SOFTWARE

Granatum: a graphical single-cell RNA-seq analysis pipeline for genomics scientists

Xun Zhu[1,2], Thomas Wolfgruber[1,2], Austin Tasato[3], Lana X Garmire[1, 2*]

*Correspondence:
LGarmire@cc.hawaii.edu
[1]Graduate Program in Molecular Biology and Bioengineering, University of Hawaii at Manoa, Honolulu, HI 96816
[2]Epidemiology Program, University of Hawaii Cancer Center, Honolulu, HI 96813
[3]Department of Electrical Engineering, University of Hawaii at Manoa, Honolulu, HI 96816

Abstract

Background: Single-cell RNA sequencing (scRNA-seq) is an increasingly popular platform to study heterogeneity at the single cell level. Computational methods to process scRNA-seq have limited accessibility to bench scientists, as they require significant amount of bioinformatics skills.

Results: We have developed Granatum, a web browser based scRNA-seq analysis pipeline to make analysis more broadly accessible to researchers. Without a single line of programming code, a user can click through the pipeline, setting parameters and visualizing results via the interactive graphical interface. The pipeline conveniently walks the users through various steps of scRNA-seq analysis. It has a comprehensive list of modules, including plate merging and batch effect removal, outlier sample removal, gene filtering, gene expression normalization, cell clustering, differential gene expression analysis, pathway/ontology enrichment analysis, protein network interaction visualization, and pseudo-time cell series construction.

Conclusions: Granatum enables much widely adoption of scRNA-seq technology by empowering the bench scientists with an easy to use graphical interface for scRNA-seq data analysis. The code is freely available for research use at: http://garmiregroup.org/granatum/code

Keywords: single-cell; gene expression; graphical; normalization; clustering; differential expression; pathway; pseudo-time; software

Background

The arrival of single-cell high-throughput RNA sequencing (scRNA-seq) has provided new opportunities for researchers to identify the expression characteristics of individual cells among complex tissues. This is a significant leap forward from bulk cell RNA expression analysis. In cancer, for example, scRNA-seq allows tumorous cells to be separated apart from healthy cells [1] and primary cells be differentiated from metastatic cells [2]. Single-cell expression data can also be used to describe trajectories of cell differentiation and development [3]. However, analyzing data from scRNA-seq brings new computational challenges, e.g., accounting for inherently high drop-out (artificial loss of RNA expression information) [4].

Software that has been developed to address these challenges may have very limited accessibility for biologists with only general computer skills, as they typically require the ability to use a computing language like R [5,6]. Other existing workflows that can be used to analyze scRNA-seq data, such as Singular (Fluidigm, Inc., South San Francisco, CA, USA), Cell Ranger/ Loupe (Pleasanton, CA, USA), and Scater [7] all require some non-graphical interactions and they may not provide a comprehensive set of scRNA-seq analysis methods. To fill this gap, we have developed Granatum, a fully interactive graphical scRNA-seq analysis tool. Granatum is the Latin word for pomegranate, which bears many seeds, resembling single cells within the entity. This tool employs an easy-to-use web browser interface for a wide range of methods suitable for scRNA-seq analysis: removal of batch effects, removal of outlier cells, normalization of expression levels, filtering of under-informative genes, clustering of cells, identification of differentially expressed genes, identification of enriched pathways/ontologies, visualization of protein networks, and reconstruction of pseudo-time paths for cells. Our software will empower a much broader

audience of research communities to study single cell complexity, by allowing them to readily explore single-cell expression data from a graphical user interface.

Implementation

Overview

Both the front-end and the back-end of Granatum are written in the R software language, and built with the Shiny framework [8]. Multiple concurrent users are handled by Shiny and each user works on its own data space. To protect the privacy of users, the data submitted by one user is not visible to any other user. The front-end is implemented as a web page with dynamically loaded pages, and is arranged in a step-wise fashion. The default theme uses the Bootstrap framework. ShinyJS [9] is used to power some of the interactive components. To allow users to redo a task, each processing step is equipped with a reset button.

Interactive widgets

The package visNetwork is used for the layout and physics simulation of the network modules [10]. DataTables are used to preview user submitted data and to show tabular data in various modules [11]. Plotly is used for the interactive outlier identification step [12]. The package ggplot2 is used for the scatter-plots and box-plots, which is also used by the Monocle package for the Pseudo-time construction step [3,13].

Back-end variable management

The expression matrix and the metadata sheet are stored separately for each user. The metadata sheet can refer to groups, batches, or other properties of the samples in the corresponding

expression matrix. These two types of tables are shared across all modules. Other variables shared across all modules include the log-transformed expression matrix, the filtered and normalized expression matrix, the dimensionally reduced matrix, species (human or mouse) and the primary metadata column.

Deployment

Granatum is deployed from a pre-configured VirtualBox Appliance (machine image), which is configured with all tool files and dependencies. VirtualBox is an open-source hypervisor developed by the Oracle Corporation – https://www.virtualbox.org/. The Granatum image is provided as an Open Virtual Appliance file, which is loaded by clicking through the *Import Appliance* function of VirtualBox installed on a Windows or Linux system. When the Granatum image is running, it opens an Ubuntu desktop and starts the Granatum server (Additional file 1). When the server has completely loaded a text, a welcome message will appear on the screen indicating that Granatum is ready for use. The server can also be accessed from a web browser outside of the VirtualBox instance, by navigating to the appropriate local port, e.g., http://localhost:8028. Accessing the server from outside of VirtualBox simplifies data transfer to/from the server, e.g., files can be loaded from the user's desktop outside of VirtualBox.

Batch-effect removal

Batch-effect removal is done using the following procedure. First, we calculate the median expression of each sample, denoted as med_i for sample i. Second, we calculate the mean of med_i for each batch, denoted as $batchMean_b$ for batch b,

$$batchMean_b = geometricMean_{i \in batch_b}(med_i).$$

Finally, each batch will be multiplied by a factor which pulls towards the global geometric mean of the sample medians, i.e., when $i \in batch_b$ and m is the number of samples,

$$sampleNew_i = sampleOld_i \cdot \frac{geometricMean_{i \in 1,..,m}(med_i)}{batchMean_b}.$$

Where *sampleNew*$_i$ and *sampleOld*$_i$ denote the expression levels (vector) for all genes within sample i before (old) and after (new) batch-effect removal.

Clustering methods

The following description of clustering algorithms assumes n being the number of genes, m being the number of samples, and k being the number of clusters.

Non-negative matrix factorization (NMF): the log-transformed expression matrix (n-by-m) is factorized into two non-negative matrices H (n-by-k) and W (k-by-m) with k being the expected number of clusters. The latter matrix is then used to determine the membership of each cluster by determining, for each column in W, which of the k entries has the highest value [14,15]. The NMF computation is implemented in the NMF R-package, as reported earlier [14,16].

K-means: K-means is done on either the log-transformed expression matrix or the 2-by-m correlation t-SNE matrix. The algorithm is implemented by the *kmeans* function in R [17].

