different instance of time.

[0107] Hardware modules can provide information to, and receive information from, other hardware modules. Accordingly, the described hardware modules may be regarded as being communicatively coupled. Where multiple of such hardware modules exist contemporaneously, communications may be achieved through signal transmission (e.g., over appropriate circuits and buses) that connects the hardware modules. In embodiments in which multiple hardware modules are configured or instantiated at different times, communications between such hardware modules may be achieved, for example, through the storage and retrieval of information in memory structures to which the multiple hardware modules have access. For example, one hardware module may perform an operation and store the output of that operation in a memory device to which it is communicatively coupled. A further hardware module may then, at a later time, access the memory device to retrieve and process the stored output. Hardware modules may also initiate communications with input or output devices, and can operate on a resource (e.g., a collection of information).

[0108] The various operations of the example methods described herein can be performed, at least partially, by one or more processors that are temporarily configured (e.g., by software) or permanently configured to perform the relevant operations. Whether temporarily or permanently configured, such processors may constitute processor-implemented modules that operate to perform one or more operations or functions. The modules referred to herein may, in some example embodiments, comprise processor-implemented modules.

[0109] Similarly, the methods or routines described herein may be at least partially processor-implemented. For example, at least some of the operations of a method can be performed by one or more processors or processor-implemented hardware modules. The performance of certain of the operations may be distributed among the one or more processors, not only residing within a single machine, but also deployed across a number of machines. In some example embodiments, the processor or processors may be located in a single location (e.g., within a home environment, an office environment or as a server farm), while in other embodiments the processors may be distributed across a number of locations.

[0110] The performance of certain of the operations may be distributed among the one or more processors, not only residing within a single machine, but also deployed across a number of machines. In some example embodiments, the one or more processors or processor-implemented modules may be located in a single geographic location (e.g., within a home environment, an office environment, or a server farm). In other example embodiments, the one or more processors or processor-implemented modules may be distributed across a number of geographic locations.

[0111] Unless specifically stated otherwise, discussions herein using words such as "processing," "computing," "calculating," "determining," "presenting," "displaying," or the like may refer to actions or processes of a machine (e.g., a computer) that manipulates or transforms data represented as physical (e.g., electronic, magnetic, or optical) quantities within one or more memories (e.g., volatile memory, non-volatile memory, or a combination thereof), registers, or other machine components that receive, store, transmit, or display information.

[0112] As used herein any reference to "one embodiment" or "an embodiment" means that a particular element, feature, structure, or characteristic described in connection with the embodiment is included in at least one embodiment. The appearances of the phrase "in one embodiment" in various places in the specification are not necessarily all referring to the same embodiment.

[0113] Some embodiments may be described using the expression "coupled" and "connected" along with their derivatives. For example, some embodiments may be described using the term "coupled" to indicate that two or more elements are in direct physical or electrical contact. The term "coupled," however, may also mean that two or more elements are not in direct contact with each other, but yet still co-operate or interact with each other. The embodiments are not limited in this context.

[0114] As used herein, the terms "comprises," "comprising," "includes," "including," "has," "having" or any other variation thereof, are intended to cover a non-exclusive inclusion. For example, a process, method, article, or apparatus that comprises a list of elements is not necessarily

limited to only those elements but may include other elements not expressly listed or inherent to such process, method, article, or apparatus. Further, unless expressly stated to the contrary, "or" refers to an inclusive or and not to an exclusive or. For example, a condition A or B is satisfied by any one of the following: A is true (or present) and B is false (or not present), A is false (or not present) and B is true (or present), and both A and B are true (or present).

[0115] In addition, use of the "a" or "an" are employed to describe elements and components of the embodiments herein. This is done merely for convenience and to give a general sense of the description. This description, and the claims that follow, should be read to include one or at least one and the singular also includes the plural unless it is obvious that it is meant otherwise.

[0116] This detailed description is to be construed as an example only and does not describe every possible embodiment, as describing every possible embodiment would be impractical, if not impossible. One could implement numerous alternative embodiments, using either current technology or technology developed after the filing date of this application.

What is Claimed:

1. A computer-implemented method comprising:

generating, from a comparison of a normalized RNA sequence dataset against a standard RNA sequence dataset, at least one conversion factor for applying to a next RNA sequence dataset; and

correcting RNA sequence data of the next RNA sequence dataset using the at least one conversion factor.

2. The computer-implemented method of claim 1, further comprising:

including the corrected RNA sequence data of the next RNA sequence dataset into the standard RNA sequence dataset.

3. The computer-implemented method of claim 1, further comprising:

obtaining a gene expression dataset comprising the RNA sequence data for one or more genes, the RNA sequence data including gene length data, guanine-cytosine (GC) content data, and depth of sequencing data; and

normalizing the RNA sequence data against the standard RNA sequence data by comparing the RNA sequence data for the one or more genes to sequence data in the standard RNA sequence dataset.

4. The computer-implemented method of claim 3, wherein normalizing the RNA sequence data comprises:

normalizing the gene length data for the one or more genes to reduce systematic bias;

normalizing the GC content data for the one or more genes to reduce systematic bias; and

normalizing the depth of sequencing data for the RNA sequence data.

5. The computer-implemented method of claim 4, wherein normalizing the gene length data comprises using a quantile normalization procedure.

6. The computer-implemented method of claim 4, wherein normalizing the GC content data comprises using a quantile normalization procedure.

7. The computer-implemented method of claim 4, wherein normalizing the depth of sequencing data comprises:

determining a ratio of expression data to reference geometric mean expression data obtained from the standard RNA sequence dataset;

determining ratios of expression data to reference geometric mean expression data for a plurality of additional RNA sequence data corresponding to the at least one gene, to develop a set of ratios for the gene expression dataset; and

determining a size factor as a median of the set of ratios.

8. The computer-implemented method of claim 3, wherein normalizing the RNA sequence data comprises applying a Reads Per Kilobase Million (RPKM) normalization, a Fragments Per Kilobase Million (FPKM) normalization, or a Transcripts Per Kilobase Million (TPM) normalization.

9. The computer-implemented method of claim 1, wherein the RNA sequence dataset is a raw dataset.

10. The computer-implemented method of claim 1, wherein the RNA sequence dataset is a Cancer Genome Atlas (TCGA) dataset.

11. The computer-implemented method of claim 1, wherein the RNA sequence dataset is a Genotype-Tissue Expression (GTEx) dataset.

12. The computer-implemented method of claim 1, wherein generating the at least one conversion factor comprises:

for a sample gene, obtaining sample data from a normalized RNA sequence dataset and obtaining sample data from the standard RNA sequence dataset;

determining a statistical mapping between the sample data of the normalized RNA sequence dataset and the sample data of the standard RNA sequence dataset; and

determining the at least one conversion factor using the statistical mapping.

13. The computer-implemented method of claim 12, wherein determining the statistical mapping comprises determining a linear mapping model between the sample data of the normalized RNA sequence dataset and the sample data of the standard RNA sequence dataset, the method further comprising:

determining an intercept and a beta value for the linear mapping model; and

determining the at least one conversion factor using the statistical mapping from the intercept and the beta value.

14. The computer-implemented method of claim 12, wherein generating the at least one conversion factor comprises:

(i) for a sample gene, obtaining sample data from normalized RNA sequence dataset and obtaining sample data from the standard RNA sequence dataset;

(ii) determining a linear mapping model between the sample data of the normalized RNA sequence dataset and the sample data of the standard RNA sequence dataset;

(iii) determining an intercept and a beta value for the linear mapping model;

(iv) performing (i) – (iv) a plurality of times for the sample gene; and

(iv) determining a gene specific conversion factor from a mean intercept and a mean beta value for the plurality of times.

15. A computing device comprising one or more memories and one or more processors configured to:

generate, from a normalization of an RNA sequence data against a standard RNA sequence dataset, at least one conversion factor for applying to a next RNA sequence dataset; and

correct RNA sequence data of the next RNA sequence dataset using the at least one conversion factor.

16. The computing device of claim 15, wherein the one or more processors are configured to:

include the corrected RNA sequence data of the next RNA sequence dataset into the standard RNA sequence dataset.

17. The computing device of claim 15, wherein the one or more processors are configured to:

obtain a gene expression dataset comprising the RNA sequence data for one or more genes, the RNA sequence data including gene length data, guanine-cytosine (GC) content data, and depth of sequencing data; and

correct the RNA sequence data against the standard RNA sequence data by comparing the RNA sequence data for the one or more genes to sequence data in the standard RNA sequence dataset.

18. The computing device of claim 17, wherein the one or more processors are configured to normalize the RNA sequence data by being configured to:

normalize the gene length data for the one or more genes to reduce systematic bias;

normalize the GC content data for the one or more genes to reduce systematic bias; and

normalize the depth of sequencing data for the RNA sequence data.

19. The computing device of claim 15, wherein the one or more processors are configured to generate the at least one conversion factor by being configured to:

for a sample gene, obtain sample data from a normalized RNA sequence dataset and obtaining sample data from the standard RNA sequence dataset;

determine a statistical mapping between the sample data of the normalized RNA sequence dataset and the sample data of the standard RNA sequence dataset; and

determine the at least one conversion factor using the statistical mapping.

20. The computing device of claim 19, wherein the one or more processors are configured to determine the statistical mapping by being configured to determine a linear mapping model between the sample data of the normalized RNA sequence dataset and the sample data of the standard RNA sequence dataset, the one or more processors being further configured to:

determine an intercept and a beta value for the linear mapping model; and

determine the at least one conversion factor using the statistical mapping from the intercept and the beta value.

21. A computer-implemented method comprising:

receiving, at one or more processors, a gene expression dataset;

identifying within the gene expression dataset, using a regression technique implemented by the one or more processors, gene expression data having multiple modal expression peaks;

for the gene expression data, normalizing, using the one or more processors, a spacing between each of the multiple model expression peaks to form a normalized gene expression data; and

storing the normalized gene expression data in a normalized gene expression dataset.

22. The computer-implemented method of claims 21, wherein the gene expression dataset is an RNA sequence dataset.

23. The computer-implemented method of claims 21, the method further comprising normalizing, using the one or more processors, a reference baseline expression based on the multiple model expression peaks, wherein the reference baseline expression identifies over-expressed gene expression data and under-expressed gene expression data.

24. The computer-implemented method of claims 21, wherein the gene expression data has a bimodal distribution, and the multiple model expression peaks consist of two expression peaks.

25. The computer-implemented method of claims 24, wherein normalizing the spacing between the two expression peaks comprises setting the spacing to 1.

26. The computer-implemented method of claims 25, the method further comprising normalizing, using the one or more processors, a reference baseline expression for the two expression peaks by setting a zero expression value between the two expression peaks.

27. The computer-implemented method of claims 24, wherein the two expression peaks of the bimodal distribution of the gene expression data correspond to a tumor specific expression peak and a tissue specific expression peak.

28. The computer-implemented method of claims 24, wherein the regression technique is a two-leaf decision tree regressor.

29. The computer-implemented method of claims 21, wherein the regression technique is a multiple-leaf decision tree regressor.

30. The computer-implemented method of claims 21, wherein the gene expression dataset is an RNA sequence dataset comprising a plurality of gene expression data each corresponding to different gene and each having a bimodal distribution.

31. A computer-implemented method comprising:

receiving, at one or more processors, a RNA sequence dataset;

identifying within the gene expression dataset, using a regression technique implemented by the one or more processors, a plurality of RNA expression data each having a bimodal distribution comprising two expression peaks;

for each of the plurality of RNA expression data, normalizing, using the one or more processors, a spacing between the two expression peaks such that each of the plurality of RNA expression data has the same spacing between the two expression peaks; and

storing the normalized RNA expression data in a normalized RNA sequence dataset.

32. The computer-implemented method of claims 31, the method further comprising shifting each of the plurality of RNA expression data to have a zero expression value between the two expression peaks.

