scAnalyzeR: a comprehensive software package with graphical user interface for single-cell RNA sequencing analysis

User Manual

scAnalyzeR

Version 1.0

04 June 2021

Contents **Page**

1. Introduction

scAnalyzeR is a comprehensive platform for analysing and visualizing single-cell RNA sequencing (scRNA-seq) data with an interactive graphical user interface. The scRNA-seq technology is becoming popular for the investigation of heterogeneous cell populations at single cell level, such as identifying cell diversity [1] and revealing cell subtypes [2]. Besides, this technology is applied for discovering disease surface markers [3] and constructing the path in progression over time for individual cells [4, 5]. Here is a brief description of the scAnalyzeR tool how it can be run successfully. This user manual covers procedures for uploading, pre-processing (discarding low-quality samples, outlier detection and removal, etc.), gene-expression normalization, highly variable genes(features) identification, cell clustering, differential gene expression analysis (including cluster-oriented marker genes finding), five basic plotting functions (violin plot, feature plot, heatmap, dynamic pca, and correlation plot), pathway enrichment analysis, and pseudo-time construction (trajectories development).

2. How to setup

There are two different ways to setting up the pipeline in your own machine:

Way 1: using from Docker image (**strongly recommended**)

1. Download and install Docker (https://www.docker.com/products/docker-desktop)
2. Pull the docker image from Docker Hub by running the following command:
   `docker pull gscdocker/scanalyzer:latest`
3. Run the docker image locally on your computer and access the link:
   To run the docker image, execute the following command:
   `docker run -d --rm -p 3838:3838 gscdocker/scanalyzer:latest`
   After running the docker image successfully, open the following link on a web browser (e.g., Firefox, Google Chrome) to access the pipeline:
   `http://localhost:3838/`

Way 2: using from source

Firstly, you need to download and install following softwares (install R then RStudio):

Download and install R and RStudio on your machine,

i. Download and install R (v-3.6.2 or above): https://cran.r-project.org/

ii. Download and install RStudio: RStudio (v-1.1.456 or above): https://rstudio.com/products/rstudio/download/

After installing the R and RStudio on your machine successfully, then, you need to clone this (https://github.com/sarwarchy20/scAnalyzeR/archive/master.zip) repository as well as unzip it.

Now, please run the following script on RStudio to install the ***renv*** R package:

```
install.packages("renv")
```

Next, run the following scripts on RStudio to install all the dependent R packages. Please replace ~ with the location of your scAnalyzeR-master unzipped folder:

```
renv::consent(provided=TRUE)
setwd("~/scAnalyzeR-master")
renv::restore()
```

Finally, run the app using RStudio by running the script below:

```
shiny::runApp('~/scAnalyzeR-master/')
```

After successfully running the scAnalyzeR, the GUI will be displayed automatically. You can customize (minimize or maximize) window size via mouse clicking. If you would like to open the scAnalyzeR in a browser, click the “Open in Browser” or copy the link (top left corner) and paste it in a browser taskbar (**Fig. 1**).