Hierarchical clustering (Hclust): Hclust is also done on either the log-transformed expression matrix or the 2-by-m correlation t-SNE matrix. The algorithm is implemented by the *hclust* function in R [18]. The heatmap with dendrograms is plotted using the *heatmap* function in R.

Correlation t-SNE

Correlation t-SNE is implemented to assess heterogeneity of the data. It is calculated using a two-step process. First, a distance matrix is calculated using the correlation distance. The correlation distance $D_{i,j}$ between sample i and sample j is defined as

$$D_{i,j} = 1 - \text{Correlation}(S_i, S_j),$$

where S_i and S_j are the i-th and j-th column (sample) of the expression matrix.

Next, t-SNE is performed using this distance matrix, which reduces the expression matrix to two dimensions. We use the Rtsne R package for this calculation [19].

Elbow-point finding algorithm in clustering

In the clustering module with automatic determination of the number of clusters, the identification of the optimum number of clusters is done prior to presenting the clustering results. First, we calculate the k-means clusters from $k = 2$ to $k = 10$. For each k, we calculate the percentage of the explained variance (EV). To find the elbow-point $k = m$ where the EV plateaus, we fit the k-EV data points with a linear elbow function. This function consists of a linearly increasing piece from 0 to m, and a constant piece from m to 10. We iterate from $m = 1$ to 10 and identify m which gives the best coefficient of determination (R^2) of linear regression as the "elbow point".

Differential expression analysis

We use SCDE (version 1.99.4) in our Differential expression (DE) analysis step. The minimum size entries parameter of the *scde.error.models* function is set to be the lesser of 2000 or the number of genes after filtering [20]. When more than two clusters are present, a pair-wise DE analysis is performed.

Gene-set enrichment analysis

The GSEA algorithm is implemented in the *fgsea* R-package which uses an optimized algorithm for fast calculation speed [21].

Pseudo-time construction

We use Monocle (version 2.2.0) in our pseudo-time construction step. When building the *CellDataSet* required for monocle's input, we set the *expressionFamily* to *negbinomial.size()*. The dimension reduction is done using the *reduceDimension* function with *max_components* set to be 2.

Results

Overview

Granatum neatly presents nine modules, arranged as steps and ordered by their dependency (Figure 1), spanning a comprehensive set of methods for single cell analysis. It starts with one or more user-supplied expression matrices and corresponding sample metadata sheet(s), followed by data-merging, batch-effect removal, outlier removal, normalization, gene filtering, clustering, differential expression, protein-protein network, and pseudo-time construction.

Comparing to other freely available tools, the workflow is flexible in several aspects: (1) It supports multiple dataset submission and batch effect removal; (2) at any point of the step, the user can reset the current step for re-analysis; (3) the user can bypass certain steps and still complete the workflow; (4) the user can select subsets of samples/data for their customized analysis need; (5) the user can identify outlier samples either automatically by a pre-set threshold, or manually by

simply clicking the samples the PCA plot or the correlation t-SNE plot; (6) multiple cores can be specified in the differential expression module for speed-up; (7) GSEA can be performed for the differentially expressed genes in all pairs of subgroups, following clustering analysis; (8) Monocle pseudo-time construction can be performed to gain insights of relationships between the cells. We elaborate the details of each step in chronological order, in the following sections.

Upload data

Granatum accepts one or multiple expression matrices as the input. Each expression matrix can be accompanied by a table describing the groups, batches, or other properties of the samples in the corresponding matrix. This accompanying table is called the metadata sheet. Multiple matrices may be uploaded sequentially. The user also specifies the species of the data, either human or mouse, for downstream functional analysis. After the input files are uploaded, preview tables for the matrix and metadata are displayed, providing the user an opportunity check that the data they have input is as expected.

*Batch-effect removal*Samples obtained in batches can create unwanted technical variation, which confound the biological variation [22]. It is thus important to remove the expression level difference due to batches. Granatum provides a batch-effect removal step, where the batches are shown as different colors in the box-plot (Figure 2). If more than one datasets are uploaded, by default each dataset is assumed to be one batch. Alternatively, if the batch numbers are indicated in the sample metadata sheet, the user may select the column in which the batch numbers are stored (blue circled in Figure 2). For datasets with a large number of cells, to maintain legibility of

the box-plot a random selection of 96 sub-samples is shown in the box-plot, and can be re-sampled freely.

Outlier identification

Computationally abnormal samples pose serious problems for many down-stream analysis procedures. It is thus crucial to identify and remove them in the early stage. Granatum's outlier identification step features PCA plot and t-SNE plot, two connected interactive scatter-plots that have different computational characteristics. A PCA plot illustrates the Euclidean distance between the samples, and a correlation t-SNE plot shows the associative distances between the samples. The interactive mode of these plots is realized by the Plotly library [12] (Figure 3A).

Outliers can be identified automatically by either using a z score threshold or setting a fixed number of outliers. In addition, the user can select or de-select each sample, by clicking, boxing or drawing a lasso on its corresponding points on either PCA or t-SNE plot (Figure 3A and 3B). This level of interaction from users is one of the many examples of thoughtful tool design, in order to empower them.

To help users select sample of a particular property, Granatum also allows for mapping any of the columns in the metadata sheet onto the scatter-plots (circled blue in Figure 3A). The complete metadata information of the selected samples can be found in a table at the bottom of the page (circled red in Figure 3A).

Normalization

Normalization is essential to most scRNA-seq data, except those with the UMI counts, before the down-stream functional analyses. The current version of Granatum has implemented three commonly used normalization algorithms: rescale to geometric mean, quantile normalization, and size-factor normalization [23,24]. A box-plot is shown post normalization, to help illustrate its effect to the median, mean, and extreme values across samples. As is the case in the batch-effect removal step, for a dataset with a large number of samples, 96 sub-samples are randomly chosen for the visualization purpose (Figure 3C).

Gene filtering

Due to scRNA-seq's relative high level of noise, it has been recommended to remove lowly expressed genes as well as lowly dispersed genes [4]. To this end, Granatum has a step to remove these genes. The user can interactively select both the average expression level threshold and the dispersion threshold (Figure 3D). The dispersion calculation and negative binomial model fitting are calculated by modifying the output of the Monocle package [3]. We have customized the visualization code to enhance integration with the other components, by setting up the threshold selection sliders and number of genes statistics message on the Granatum web page (Figure 3D). On the mean-dispersion plot, each gene is represented by a point, where the x-axis is the mean of the expression levels after log transformation, and the y-axis is the dispersion factor calculated from a negative binomial model. The preserved genes are highlighted as black and the genes to be removed are labeled as gray colors. The number of genes before and after filtering are also displayed.

Clustering

Clustering is a routine heuristic analysis for scRNA-seq data. Granatum selects five commonly used algorithms: non-negative matrix factorization [14], k-means, k-means combined with correlation t-SNE, hierarchical clustering (hclust), and hclust combined with correlation t-SNE. The number of clusters may be set manually, or automatically determined using an elbow-point finding algorithm (Methods, Figure 4A). For the latter approach, the algorithm will attempt to cluster samples with number of clusters (k) ranging from 2 to 10, and determine the best number by finding the elbow-point k. k indicates the starting point of plateau for explained variance (EV), above which EV creases only minimally. If hclust is selected, a heatmap with hierarchical grouping and dendrograms be shown in a pop-up window (Figure 4B).