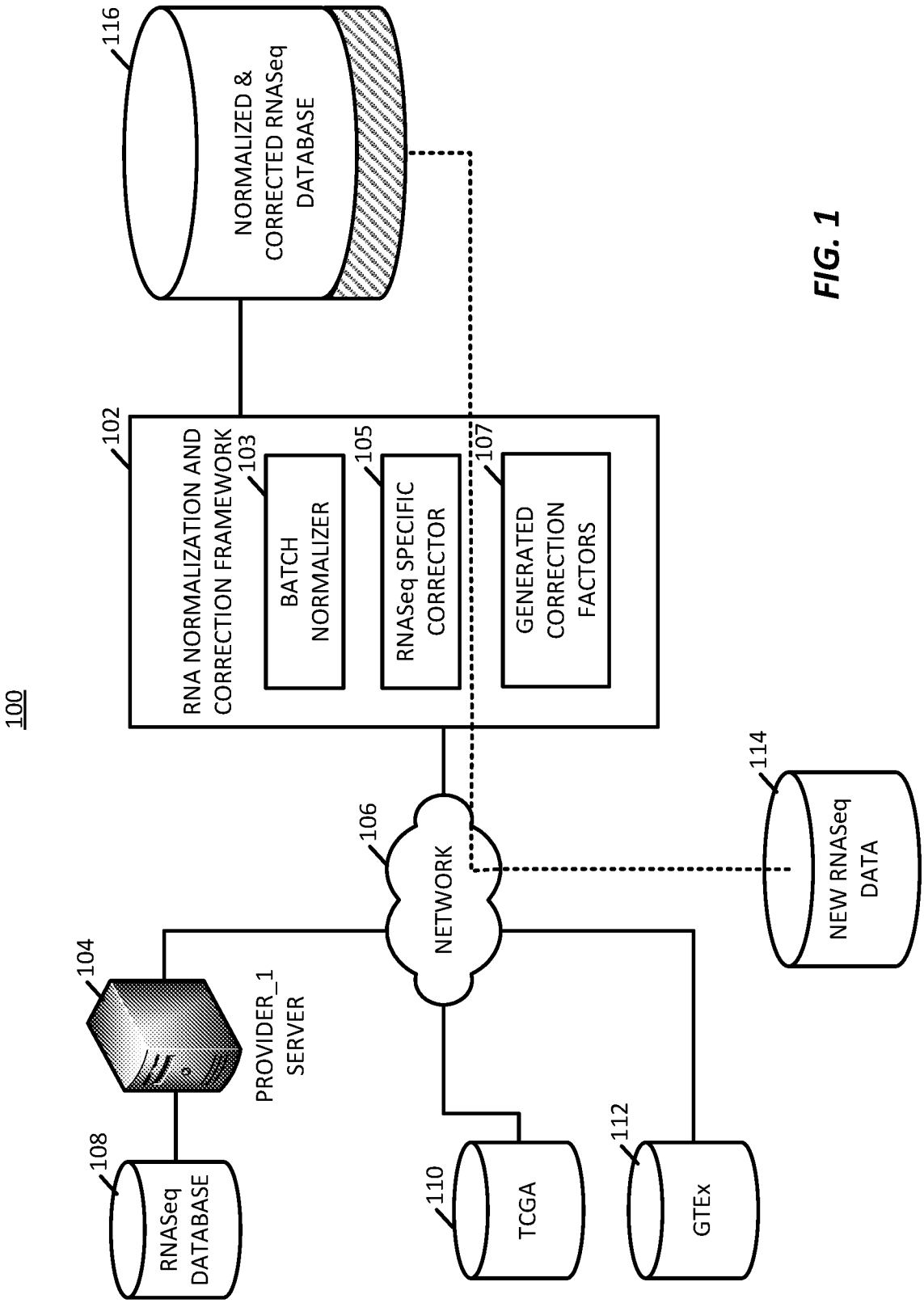

100
116
NORMALIZED & CORRECTED RNASeq DATABASE
102
RNA NORMALIZATION AND CORRECTION FRAMEWORK
103
BATCH NORMALIZER
105
RNASeq SPECIFIC CORRECTOR
107
GENERATED CORRECTION FACTORS
106
NETWORK
114
NEW RNASeq DATA
104
PROVIDER_1 SERVER
108
RNASeq DATABASE
110
TCGA
112
GTEx

FIG. 1

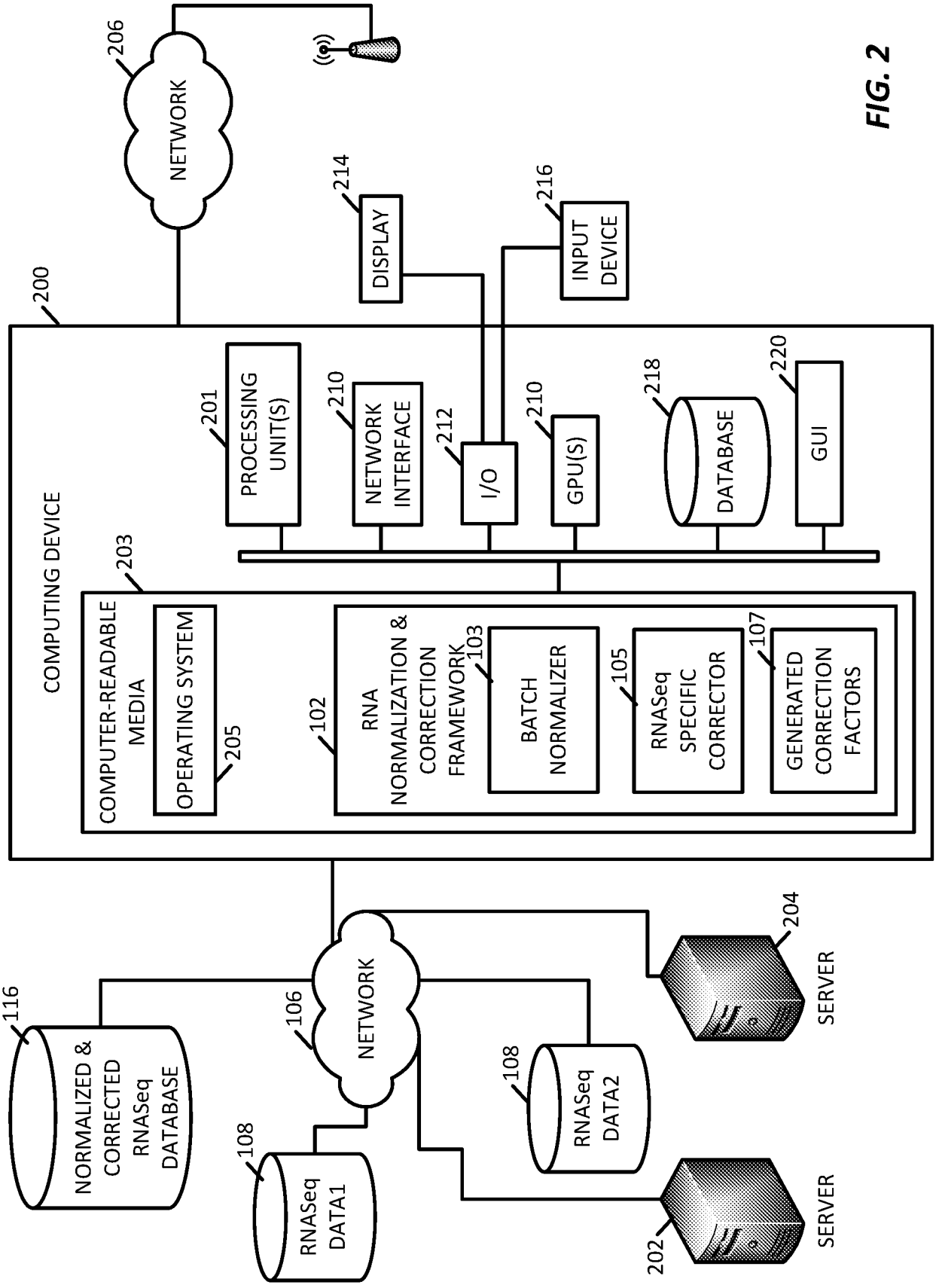

206
NETWORK
214
DISPLAY
216
INPUT DEVICE
200
COMPUTING DEVICE
201
PROCESSING UNIT(S)
210
NETWORK INTERFACE
212
I/O
210
GPU(S)
218
DATABASE
220
GUI
203
COMPUTER-READABLE MEDIA
OPERATING SYSTEM
205
102
RNA NORMALIZATION & CORRECTION FRAMEWORK
103
BATCH NORMALIZER
105
RNASeq SPECIFIC CORRECTOR
107
GENERATED CORRECTION FACTORS
116
NORMALIZED & CORRECTED RNASeq DATABASE
106
NETWORK
204
SERVER
108
RNASeq DATA2
108
RNASeq DATA1
202
SERVER

FIG. 2

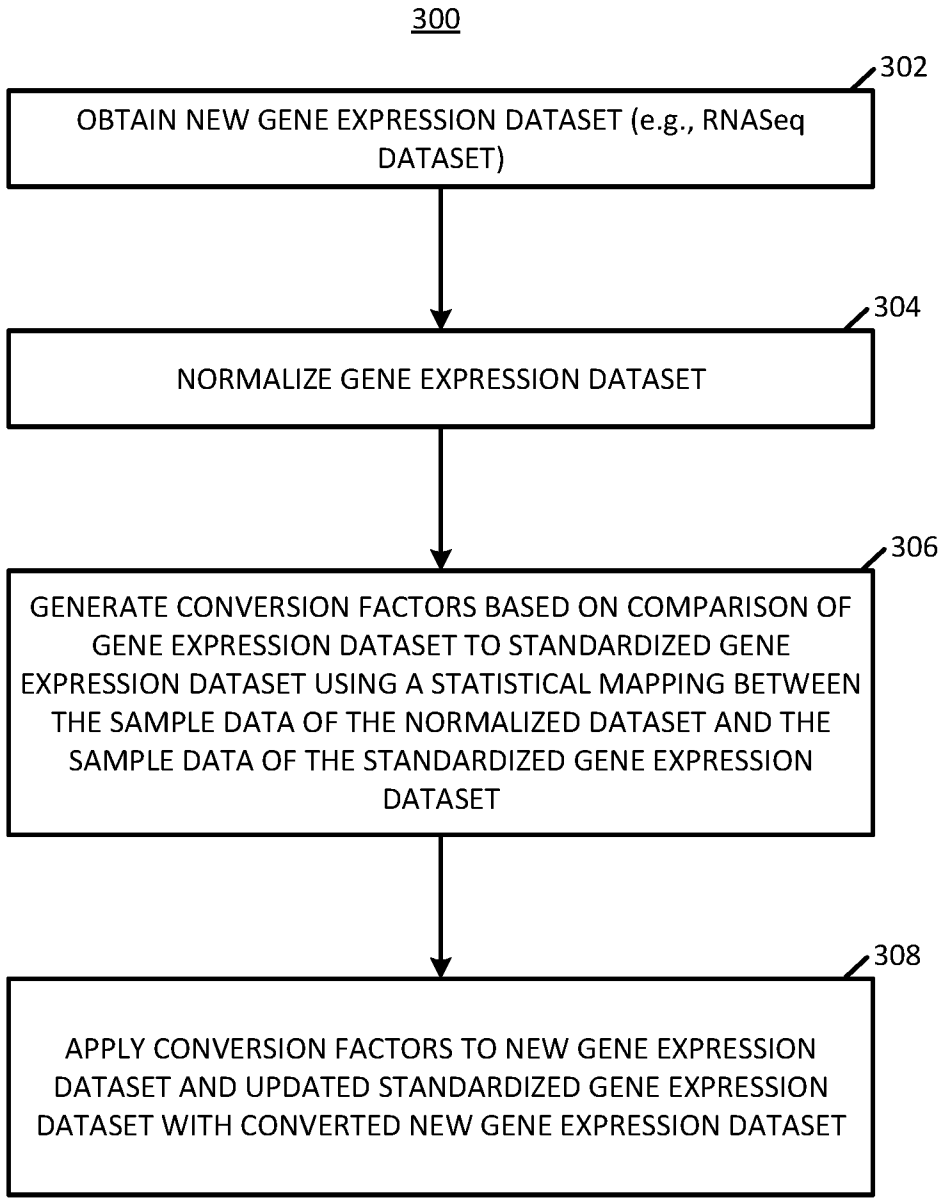

300
302
OBTAIN NEW GENE EXPRESSION DATASET (e.g., RNASeq DATASET)
304
NORMALIZE GENE EXPRESSION DATASET
306
GENERATE CONVERSION FACTORS BASED ON COMPARISON OF GENE EXPRESSION DATASET TO STANDARDIZED GENE EXPRESSION DATASET USING A STATISTICAL MAPPING BETWEEN THE SAMPLE DATA OF THE NORMALIZED DATASET AND THE SAMPLE DATA OF THE STANDARDIZED GENE EXPRESSION DATASET
308
APPLY CONVERSION FACTORS TO NEW GENE EXPRESSION DATASET AND UPDATED STANDARDIZED GENE EXPRESSION DATASET WITH CONVERTED NEW GENE EXPRESSION DATASET