Next, the resulting cluster labels obtained above, are then super-imposed onto the two unsupervised PCA and correlation t-SNE plots (Figure 4A). The user can also represent user-defined labels in the sample metadata as different colors in these plots. By comparing the two sets of labels, the users can quickly check the concordance between the prior metadata labels and the computed clusters.

Differential expression

After obtaining a set of clusters, it is intuitively important to identify genes that are differentially expressed between any two clusters. Granatum uses the state-of-the-art SCDE method for its single-cell DE analysis [20]. The DE comparison is performed in a pair-wise fashion when more than two clusters are present. This step is computationally time and memory consuming. To shorten computation time, a user can select the number of cores for parallelization on multi-core machines (Figure 5A). When SCDE is completed, tabbed tables show the genes sorted by their Z-scores,

along with the model coefficients (Figure 5B). As another feature to empower the users, the gene symbols are linked to their corresponding GeneCards pages (www.genecards.org) [25]. The DE results can be downloaded as a CSV file via the "Download CSV table" button.

To investigate the collective biological functions of these genes, the user can further perform Gene Set Enrichment Analysis (GSEA) with either KEGG pathways or Gene Ontology (GO) terms (circled blue in Figure 5B) [26–29]. We have employed a very intuitive bubble-plot to visualize the GSEA results, where the vertical position of the bubble indicates the enrichment score of the gene sets, and the size of the bubble indicates number of genes in that set (KEGG pathway or GO term) (Figure 5C).

Protein network visualization

Protein-protein interaction (PPI) network gives straightforward and systematic understanding of the connections between these differentially expressed genes. Granatum selects the top K (default K=200) genes in the DE results, and super impose the PPI network on them. Genes that are not connected to any other genes in the list are removed from the PPI network. We use visNetwork to enable the interactive display of the graph [10]. The user can freely rearrange the graph by dragging the nodes to the desired location, and reconfiguring the layout to achieve good visibility of the modules (via elastic-spring physics simulation) (Figure 6A). In this interactive graph, the Z-scores are mapped as colors on the nodes where red indicates up-regulation and blue indicates down-regulation.

*Pseudo-time construction*Granatum has included the Monocle algorithm, a widely-used method to reconstruct a pseudo-timeline for the samples [3]. Monocle uses the Reversed Graph Embedding

algorithm to learn the structure of the data, and the Principal Graph algorithm to find the time-lines and branching points of the samples. We superimpose the timeline on the samples scatter-plot projected on the two components of the learned projection matrix. The user may map any pre-defined labels or numeric assays provided in the metadata sheet on to the scatter-plot (Figure 6B). The plotting functions are adapted from the visualization code in Monocle.

Discussion

The field of scRNA-seq is fast-evolving both in terms of the development of instrumentation and the innovation of computational methods. However, it becomes exceedingly hard for a wet-lab researcher without formal bioinformatics training to catch up with the latest iterations of algorithms [5]. This poses major barriers to them and many resort to sending their generated data to third-party bioinformaticians, before they are able to visualize the data themselves. This segregation often prolongs the research cycle time, as it often takes significant effort to maintain effective communications between the two sides (sometimes even more complicated with a third party of the genomics core). Also, issues with the experimentations do not get the chance to be spotted early enough, to avoid significance loss of time and cost in the projects. It is thus very attractive to have a non-programming graphical application which includes state-of-the-art algorithms as routine procedures, in the hands of the bench-scientist who generate the scRNA-seq data.

Granatum is our attempt to fill this void. It is to our knowledge the first solution that aims to cover the entire scRNA-seq workflow with an intuitive, step-wise graphical user interface. Throughout the development process our priority has been to make sure that it is fully accessible to

researchers with no programming experiments. We have strived to achieve that the plots and tables are self-explanatory, interactive and visually pleasant. We have sought inputs from our single-cell bench-side collaborators, to ensure that the terminologies are easy to understand by them. We also supplement Granatum with a manual and video that guide the users through the entire workflow, using example datasets. Currently Granatum targets users who have their expression matrices and metadata sheets ready. However, we are developing the next version of Granatum, which will handle the entire scRNA-seq data processing and analysis pipeline including FASTQ quality control, alignment, and expression quantification. In the future, we will enrich Granatum with capacities to analyze and integrate other types of genomics data in single cells, such as exome-seq and methylation data.

Conclusions

We have developed a graphical web application called Granatum, which enables bench researchers with no programming expertise to analyze state-of-the-art scRNA-Seq data. This tool offers many interactive features to allow routine computational procedures with a great amount of flexibility. We expect that this platform will empower the bench-side researchers with more independence in the fast-evolving single cell genomics field.

Figure legends

Figure 1: Granatum workflow. Granatum is built with the Shiny framework, which supports both front-end and the back-end. The user uploads one or more expression matrices with

corresponding metadata for samples. The back-end stores data separately for each individual user, and invokes third-party libraries on demand.

Figure 2: The batch-effect removal steps. A box-plot is shown for the samples. The colors indicate the batch labels, which can be selected using the batch factor selection box circled in blue. In cases where more than 96 cells are present in the data, only a random sample of 96 cells are shown. The user can re-sample the data by clicking the "Re-plot random 96 cells" button.

Figure 3: The outlier removal, normalization and gene filtering steps. A) The main interface of the outlier removal step. The two scatter-plots are the PCA and correlation t-SNE plots, with colors indicate the cell labels (box circled in blue). The metadata table (circled in red) shows the labels for the selected cells. B) The pop-up window for automatic outlier detection options after the "auto-identify" button is clicked. C) The normalization step. The box-plot shows the expression levels of each cell in log-scale. In cases where more than 96 cells are present in the data, only a random sample of 96 cells are shown. D) The Gene filtering step. The y-axis of the scatter-plot is the empirical dispersion, estimated by a negative binomial model. The x-axis is the log mean expression of each gene. The user can change the threshold by dragging the two sliders circled in blue.

Figure 4: The Clustering step. A) Main interface. PCA and t-SNE plots are shown with colors mapped to user-selected sample labels. After clustering, samples are marked with their assigned cluster numbers. The user can either choose a specific number of clusters or let Granatum

compute the best number of clusters. B) When Hclust (Euclidean) is selected, a pop-up window will show a heatmap of the expression matrix with dendrograms.

Figure 5: The Differential expression (DE) step. A) Before running DE, the user may select the number of cores to use for speed. B) After DE, top differentially expressed genes for each pair of clusters are shown. Gene Set Enrichment Analysis (GSEA) can be performed, using either KEGG pathways or GO terms (circled in blue). C) The results of GSEA. The pathways on the x-axis are sorted top 20 enriched gene sets. The height of the bubble indicates the absolute normalized enrichment score, and the size of the bubble indicates the number of genes in the set.

Figure 6: The Protein network and Pseudo-time construction steps. A) The Protein network step. The A tabbed panel shows the connected gene modules on the PPI network between each pair of clusters. The color on each node (gene) indicates its Z-score in the differential expression test. Red and blue colors indicates up- and down- regulation. B) The Pseudo-time construction step. Monocle algorithm is customized to visualize the paths among individual cells. The user can represent sample labels from the metadata as colors in the plot.

Supplementary files

Additional file 1: Granatum deployment. A screenshot of an activated VirtualBox Appliance running the Granatum server is shown behind a web browser outside of the Appliance, which is accessing the server with the URL http://localhost:8028/. The server can be started by double-clicking the Granatum desktop icon within the Appliance and stopped by closing the Terminal

window, which pops up when the server is activated. All data to/from the server can be handled outside of the Appliance from the external browser.

Availability of data and material

The package Granatum is freely available for research use, and can be downloaded at:

http://garmiregroup.org/granatum/code

A demonstration video can be found at

http://garmiregroup.org/granatum/video

Declarations

NA

Competing interests

The authors declared no conflict of interest.

Funding

This research was supported by grants K01ES025434 awarded by NIEHS through funds provided by the trans-NIH Big Data to Knowledge (BD2K) initiative (http://datascience.nih.gov/bd2k), P20 COBRE GM103457 awarded by NIH/NIGMS, NICHD R01 HD084633 and NLM R01LM012373 and Hawaii Community Foundation Medical Research Grant 14ADVC-64566 to LX Garmire.

Authors' contributions

LXG envisioned the project. XZ developed the majority of the pipeline. TW and AT assisted in developing the pipeline. TW documented the user manual and performed packaging. XZ, TW and LXG wrote the manuscript. All authors have read, revised, and approved the final manuscript.

Acknowledgements

We thank Drs. Michael Ortega and Paula Benny for providing valuable feedback during testing the tool. We also thank other group members in Garmire group for suggestions in the tool development.

List of abbreviations

scRNA-seq: Single-cell high-throughput RNA sequencing

DE: differential expression

GSEA: Gene-set enrichment analysis

KEGG: Kyoto Encyclopedia of Genes and Genomes

GO: Gene ontology

PCA: Principal component analysis

t-SNE: t-Distributed Stochastic Neighbor Embedding

NMF: Non-negative matrix factorization

Hclust: Hierarchical clustering

PPI: Protein-protein interaction

References

1. Patel AP, Tirosh I, Trombetta JJ, Shalek AK, Gillespie SM, Wakimoto H, et al. Single-cell RNA-seq highlights intratumoral heterogeneity in primary glioblastoma. Science (80-.). [Internet]. Department of Neurosurgery, Massachusetts General Hospital and Harvard Medical School, Boston, MA 02114, USA. Department of Pathology and Center for Cancer Research, Massachusetts General Hospital and Harvard Medical School, Boston, MA 02114, USA. Broad I; 2014;344:1396–401. Available from: http://dx.doi.org/10.1126/science.1254257

2. Lewis BP, Burge CB, Bartel DP. Conserved seed pairing, often flanked by adenosines, indicates that thousands of human genes are microRNA targets. Cell. Elsevier; 2005;120:15–20.

3. Trapnell C, Cacchiarelli D, Grimsby J, Pokharel P, Li S, Morse M, et al. The dynamics and regulators of cell fate decisions are revealed by pseudotemporal ordering of single cells. Nat. Biotechnol. Nature Research; 2014;32:381–6.

4. Brennecke P, Anders S, Kim JK, Kołodziejczyk AA, Zhang X, Proserpio V, et al. Accounting for technical noise in single-cell RNA-seq experiments. Nat. Methods. Nature Publishing Group; 2013;

5. Poirion OB, Zhu X, Ching T, Garmire L. Single-Cell Transcriptomics Bioinformatics and Computational Challenges [Internet]. Front. Genet. . 2016. p. 163. Available from: http://journal.frontiersin.org/article/10.3389/fgene.2016.00163

6. Team RC. R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. 2015, URL h ttp. www. R-project. org. 2016;

7. McCarthy DJ, Campbell KR, Lun ATL, Wills QF. scater: pre-processing, quality

control, normalisation and visualisation of single-cell RNA-seq data in R. bioRxiv [Internet]. Cold Spring Harbor Labs Journals; 2016; Available from: http://biorxiv.org/content/early/2016/08/15/069633

8. RStudio, Inc. Easy web applications in R. 2013.

9. Attali D. shinyjs: Easily Improve the User Experience of Your Shiny Apps in Seconds [Internet]. 2016. Available from: https://cran.r-project.org/package=shinyjs

10. Almende B.V., Thieurmel B. visNetwork: Network Visualization using “vis.js” Library [Internet]. 2016. Available from: https://cran.r-project.org/package=visNetwork

11. Xie Y. DT: A Wrapper of the JavaScript Library “DataTables” [Internet]. 2016. Available from: https://cran.r-project.org/package=DT

12. Sievert C, Parmer C, Hocking T, Chamberlain S, Ram K, Corvellec M, et al. plotly: Create Interactive Web Graphics via “plotly.js” [Internet]. 2016. Available from: https://cran.r-project.org/package=plotly

13. Wickham H. ggplot2: Elegant Graphics for Data Analysis [Internet]. Springer-Verlag New York; 2009. Available from: http://ggplot2.org

14. Zhu X, Ching T, Pan X, Weissman S, Garmire L. Detecting heterogeneity in single-cell RNA-Seq data by non-negative matrix factorization. PeerJ Prepr. PeerJ Inc. San Francisco, USA; 2016;4:e1839v1.

15. Brunet J-P, Tamayo P, Golub TR, Mesirov JP. Metagenes and molecular pattern discovery using matrix factorization. Proc. Natl. Acad. Sci. [Internet]. 2004;101:4164–9. Available from: http://www.pnas.org/content/101/12/4164.abstract

16. Gaujoux R, Seoighe C. Algorithms and framework for nonnegative matrix

412 factorization (NMF). 2010.

413 17. Lloyd S. Least squares quantization in PCM. IEEE Trans. Inf. theory. IEEE;

414 1982;28:129–37.

415 18. Murtagh F, Contreras P. Methods of hierarchical clustering. arXiv Prepr.

416 arXiv1105.0121. 2011;

417 19. Krijthe J. Rtsne: T-Distributed Stochastic Neighbor Embedding using Barnes-Hut

418 Implementation. R Packag. version 0.10, URL http//CRAN. R-project. org/package=

419 Rtsne. 2015;

420 20. Kharchenko P V, Silberstein L, Scadden DT. Bayesian approach to single-cell

421 differential expression analysis. Nat. Methods. Nature Publishing Group;

422 2014;11:740–2.

423 21. Sergushichev A. An algorithm for fast preranked gene set enrichment analysis

424 using cumulative statistic calculation. bioRxiv [Internet]. Cold Spring Harbor Labs

425 Journals; 2016; Available from: http://biorxiv.org/content/early/2016/06/20/060012

426 22. Hicks SC, Teng M, Irizarry RA. On the widespread and critical impact of systematic

427 bias and batch effects in single-cell RNA-Seq data. bioRxiv. Cold Spring Harbor Labs

428 Journals; 2015;25528.

429 23. Bolstad BM, Irizarry RA, Åstrand M, Speed TP. A comparison of normalization

430 methods for high density oligonucleotide array data based on variance and bias.