FIG. 3

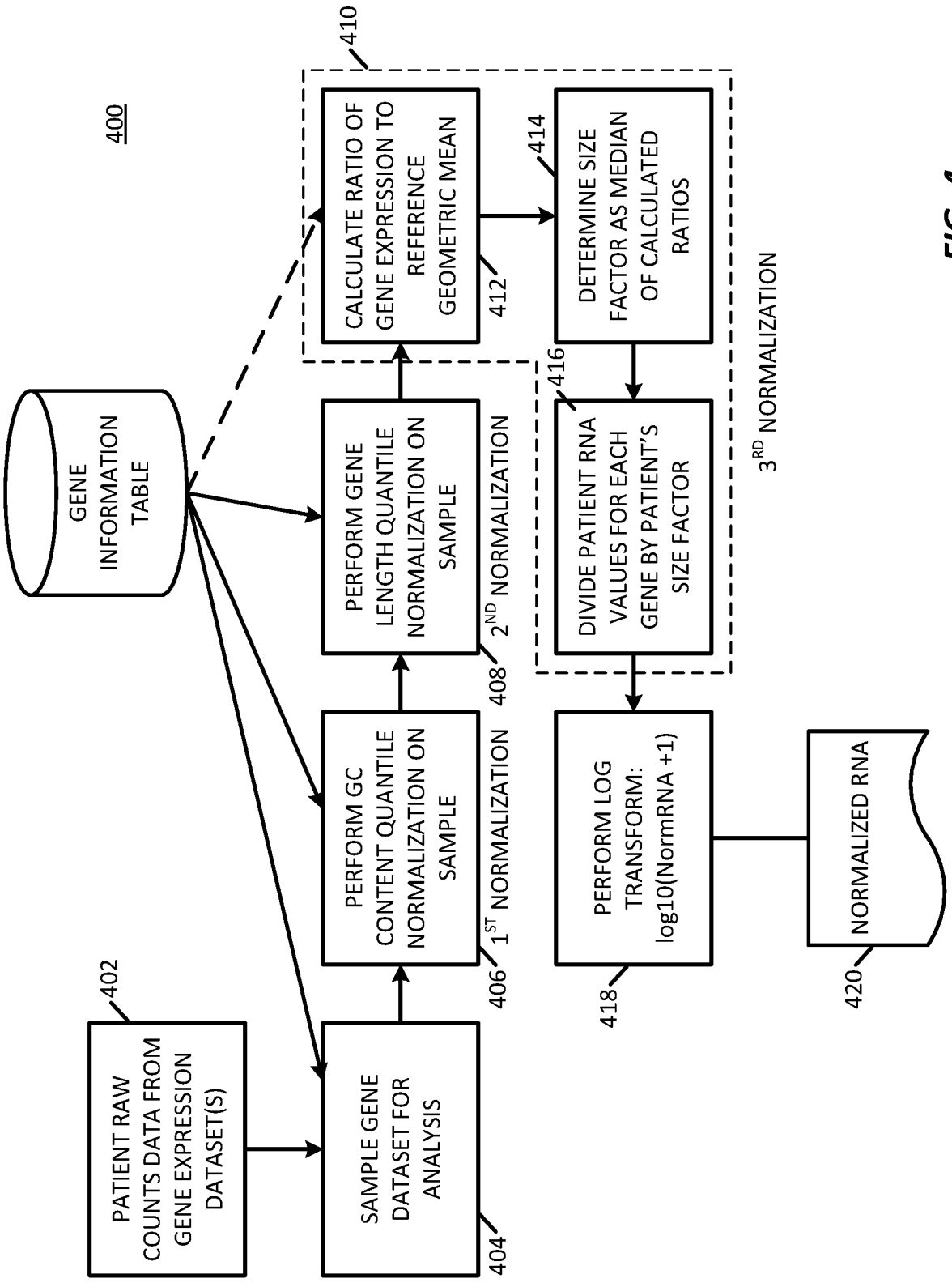

400
PATIENT RAW COUNTS DATA FROM GENE EXPRESSION DATASET(S)
402
SAMPLE GENE DATASET FOR ANALYSIS
404
PERFORM GC CONTENT QUANTILE NORMALIZATION ON SAMPLE
406
1ST NORMALIZATION
PERFORM GENE LENGTH QUANTILE NORMALIZATION ON SAMPLE
408
2ND NORMALIZATION
GENE INFORMATION TABLE
410
CALCULATE RATIO OF GENE EXPRESSION TO REFERENCE GEOMETRIC MEAN
412
DETERMINE SIZE FACTOR AS MEDIAN OF CALCULATED RATIOS
414
DIVIDE PATIENT RNA VALUES FOR EACH GENE BY PATIENT'S SIZE FACTOR
416
3RD NORMALIZATION
PERFORM LOG TRANSFORM: log10(NormRNA +1)
418
NORMALIZED RNA
420

FIG. 4

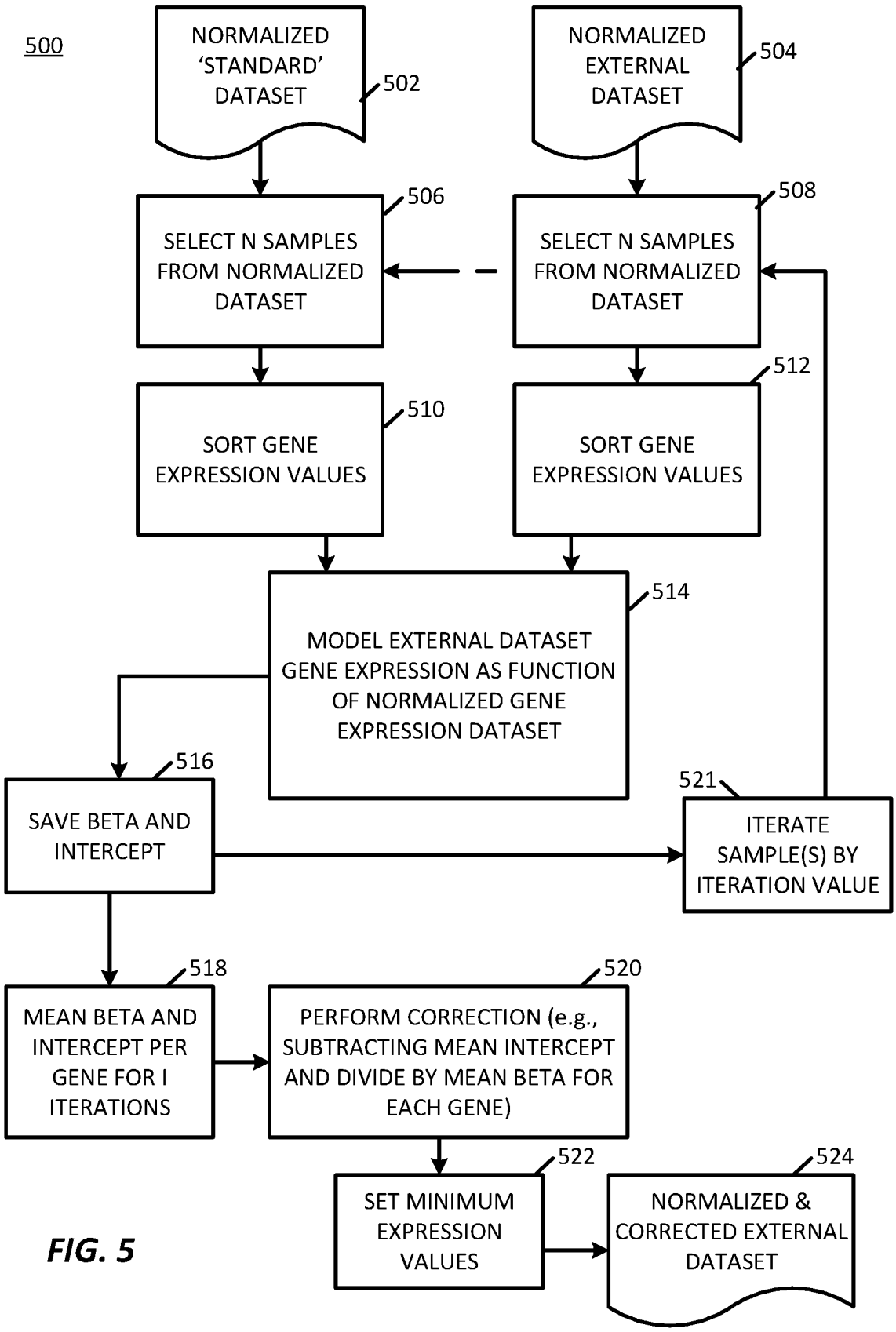

500
NORMALIZED 'STANDARD' DATASET
502
NORMALIZED EXTERNAL DATASET
504
SELECT N SAMPLES FROM NORMALIZED DATASET
506
SELECT N SAMPLES FROM NORMALIZED DATASET
508
SORT GENE EXPRESSION VALUES
510
SORT GENE EXPRESSION VALUES
512
MODEL EXTERNAL DATASET GENE EXPRESSION AS FUNCTION OF NORMALIZED GENE EXPRESSION DATASET
514
SAVE BETA AND INTERCEPT
516
ITERATE SAMPLE(S) BY ITERATION VALUE
521
MEAN BETA AND INTERCEPT PER GENE FOR I ITERATIONS
518
PERFORM CORRECTION (e.g., SUBTRACTING MEAN INTERCEPT AND DIVIDE BY MEAN BETA FOR EACH GENE)
520
SET MINIMUM EXPRESSION VALUES
522
NORMALIZED & CORRECTED EXTERNAL DATASET
524

FIG. 5

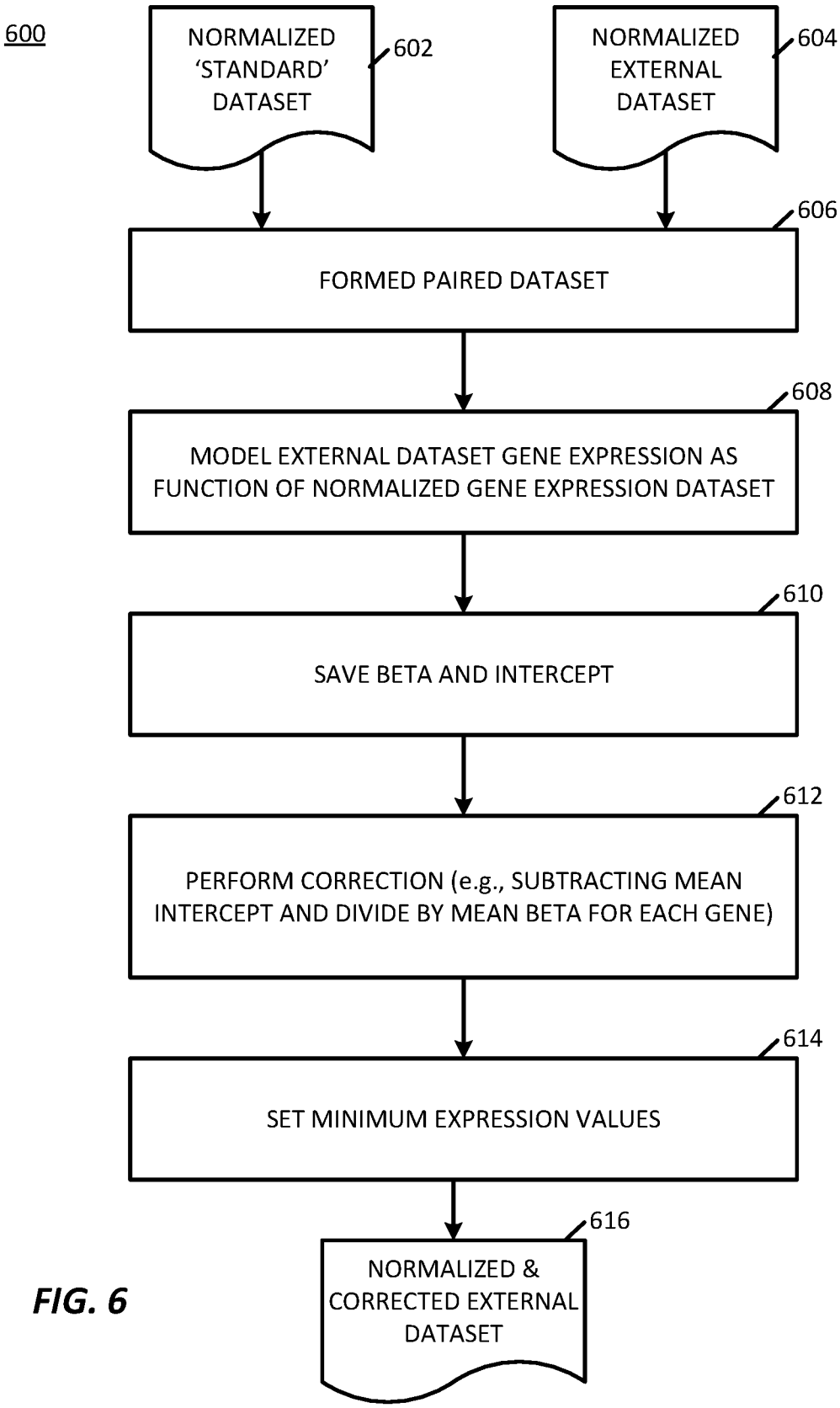

600
NORMALIZED 'STANDARD' DATASET
602
NORMALIZED EXTERNAL DATASET
604
606
FORMED PAIRED DATASET
608
MODEL EXTERNAL DATASET GENE EXPRESSION AS FUNCTION OF NORMALIZED GENE EXPRESSION DATASET
610
SAVE BETA AND INTERCEPT
612
PERFORM CORRECTION (e.g., SUBTRACTING MEAN INTERCEPT AND DIVIDE BY MEAN BETA FOR EACH GENE)
614
SET MINIMUM EXPRESSION VALUES
616
NORMALIZED & CORRECTED EXTERNAL DATASET

FIG. 6

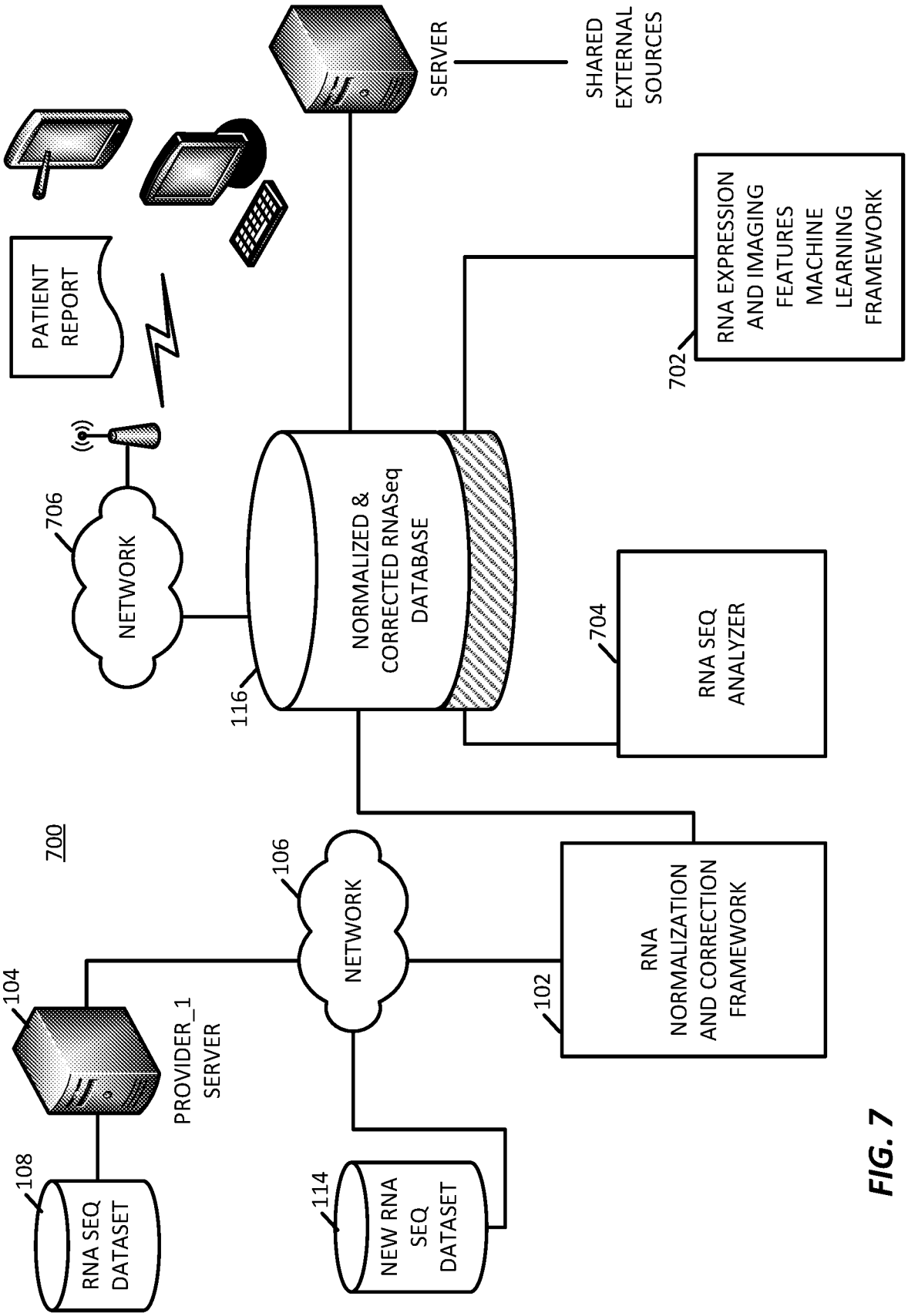

700
PROVIDER_1 SERVER
104
RNA SEQ DATASET
108
NETWORK
106
NEW RNA SEQ DATASET
114
RNA NORMALIZATION AND CORRECTION FRAMEWORK
102
RNA SEQ ANALYZER
704
NORMALIZED & CORRECTED RNASeq DATABASE
116
NETWORK
706
PATIENT REPORT
SERVER
SHARED EXTERNAL SOURCES
RNA EXPRESSION AND IMAGING FEATURES MACHINE LEARNING FRAMEWORK
702

FIG. 7

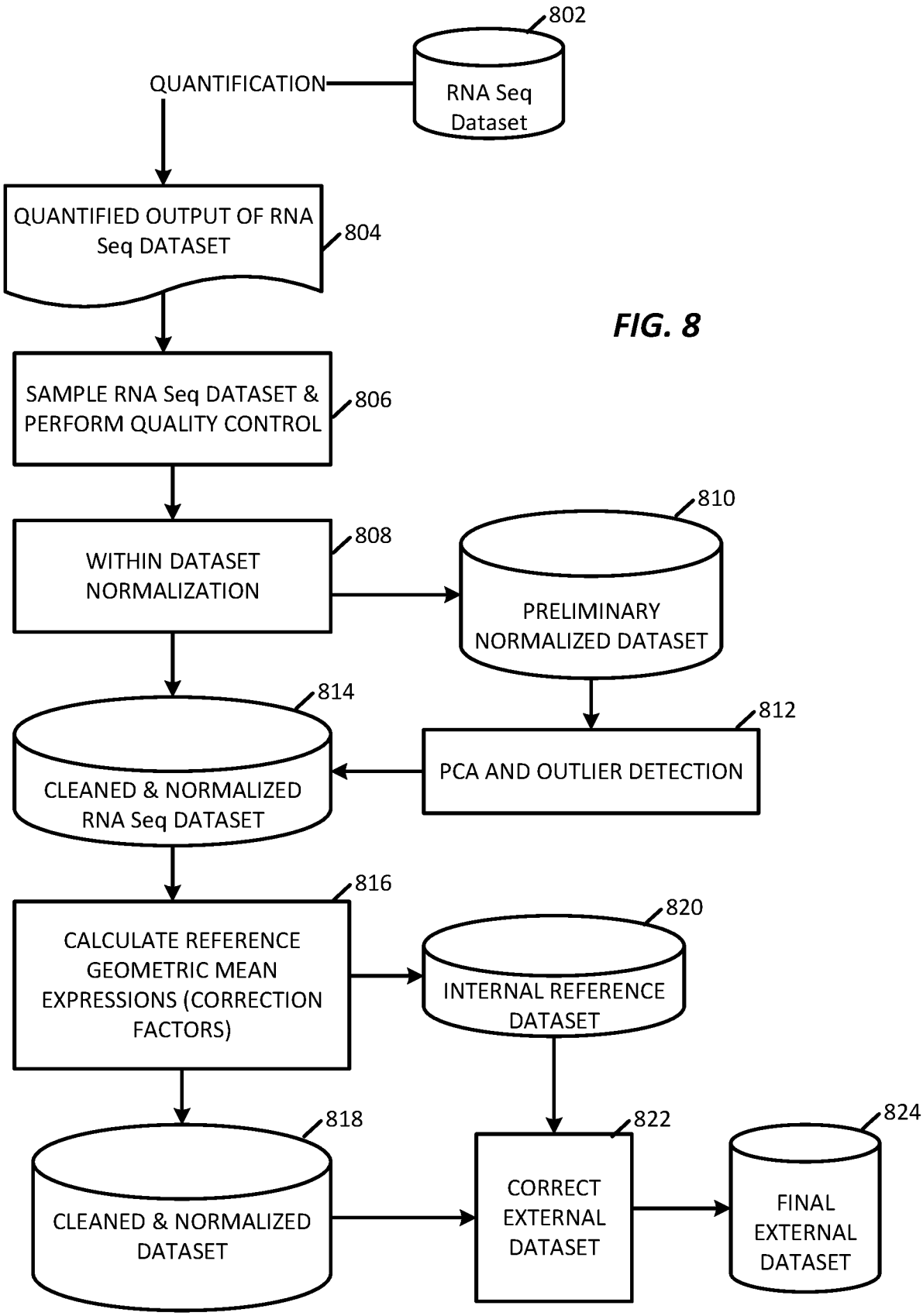

802
QUANTIFICATION
RNA Seq Dataset
QUANTIFIED OUTPUT OF RNA Seq DATASET
804
SAMPLE RNA Seq DATASET & PERFORM QUALITY CONTROL
806
WITHIN DATASET NORMALIZATION
808
810
PRELIMINARY NORMALIZED DATASET
814
CLEANED & NORMALIZED RNA Seq DATASET
812
PCA AND OUTLIER DETECTION
816
CALCULATE REFERENCE GEOMETRIC MEAN EXPRESSIONS (CORRECTION FACTORS)
820
INTERNAL REFERENCE DATASET
818
CLEANED & NORMALIZED DATASET
822
CORRECT EXTERNAL DATASET
824
FINAL EXTERNAL DATASET

FIG. 8

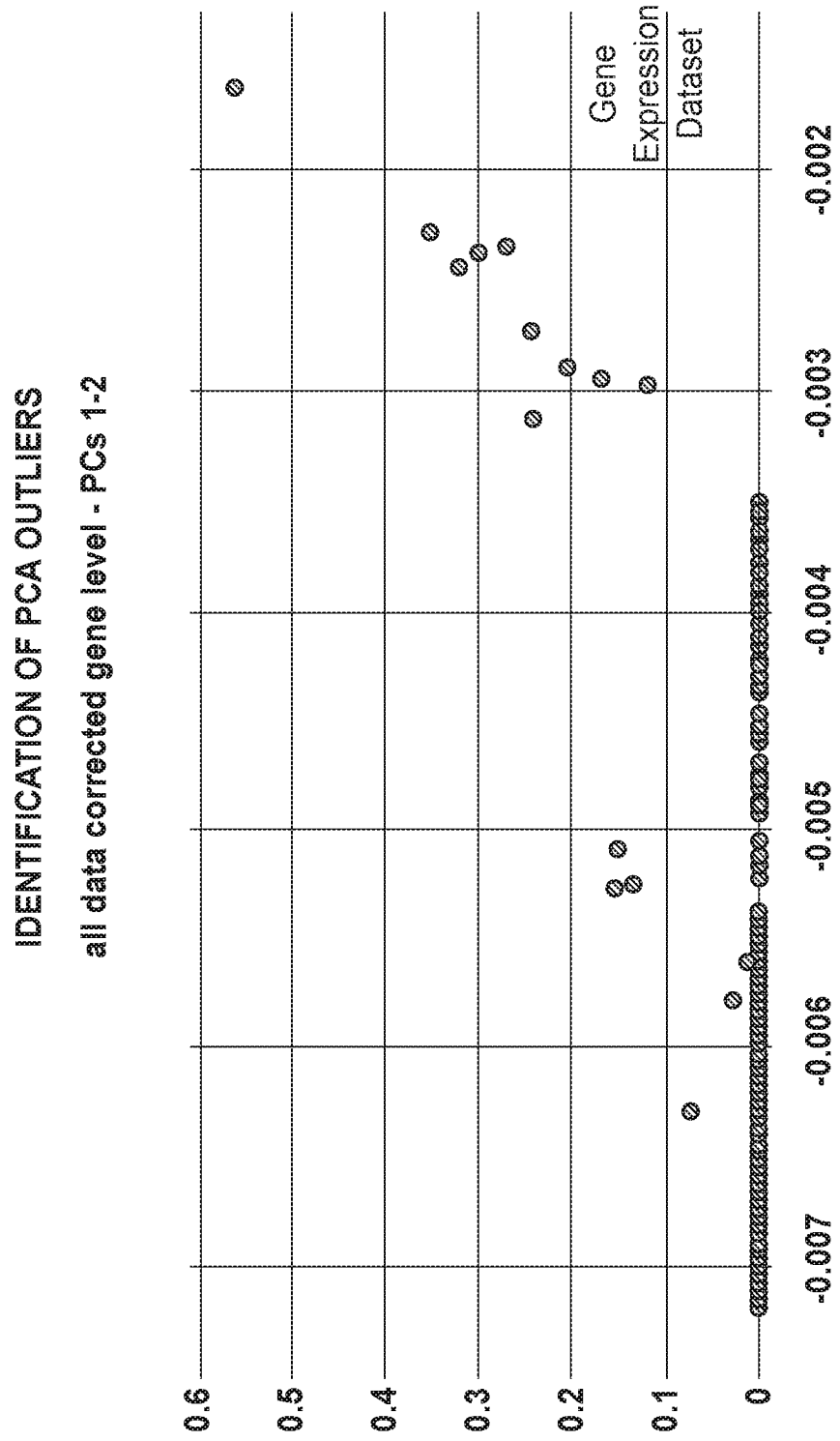

IDENTIFICATION OF PCA OUTLIERS
all data corrected gene level - PCs 1-2
Gene Expression Dataset
0.6
0.5
0.4
0.3
0.2
0.1
0
-0.007
-0.006
-0.005
-0.004
-0.003
-0.002