431 Bioinformatics. Oxford Univ Press; 2003;19:185–93.

432 24. Love MI, Huber W, Anders S. Moderated estimation of fold change and dispersion

433 for RNA-Seq data with DESeq2. bioRxiv. Cold Spring Harbor Labs Journals; 2014;

434 25. Rebhan M, Chalifa-Caspi V, Prilusky J, Lancet D. GeneCards: integrating

information about genes, proteins and diseases. Trends Genet. Elsevier Current Trends; 1997;13:163.

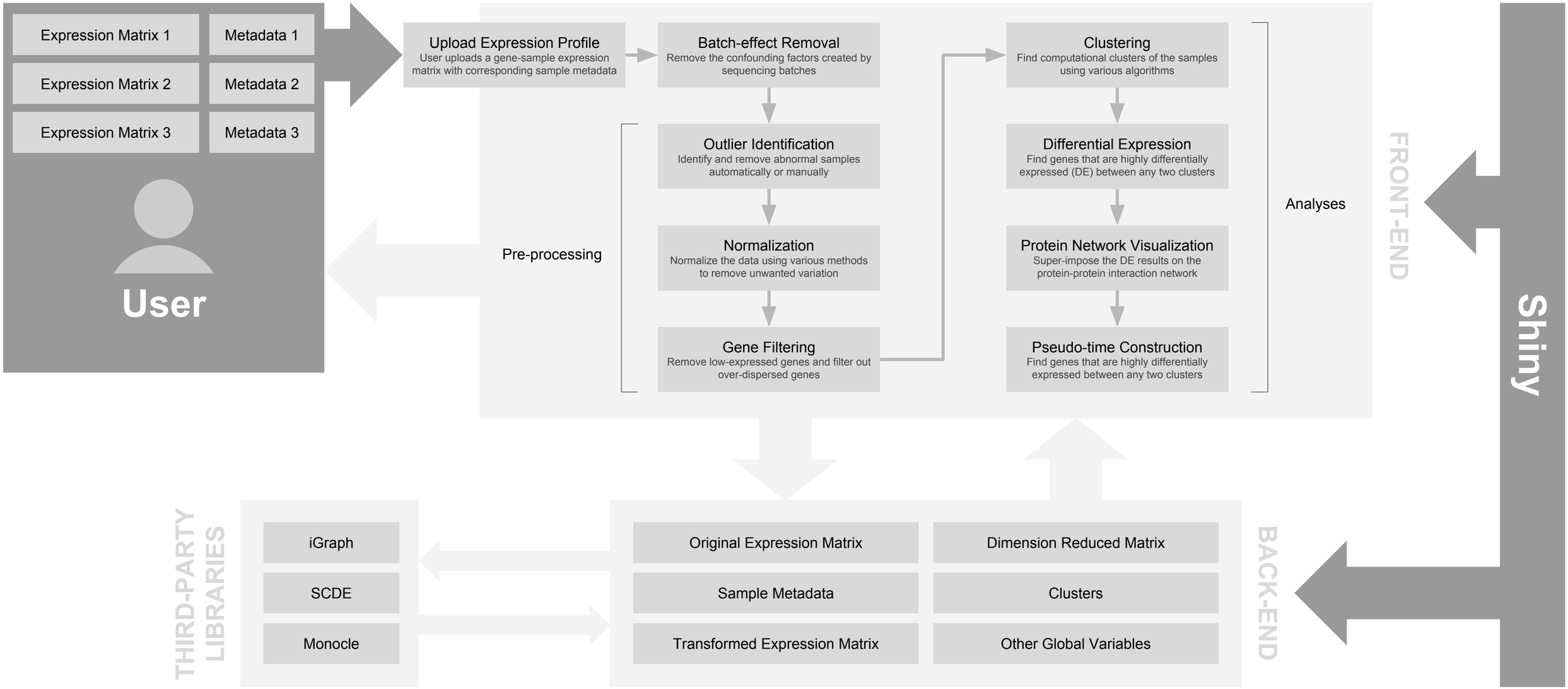
26. Subramanian A, Tamayo P, Mootha VK, Mukherjee S, Ebert BL, Gillette MA, et al. Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. Proc. Natl. Acad. Sci. National Acad Sciences; 2005;102:15545–50.

27. Kanehisa M, Furumichi M, Tanabe M, Sato Y, Morishima K. KEGG: new perspectives on genomes, pathways, diseases and drugs. Nucleic Acids Res. Oxford Univ Press; 2017;45:D353--D361.

28. Consortium GO, others. Gene ontology consortium: going forward. Nucleic Acids Res. Oxford Univ Press; 2015;43:D1049--D1056.

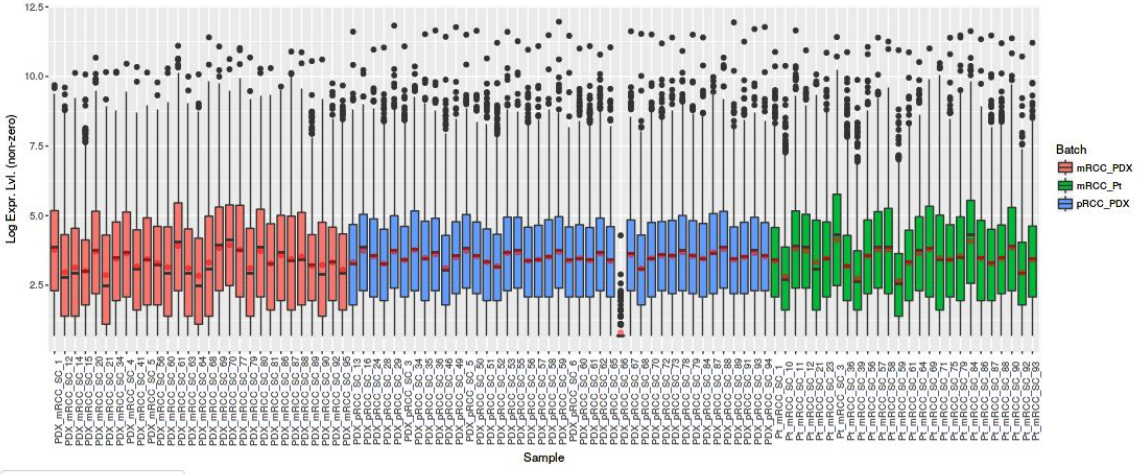
29. Ashburner M, Ball CA, Blake JA, Botstein D, Butler H, Cherry JM, et al. Gene Ontology: tool for the unification of biology. Nat. Genet. Nature Publishing Group; 2000;25:25–9.

Granatum



Batch-effect removal

Data generated in batches may have confounding effects on results. To address this, select the factor that distinguishes cells in different batches, e.g., "Dataset", and check the underlying box before clicking a normalization button.



Re-plot random 96 cells

Batch factor:

Type

Remove batch effect

Reset Submit

A

B

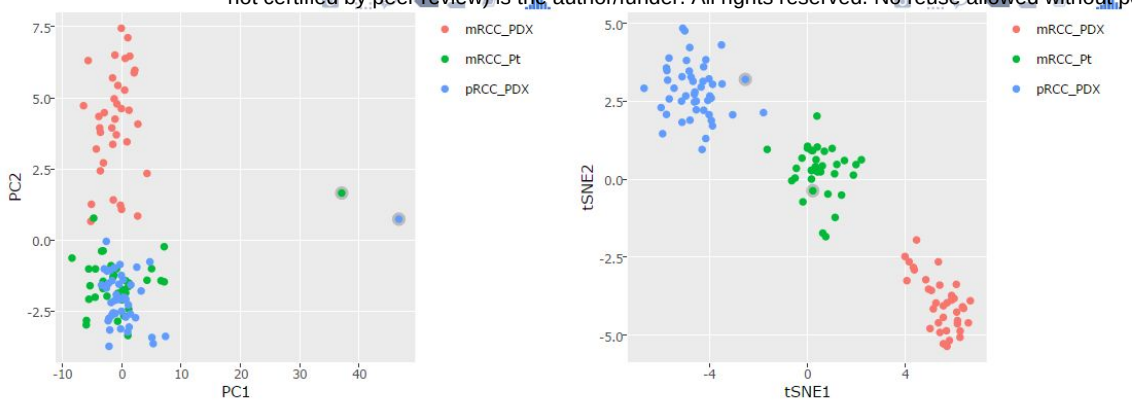
C

D

Outlier removal

Cell labels (from metadata)

Type



Cluster using only top expressed genes

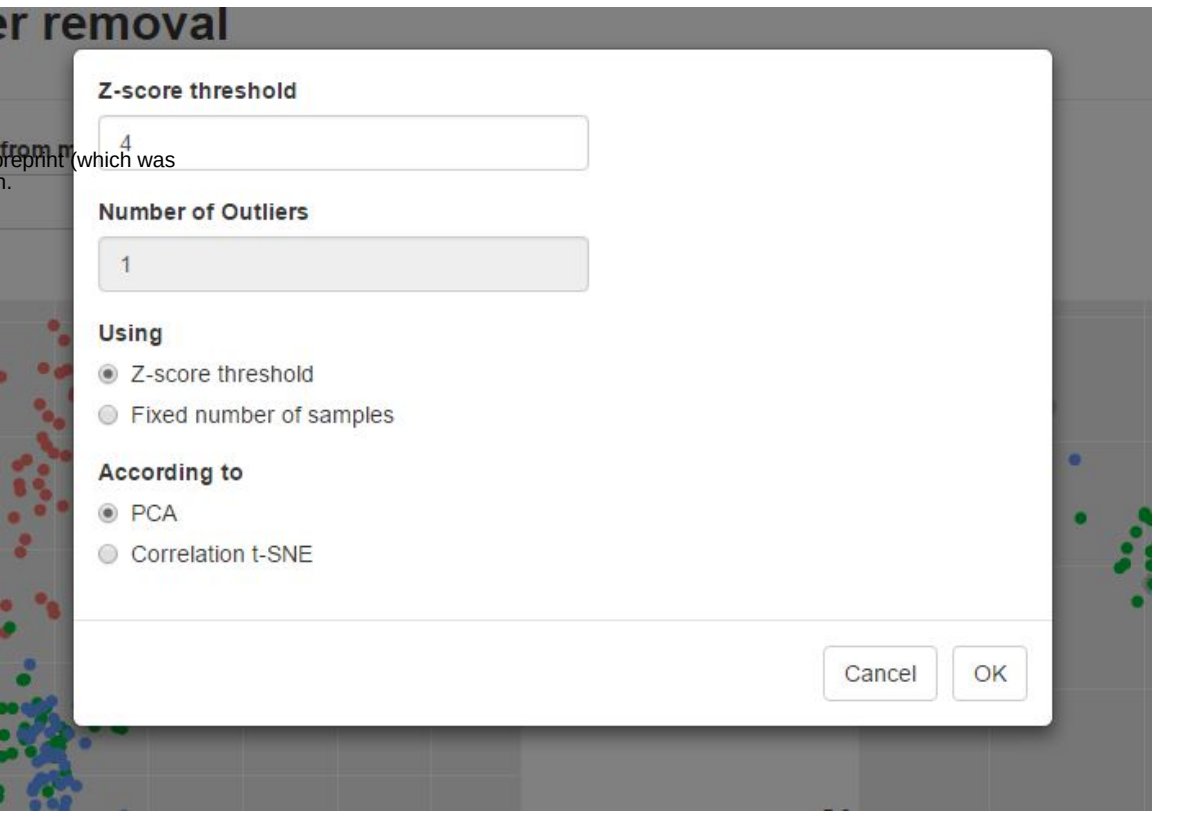
Auto-identify | Remove selected | De-select all

Reset | Submit

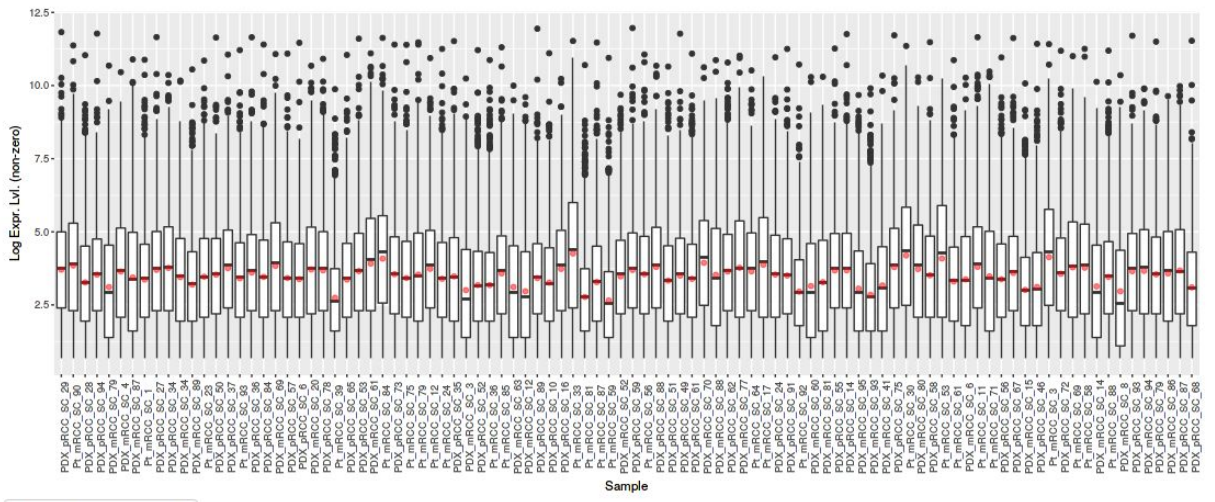
Selected cells:

id	Type	State	Pt_PDX	Mapped_reads	GSM	SRX	SRR
Pt_mRCC_SC_5	mRCC_Pt	mRCC	Pt	36234	GSM1887310	SRX1253756	SRR2431431
PDX_pRCC_SC_66	pRCC_PDX	pRCC	PDX	7603	GSM1887283	SRX1253736	SRR2431411