FIG. 9

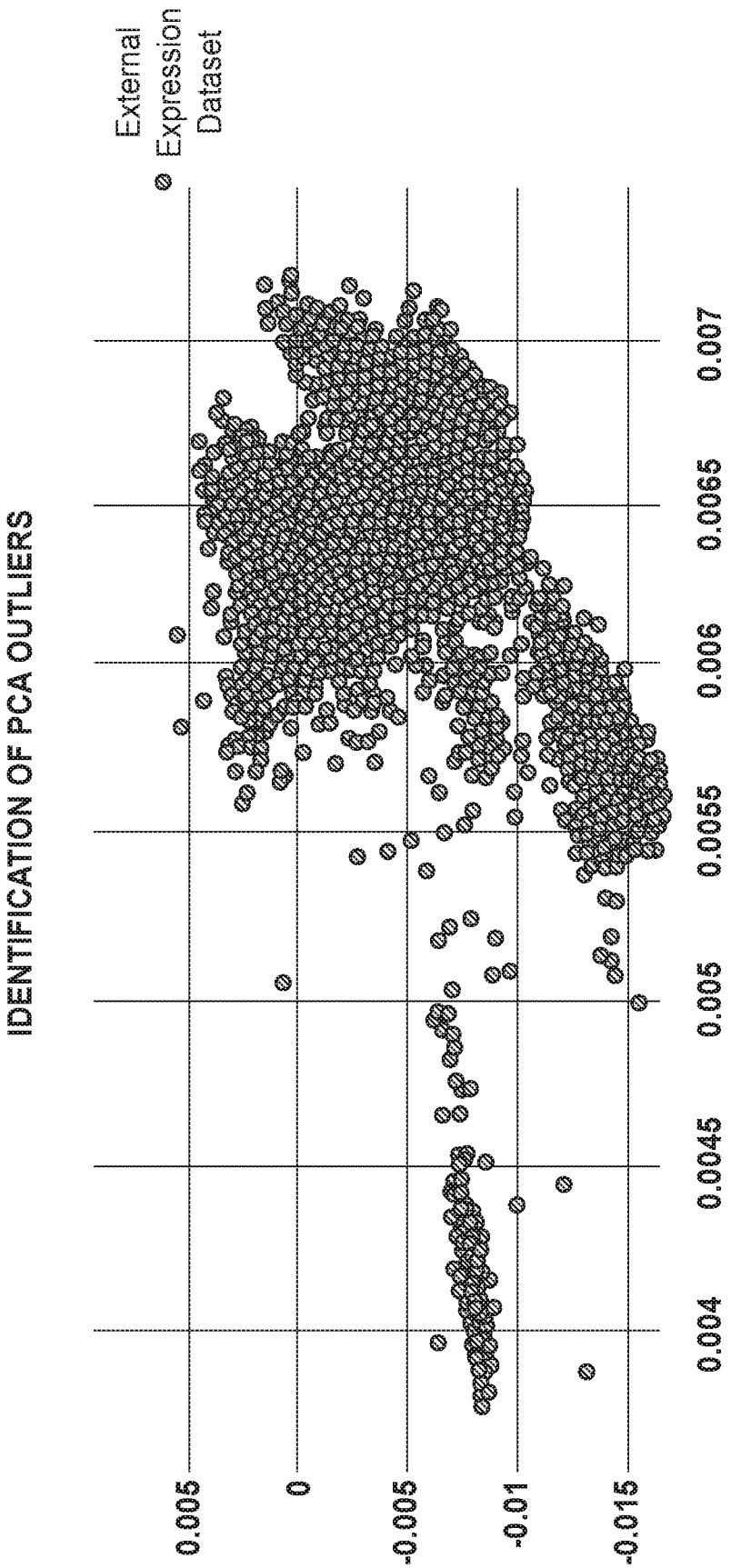

IDENTIFICATION OF PCA OUTLIERS
External Expression Dataset
0.005
0
-0.005
-0.01
-0.015
0.004
0.0045
0.005
0.0055
0.006
0.0065
0.007

FIG. 10

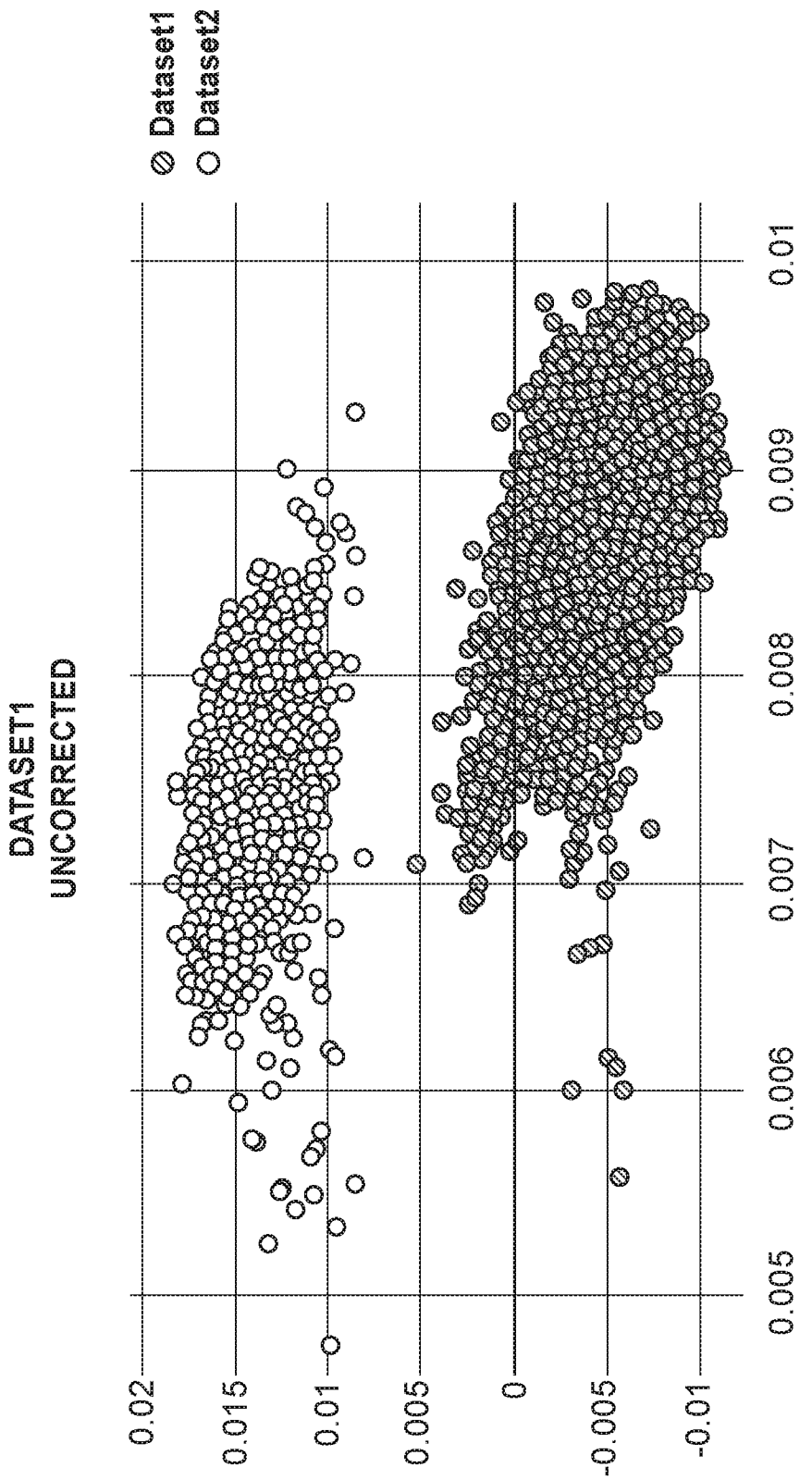

DATASET1
UNCORRECTED
Dataset1
Dataset2
0.02
0.015
0.01
0.005
0
-0.005
-0.01
0.005
0.006
0.007
0.008
0.009
0.01

FIG. 11A

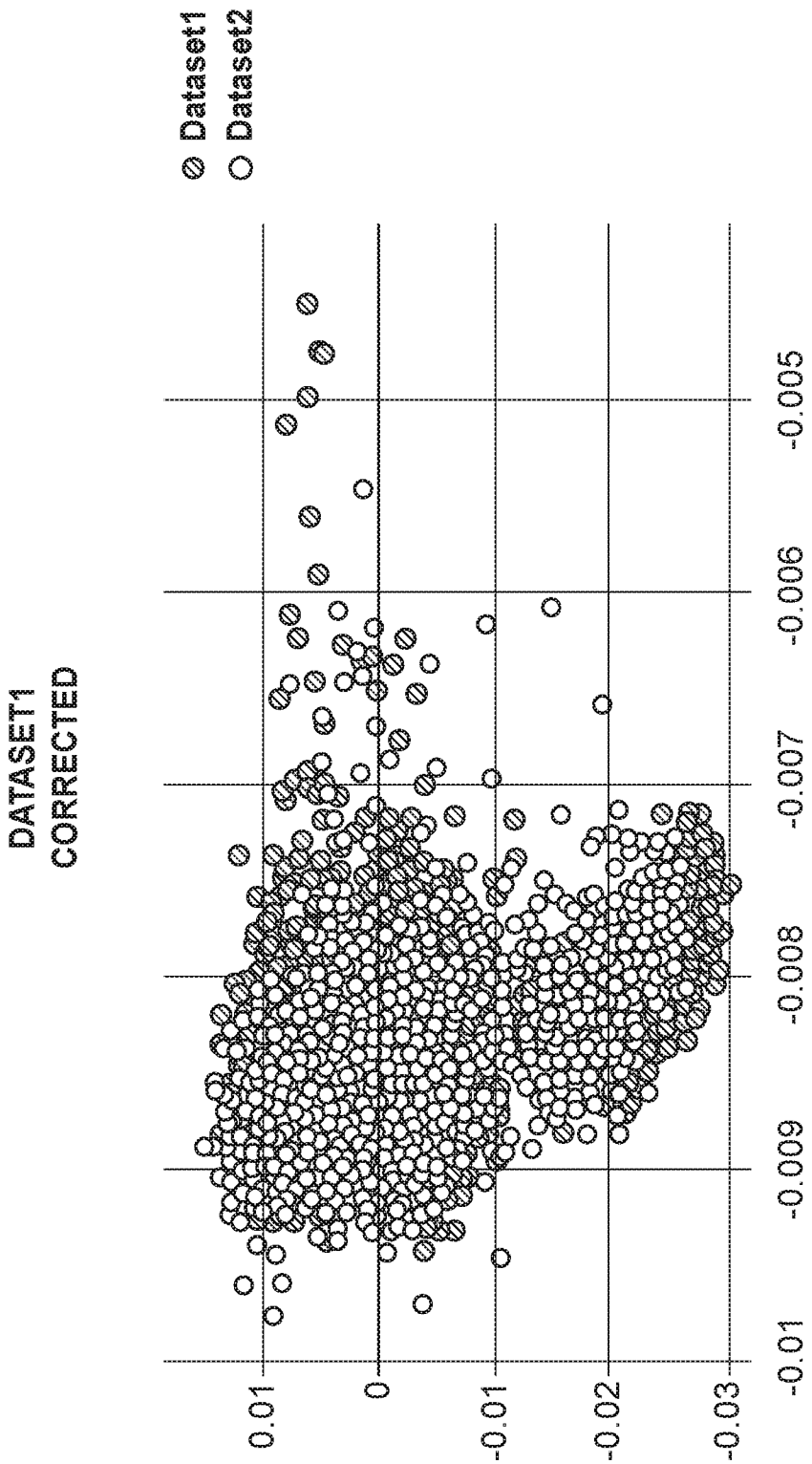

Dataset1
Dataset2
DATASET1
CORRECTED
0.01
0
-0.01
-0.02
-0.03
-0.005
-0.006
-0.007
-0.008
-0.009
-0.01

FIG. 11B

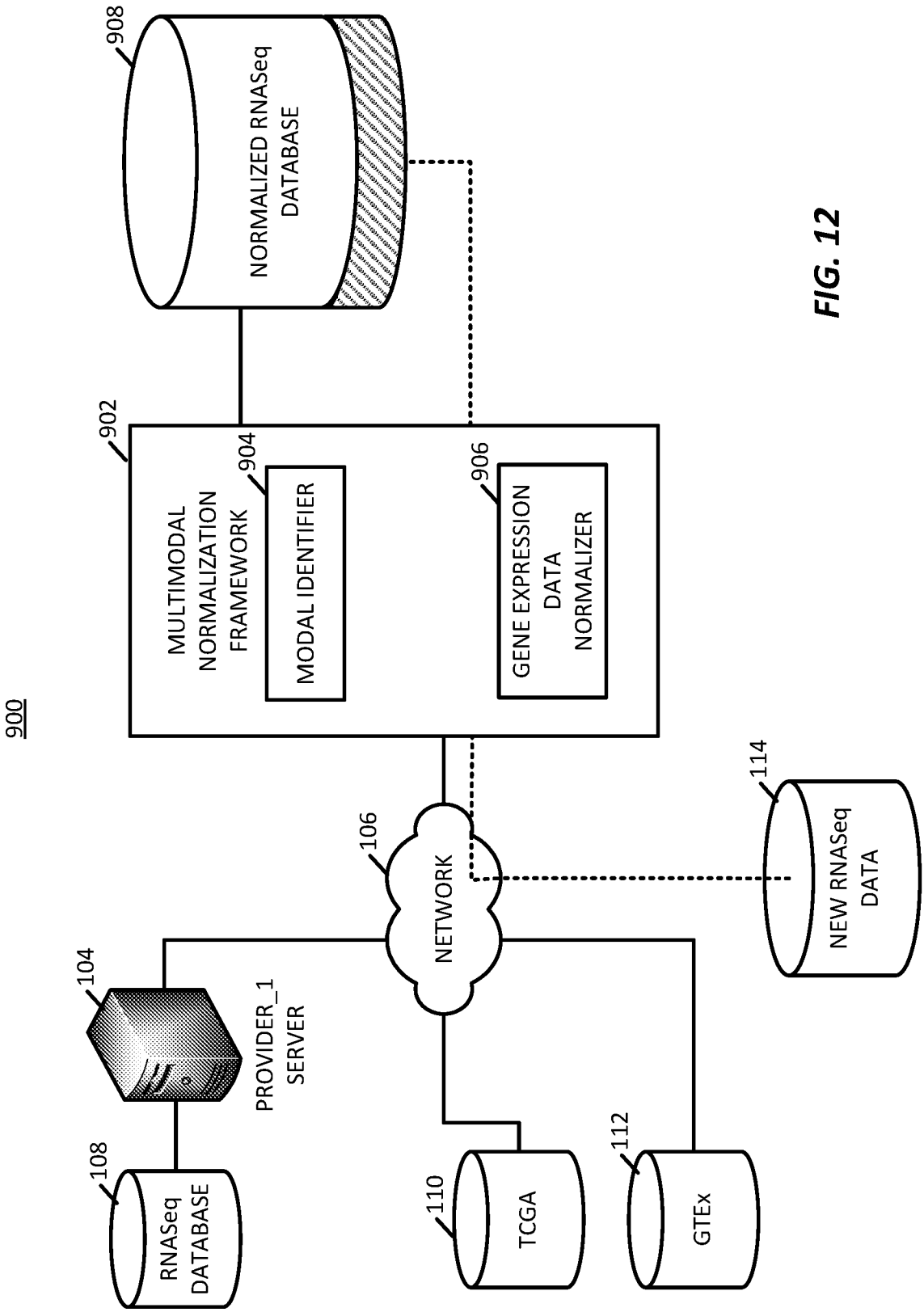

900
908
NORMALIZED RNASeq DATABASE
902
MULTIMODAL NORMALIZATION FRAMEWORK
904
MODAL IDENTIFIER
906
GENE EXPRESSION DATA NORMALIZER
106
NETWORK
114
NEW RNASeq DATA
104
PROVIDER_1 SERVER
108
RNASeq DATABASE
110
TCGA
112
GTEx

FIG. 12

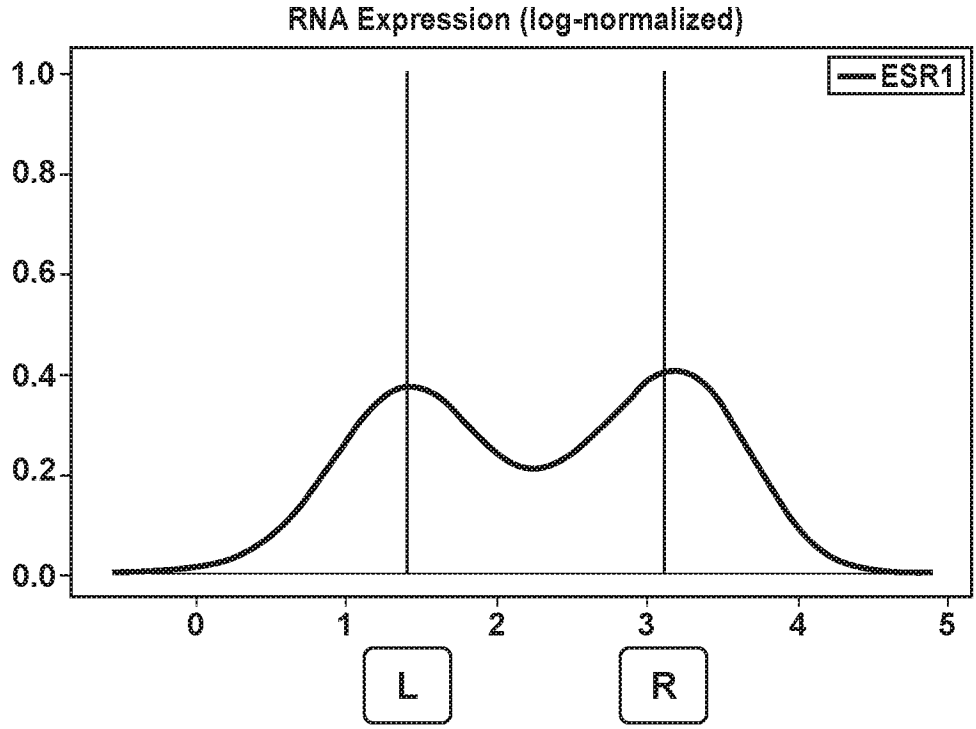

RNA Expression (log-normalized)
ESR1
1.0
0.8
0.6
0.4
0.2
0.0
0
1
2
3
4
5
L
R

FIG. 13

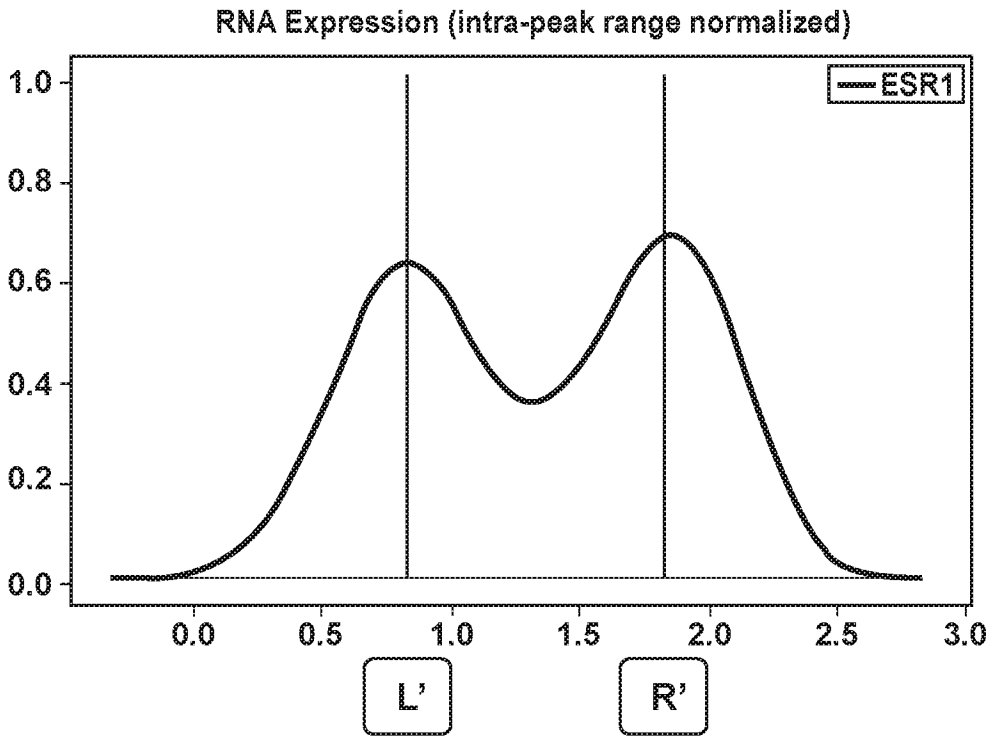

RNA Expression (intra-peak range normalized)
ESR1
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.5
1.0
1.5
2.0
2.5
3.0
L'
R'

FIG. 15

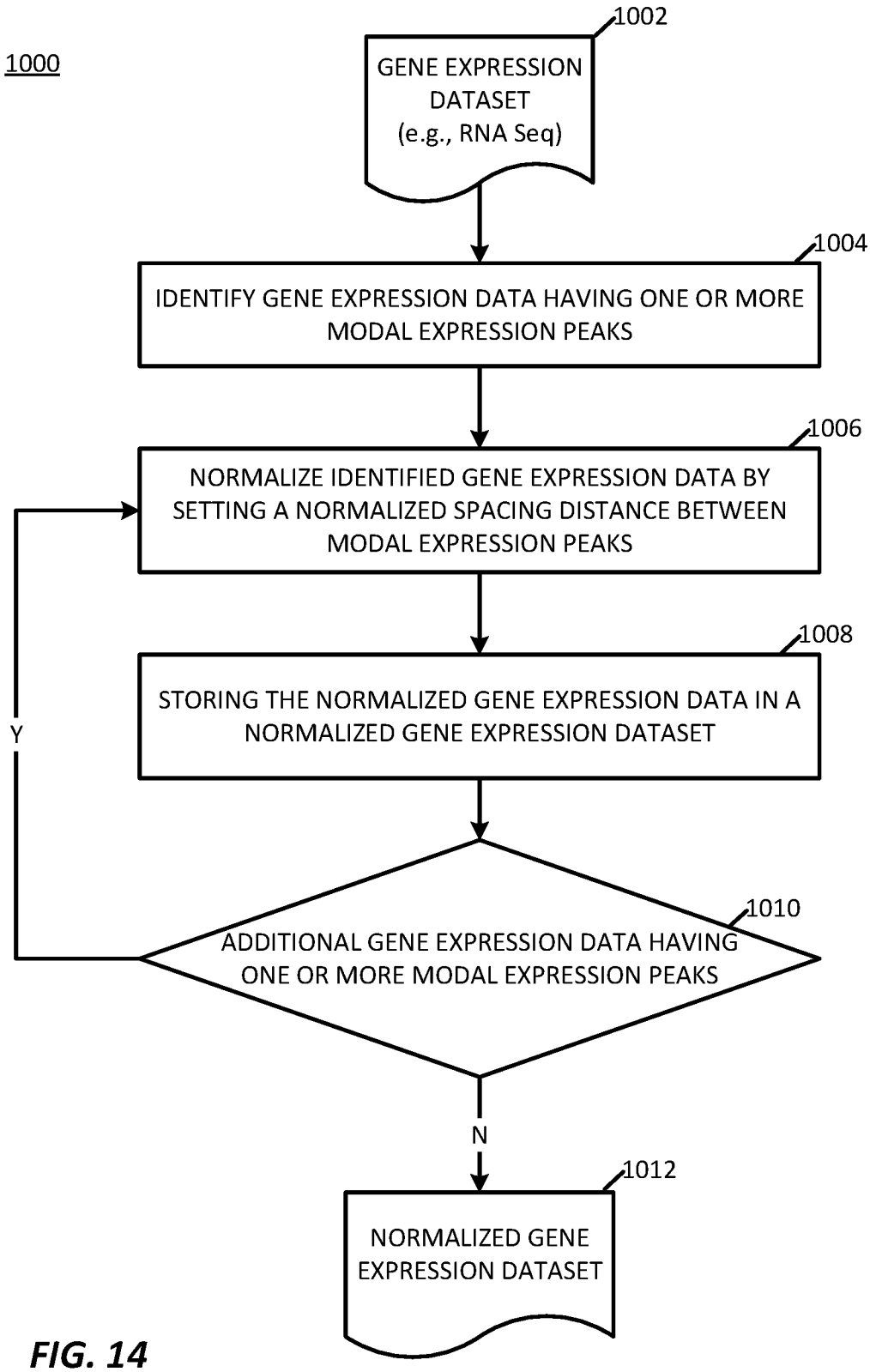

1000
1002
GENE EXPRESSION DATASET (e.g., RNA Seq)
1004
IDENTIFY GENE EXPRESSION DATA HAVING ONE OR MORE MODAL EXPRESSION PEAKS
1006
NORMALIZE IDENTIFIED GENE EXPRESSION DATA BY SETTING A NORMALIZED SPACING DISTANCE BETWEEN MODAL EXPRESSION PEAKS
1008
STORING THE NORMALIZED GENE EXPRESSION DATA IN A NORMALIZED GENE EXPRESSION DATASET
1010
ADDITIONAL GENE EXPRESSION DATA HAVING ONE OR MORE MODAL EXPRESSION PEAKS
Y
N
1012
NORMALIZED GENE EXPRESSION DATASET

FIG. 14

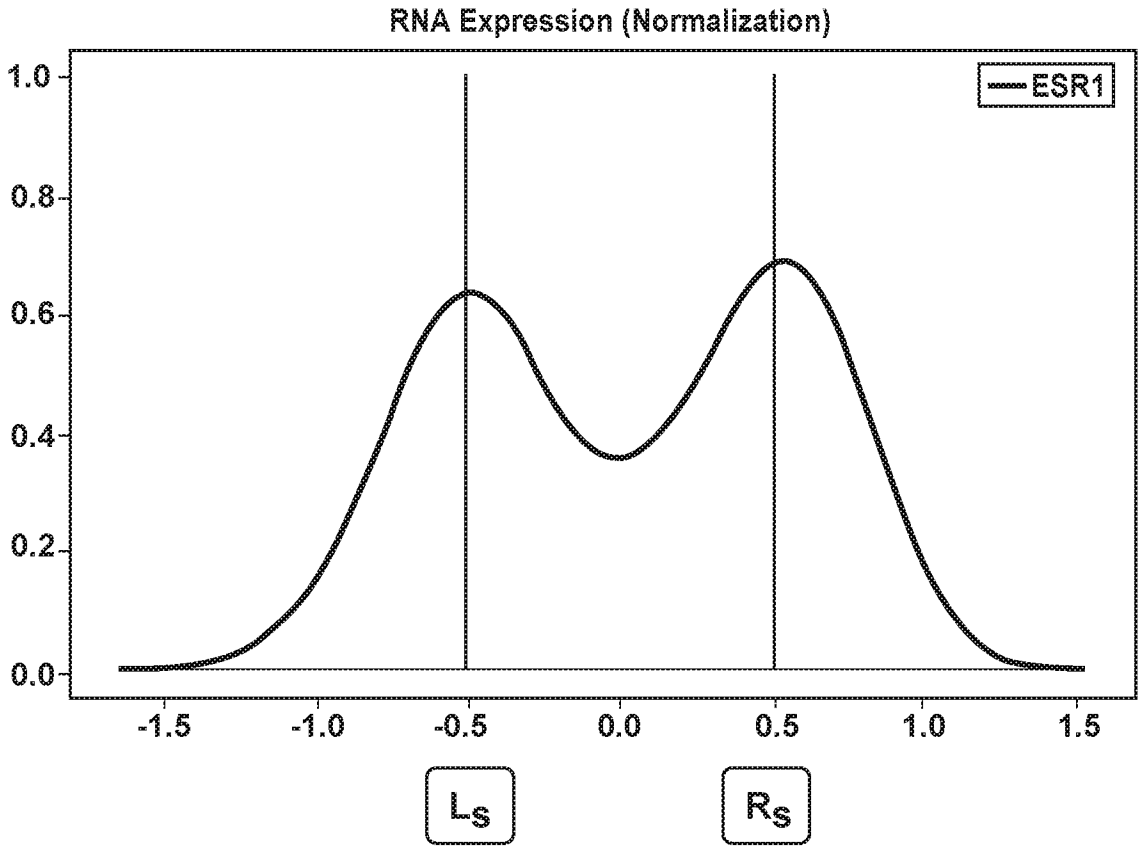

RNA Expression (Normalization)
ESR1
1.0
0.8
0.6
0.4
0.2
0.0
-1.5
-1.0
-0.5
0.0
0.5
1.0
1.5
Ls
Rs