Showing 1 to 2 of 2 entries



Normalization

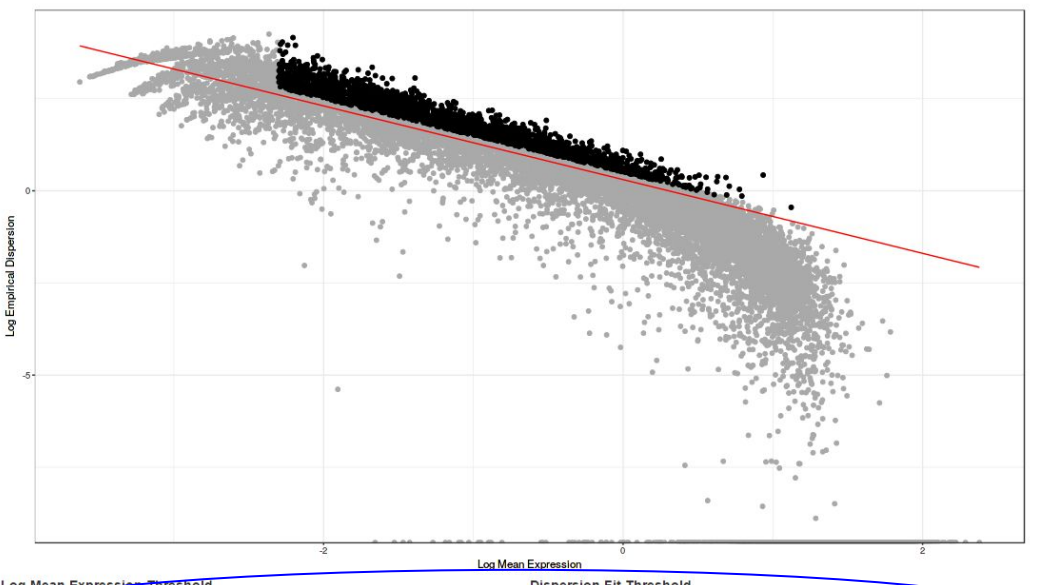


Re-plot random 96 cells

Rescale to genometric mean | Quantile normalization | Size-factor normalization

Reset | Submit

Gene filtering



Log Mean Expression Threshold

Dispersion Fit Threshold

Starting number of genes: 19924

Post-filtering number of genes: 2252

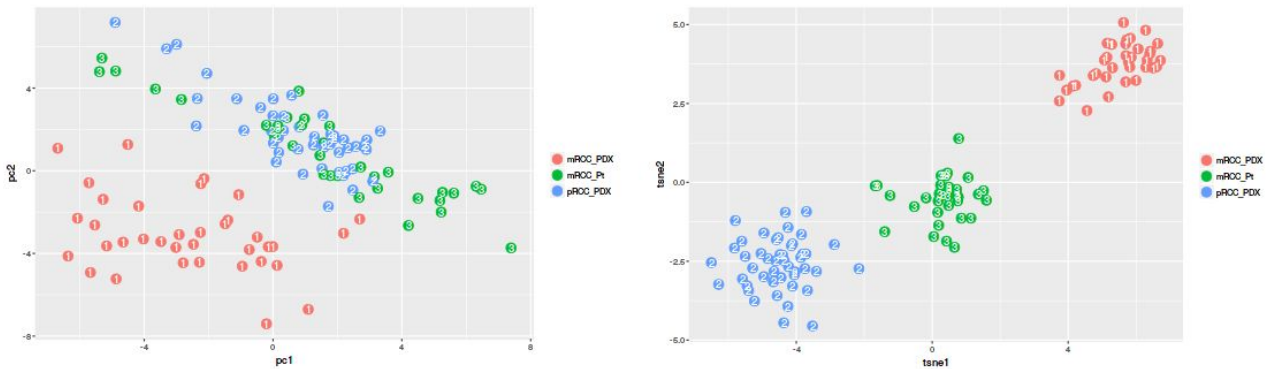
Submit

A

Clustering

Cell labels

Type



Clustering method

- Non-negative matrix factorization
- K-means (Euclidean)
- K-means (correlation t-SNE)
- Hierarchical clustering (Euclidean) with heatmap
- Hierarchical clustering (correlation t-SNE)

Automatically choose the number of clusters (might take long time)

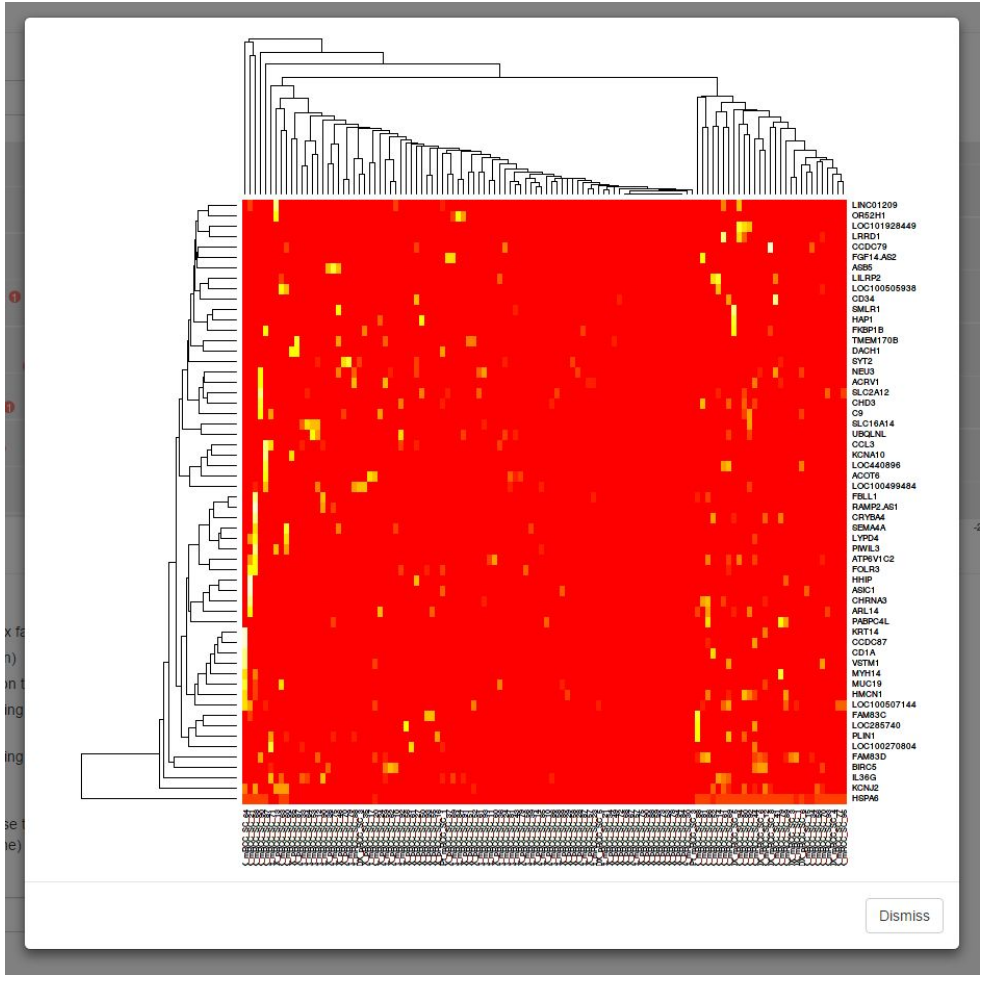
Number of clusters

3

Run clustering

Submit

B



A

Differential expression

Number of processor cores

1

Start analysis

B

Differential expression

Cell labels

Type

Numbers in tabs below indicate which clusters have been compared. Genes are sorted most to least differentially expressed by absolute Z-score value.