FIG. 16

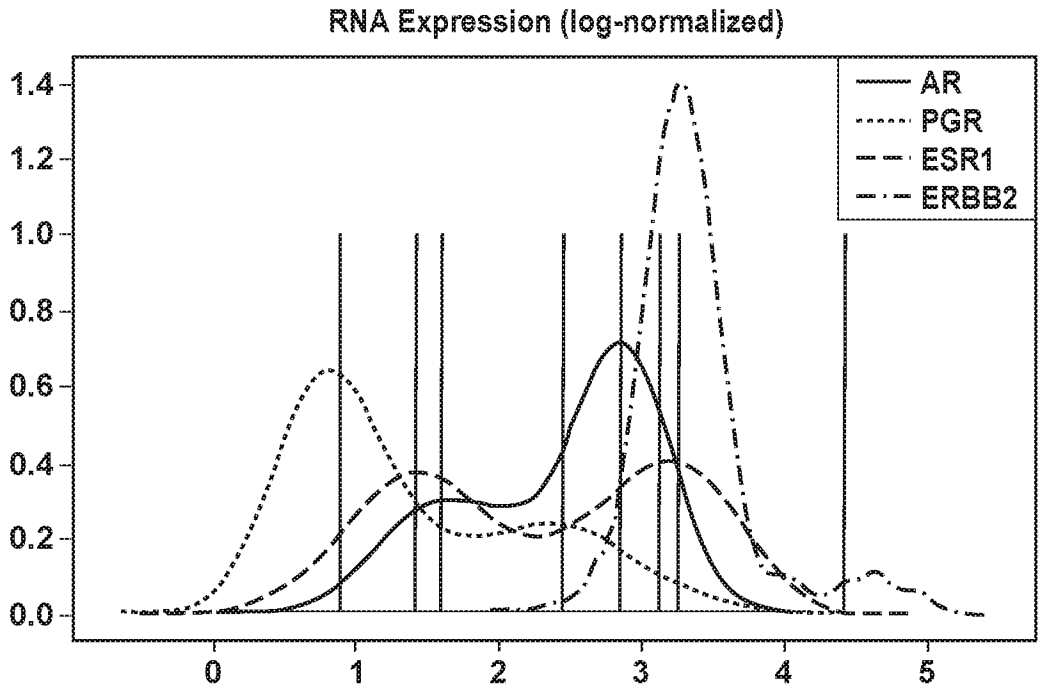


FIG. 17A

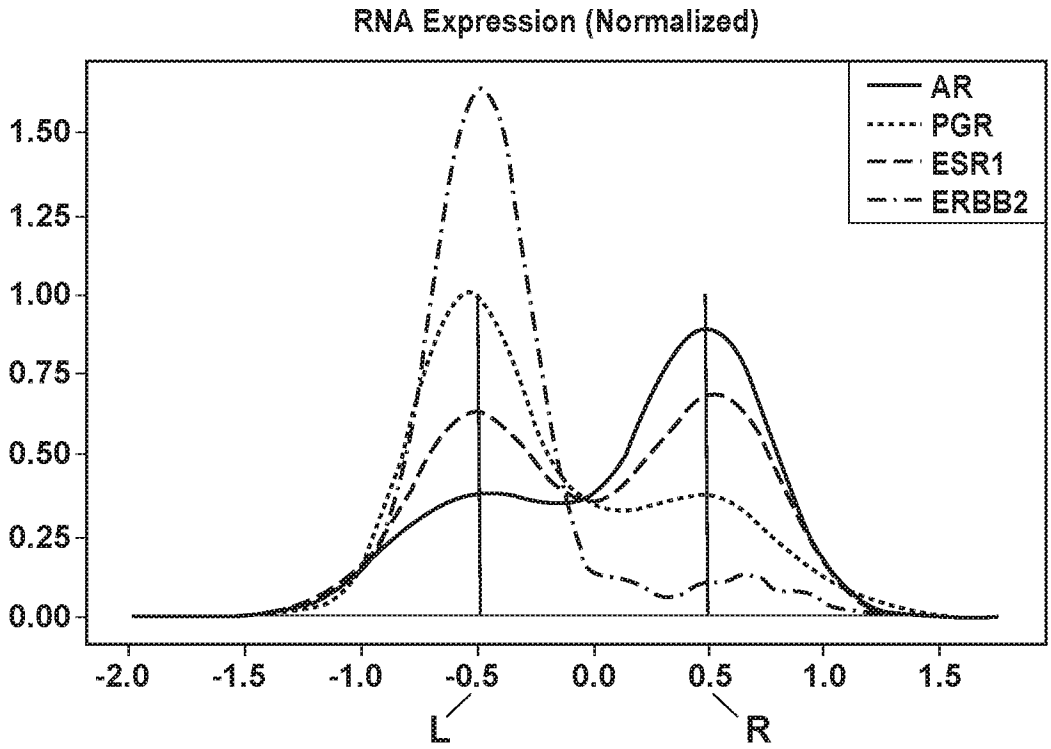


FIG. 17B

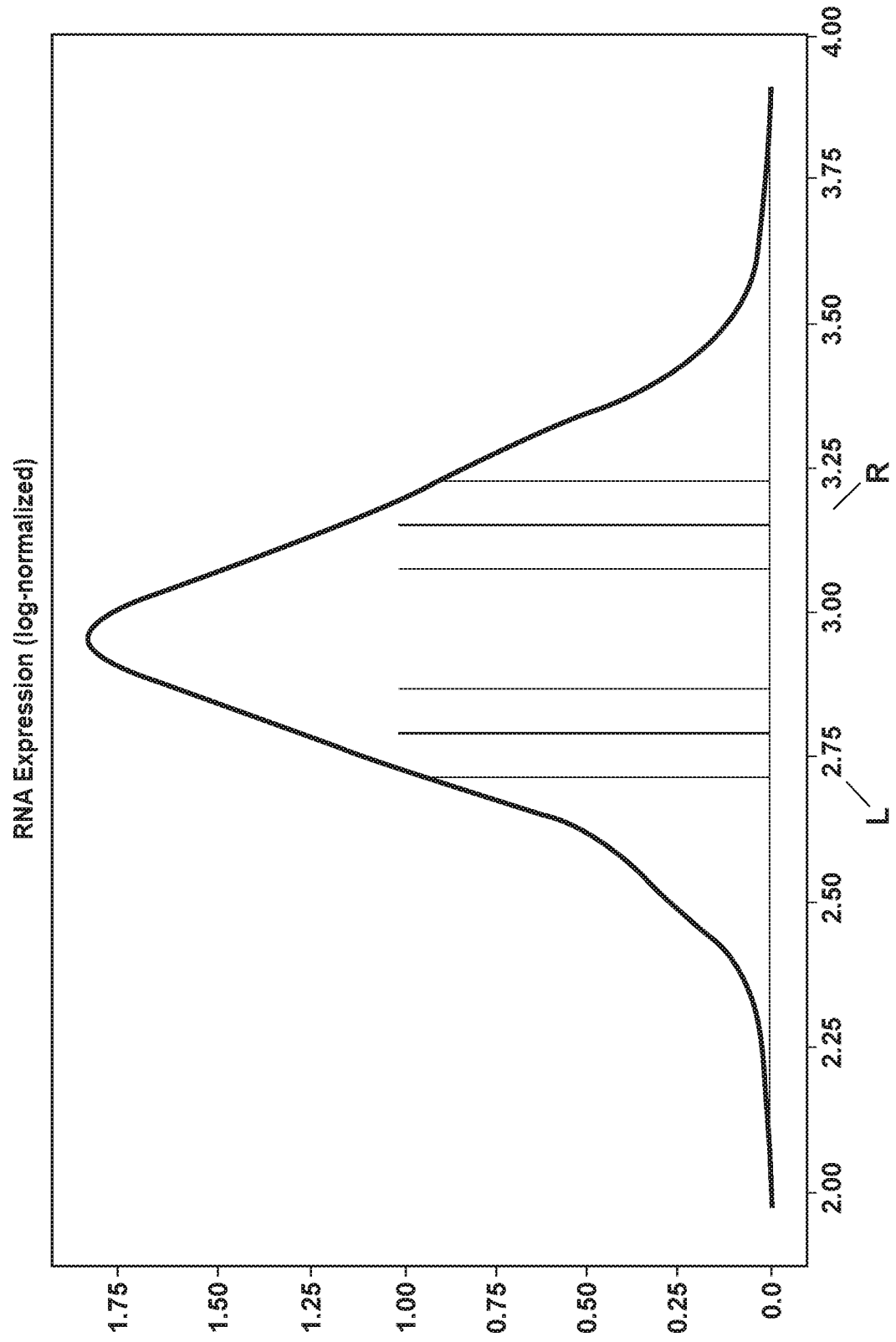

RNA Expression (log-normalized)
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.0
2.00
2.25
2.50
2.75
3.00
3.25
3.50
3.75
4.00
L
R

FIG. 18

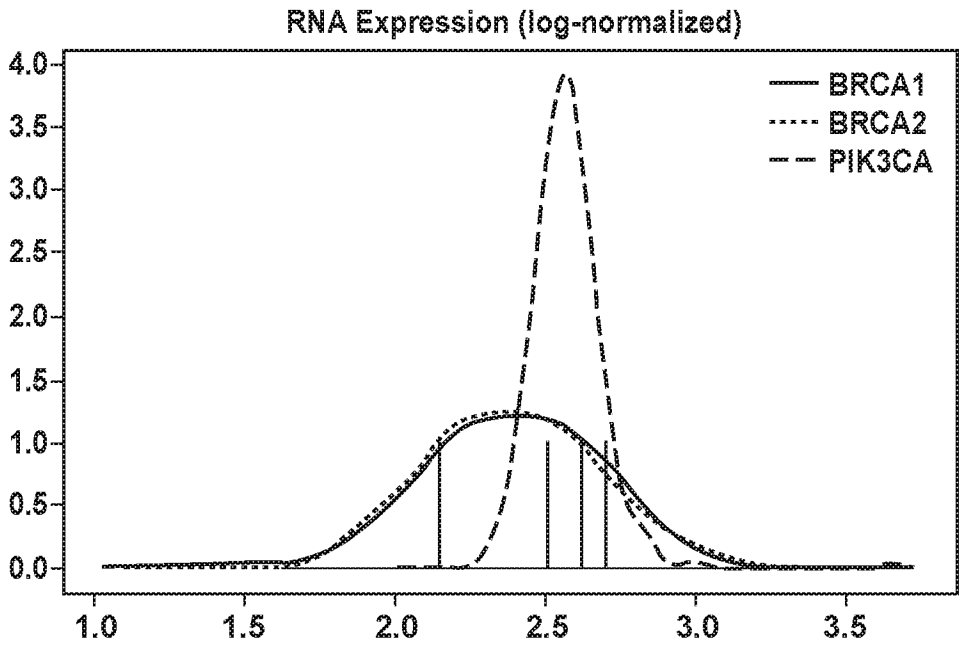


FIG. 19A

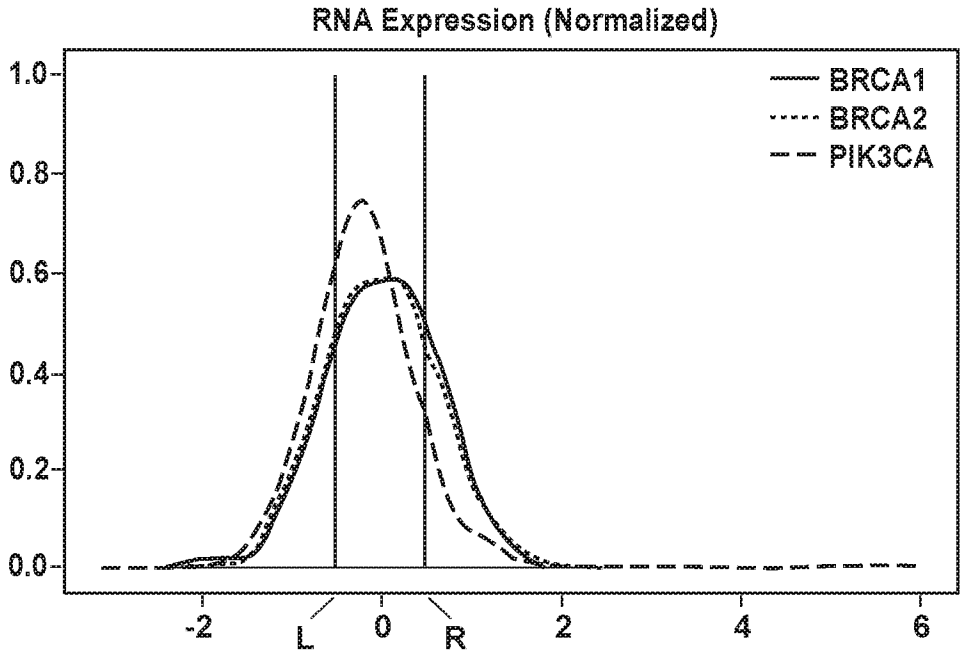


FIG. 19B

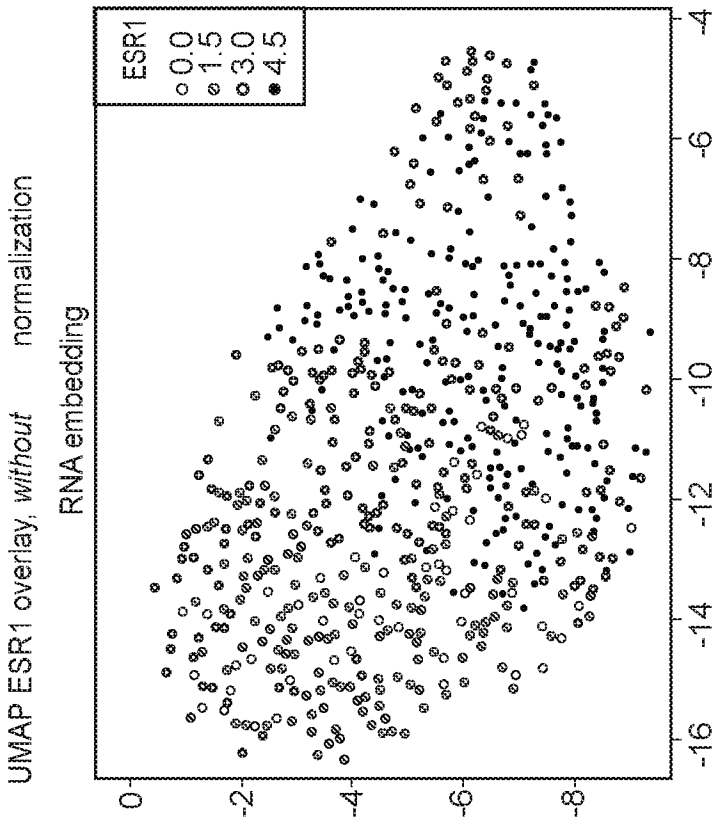


FIG. 20B

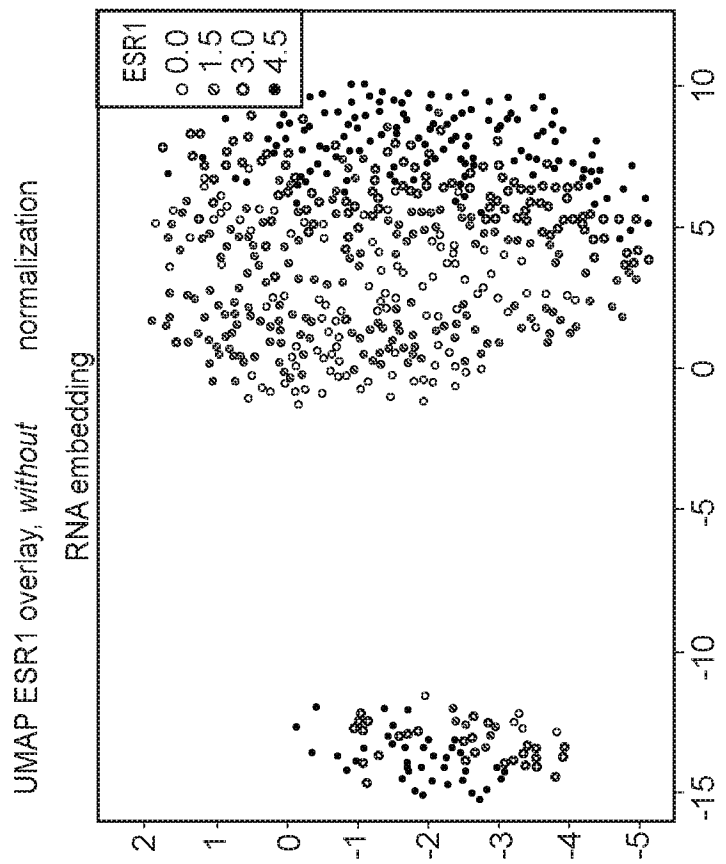
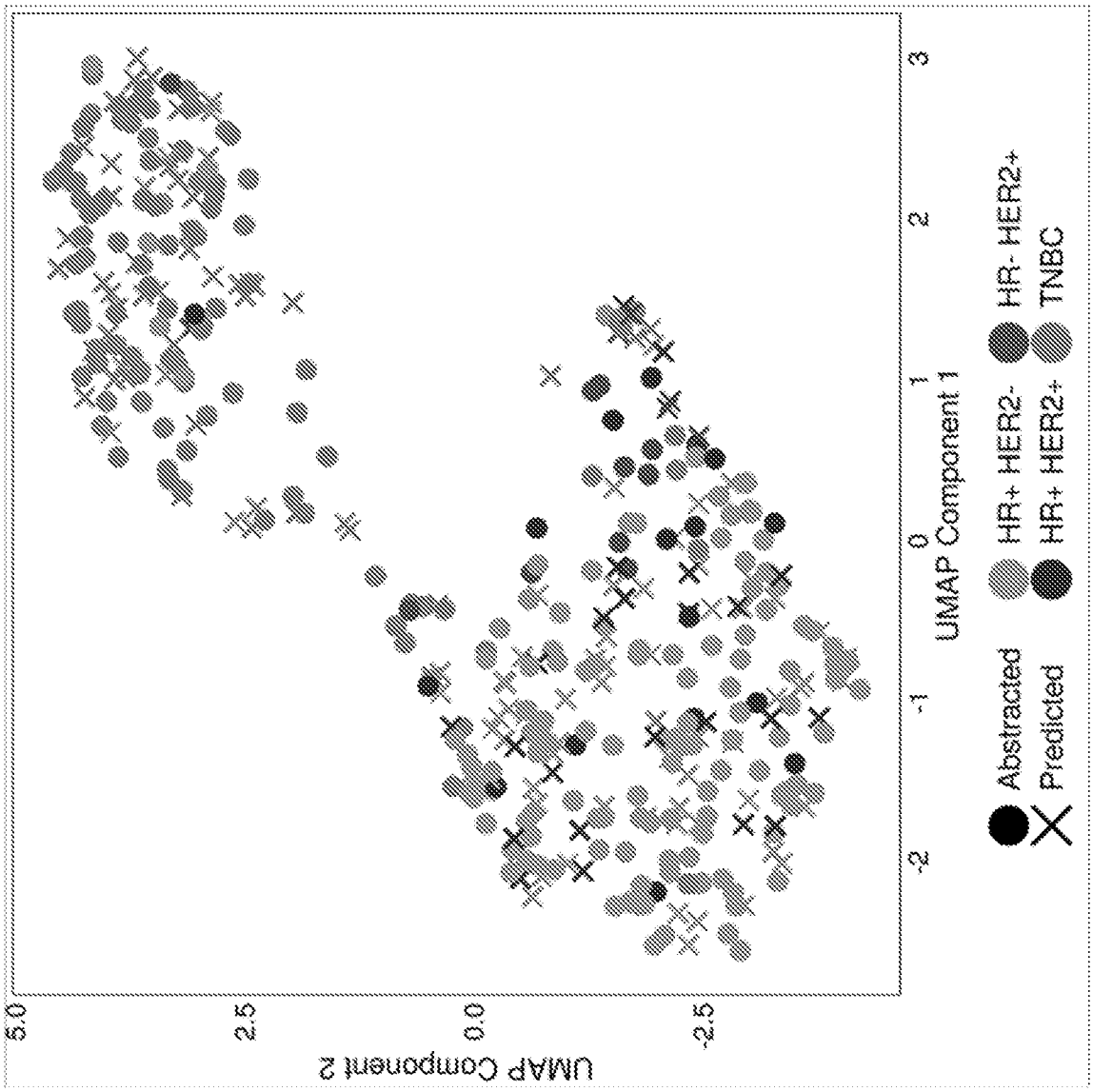


FIG. 20A

FIG. 21


UMAP Component 2
5.0
2.5
0.0
-2.5
UMAP Component 1
-2
-1
0
1
2
3
Abstracted
Predicted
HR+ HER2-
HR+ HER2+
HR- HER2+
TNBC

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US2019/052801

A. CLASSIFICATION OF SUBJECT MATTER

IPC(8) - C12Q 1/6869; C12Q 1/6886; G06F 19/20; G06F 19/22 (2019.01)

CPC - C12Q 1/6869; C12Q 1/68; C12Q 1/6883; C12Q 1/6886; C12Q 2600/158; G16B 25/10; G16B 30/00 (2019.08)

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

USPC - 435/6; 506/16; 702/19; 702/20 (keyword delimited)

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 2016/0224724 A1 (SEQUENOM INC) 04 August 2016 (04.08.2016) entire document	1-3, 9, 12-17, 19, 20
---		---
Y		4, 8, 10, 11, 18
Y	US 2016/0355806 A1 (THE GENERAL HOSPITAL CORPORATION) 08 December 2016 (08.12.2016) entire document	4, 8, 18
Y	US 2014/0315199 A1 (LIFE TECHNOLOGIES CORPORATION) 23 October 2014 (23.10.2014) entire document	10
Y	US 2015/0119262 A1 (CASE WESTERN RESERVE UNIVERSITY) 30 April 2015 (30.04.2015) entire document	11
A	FILLOUX et al. An integrative method to normalize RNA-Seq data. 14 June 2014 (14.06.2014). [retrieved on 09.01.2020]. Retrieved from the Internet: <URL: https://bmcbioinformatics.biomedcentral.com/track/pdf/10.1186/1471-2105-15-188> entire document	1-20
A	US 2018/0010136 A1 (HUNT, III et al) 11 January 2018 (11.01.2018) entire document	1-20
A	US 2016/0186262 A1 (REPRODUCTIVE GENETICS AND TECHNOLOGY SOLUTIONS, LLC) 30 June 2016 (30.06.2016) entire document	1-20
A	US 2014/0193818 A1 (VERINATA HEALTH INC) 10 July 2014 (10.07.2014) entire document	1-20

☐ 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	Date of mailing of the international search report
09 January 2020	28 JAN 2020
Name and mailing address of the ISA/US	**Authorized officer**
Mail Stop PCT, Attn: ISA/US, Commissioner for Patents P.O. Box 1450, Alexandria, VA 22313-1450 Facsimile No. 571-273-8300	Blaine R. Copenheaver PCT Helpdesk: 571-272-4300 PCT OSP: 571-272-7774

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US2019/052801

Box No. II Observations where certain claims were found unsearchable (Continuation of item 2 of first sheet)

This international search report has not been established in respect of certain claims under Article 17(2)(a) for the following reasons:

1. ☐ Claims Nos.:
because they relate to subject matter not required to be searched by this Authority, namely:

2. ☐ Claims Nos.:
because they relate to parts of the international application that do not comply with the prescribed requirements to such an extent that no meaningful international search can be carried out, specifically:

3. ☐ Claims Nos.:
because they are dependent claims and are not drafted in accordance with the second and third sentences of Rule 6.4(a).

Box No. III Observations where unity of invention is lacking (Continuation of item 3 of first sheet)

This International Searching Authority found multiple inventions in this international application, as follows:
See extra sheet(s).

1. ☐ As all required additional search fees were timely paid by the applicant, this international search report covers all searchable claims.

2. ☐ As all searchable claims could be searched without effort justifying additional fees, this Authority did not invite payment of additional fees.

3. ☐ As only some of the required additional search fees were timely paid by the applicant, this international search report covers only those claims for which fees were paid, specifically claims Nos.:

4. ☒ No required additional search fees were timely paid by the applicant. Consequently, this international search report is restricted to the invention first mentioned in the claims; it is covered by claims Nos.:
1-20

Remark on Protest

☐ The additional search fees were accompanied by the applicant's protest and, where applicable, the payment of a protest fee.

☐ The additional search fees were accompanied by the applicant's protest but the applicable protest fee was not paid within the time limit specified in the invitation.

☐ No protest accompanied the payment of additional search fees.

Continued from Box No. III Observations where unity of invention is lacking

This application contains the following inventions or groups of inventions which are not so linked as to form a single general inventive concept under PCT Rule 13.1. In order for all inventions to be examined, the appropriate additional examination fees must be paid.

Group I, claims 1-20, drawn to a computer-implemented method comprising: generating, from a comparison of a normalized RNA sequence dataset against a standard RNA sequence dataset, at least one conversion factor for applying to a next RNA sequence dataset.

Group II, claims 21-32, drawn to a computer-implemented method comprising: identifying within the gene expression dataset, using a regression technique implemented by the one or more processors.

The inventions listed as Groups I-II do not relate to a single general inventive concept under PCT Rule 13.1 because, under PCT Rule 13.2, they lack the same or corresponding special technical features for the following reasons: the special technical feature of the Group I invention: generating, from a comparison of a normalized RNA sequence dataset against a standard RNA sequence dataset, at least one conversion factor for applying to a next RNA sequence dataset; and correcting RNA sequence data of the next RNA sequence dataset using the at least one conversion factor as claimed therein is not present in the invention of Group II. The special technical feature of the Group II invention: identifying within the gene expression dataset, using a regression technique implemented by the one or more processors, expression data having multiple modal expression peaks; for the expression data, normalizing, using the one or more processors, a spacing between each of the multiple model expression peaks; and storing the normalized expression data in a normalized dataset as claimed therein is not present in the invention of Group I.

Groups I and II lack unity of invention because even though the inventions of these groups require the technical feature of a computer-implemented method comprising: a RNA sequence dataset, this technical feature is not a special technical feature as it does not make a contribution over the prior art.

Specifically, US 2018/0010136 to Hunt, III et al. teaches a computer-implemented method comprising: a RNA sequence dataset (para. [0131], [0139], [0364], [0373]).

Since none of the special technical features of the Group I or II inventions are found in more than one of the inventions, unity of invention is lacking.