1 vs. 2 | 1 vs. 3 | 2 vs. 3

Show 10 entries Search:

gene	lb	mle	ub	ce	Z	cZ
CDH6	-14.916794	-5.746980	-4.563778	-4.563778	-7.160847	-6.337979
HSPA6	5.070865	6.338581	14.536479	5.070865	7.160813	6.337979
KRT81	12.001047	12.930705	13.437792	12.001047	7.160813	6.337979
CSF2	11.493960	12.465876	13.141991	11.493960	7.160809	6.337979
TCN1	-13.353277	-12.803934	-8.028869	-8.028869	-7.157471	-6.337979
DKK1	10.986874	12.043304	12.677162	10.986874	7.155977	6.337979
SLC15A1	-13.564563	-13.057477	-6.296324	-6.296324	-7.155594	-6.337979
SAMD5	-12.592648	-12.043304	-11.324932	-11.324932	-7.146775	-6.337979
MEG3	-12.592648	-11.874275	-11.155903	-11.155903	-7.140434	-6.337979
DCAF4L1	6.761153	12.423619	13.015220	6.761153	6.836788	6.028128
gene	lb	mle	ub	ce	Z	cZ

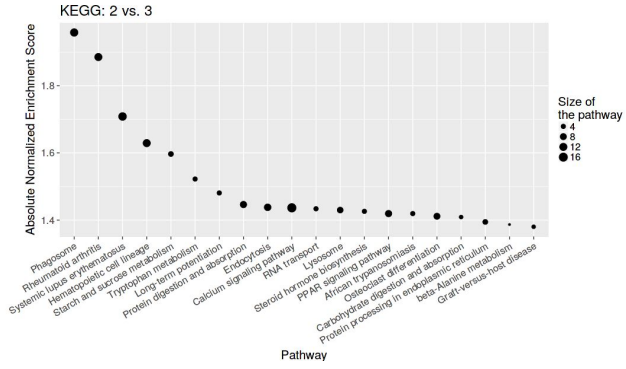
Showing 1 to 10 of 2,252 entries

Previous 1 2 3 4 5 … 226 Next

KEGG enrichment | Gene Ontology enrichment

Download CSV table | Submit

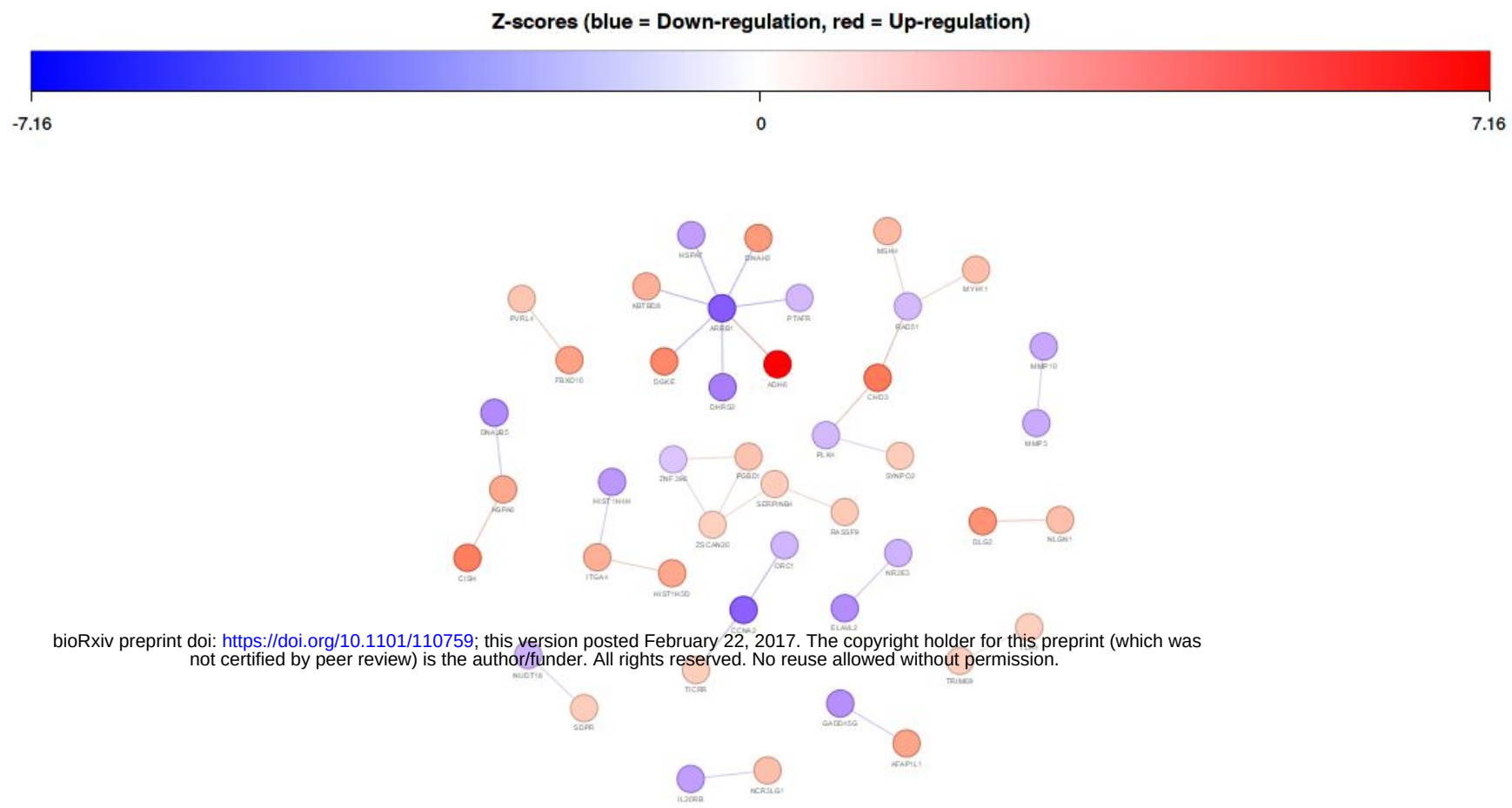
C



A

Protein network

1 vs. 2 | 1 vs. 3 | 2 vs. 3



Proceed

B

Pseudo-time construction

Cell labels

Type