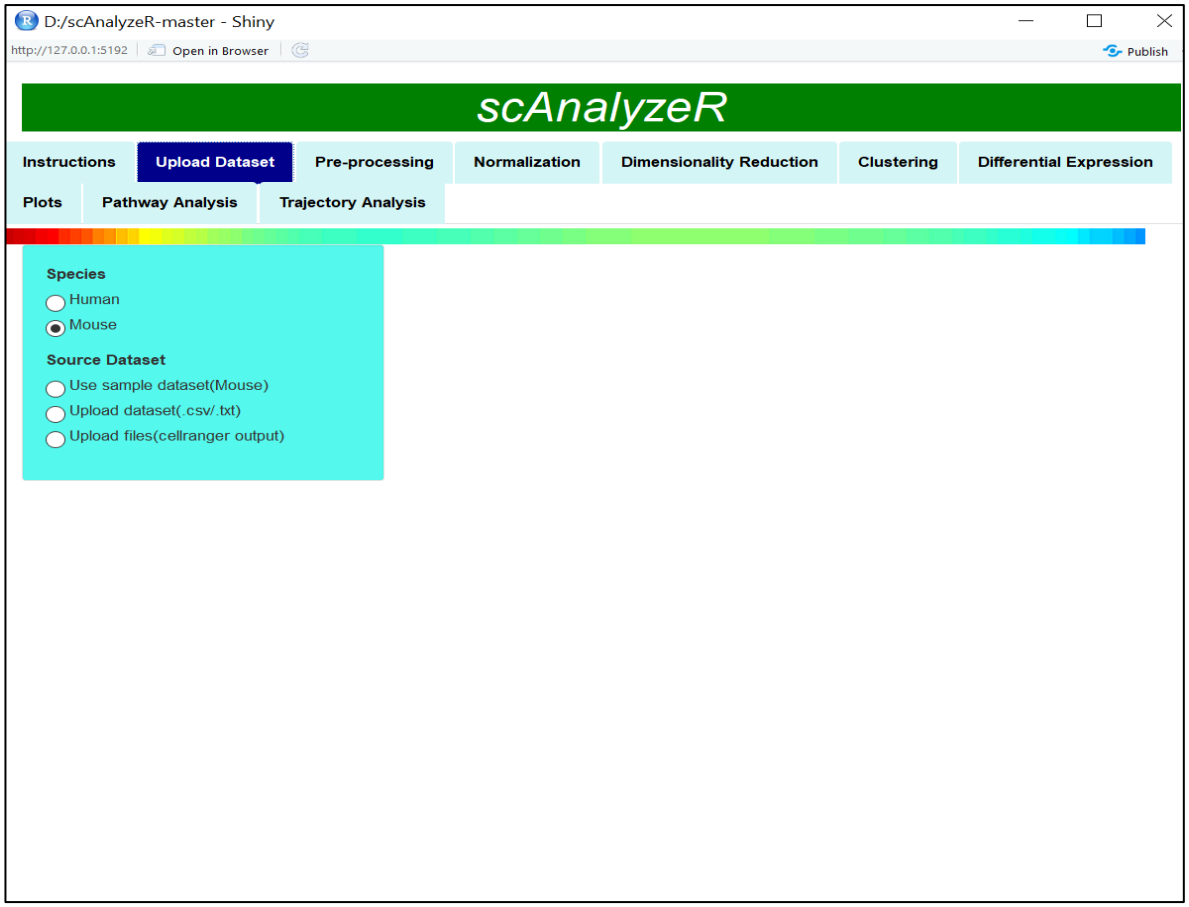


Figure 1. *Interface of the scAnalyzeR*

3. Upload dataset

This module is the first step for single cell RNA sequencing (scRNA-seq) data analysis using our developed tool, scAnalyzeR. Before uploading the data, first, choose the right species (Human or Mouse) option from where the data was sequenced. After selecting the appropriate species option, then select the perfect data source option which has the best match with your data file format. A sample dataset (mouse ovarian surface epithelium, 14,169 genes across 329 control cells) [6] is integrated with the tool, only for demonstration purpose. If you wish to use it, select the "Upload sample dataset (Mouse)," then the data file is loading automatically, and a progress bar will be shown on the bottom right corner of the interface. A text notification also will be displayed after loading the data file successfully (**Fig. 3**).

If you select the "Upload dataset(.csv/.txt)" option, it will show the data uploading facilities (interface) with some parameters, i.e., Browse, Header, Separator, and Quote. The comma (as a separator), double quote (as a quote), and header are selected by default. You can set different option(s) that will satisfy with your dataset format, which is going to be uploaded. For this data source option, the dataset must be a single file, and the first row provides the identities of the

cells (e.g., cell barcodes), and the first column contains the gene symbols. By clicking the "Browse" button, select the desired file (from your machine directory where the data file is located). After clicking the "open" menu, the file is uploading along with a progress bar, and it will show a text notification (how many genes and cells does exist in the uploaded data file) on the right side of the interface after completion the uploading.

For cellranger's output data uploading (**Fig. 2**), first, choose the "Upload files(cellragner output)," then click the "Browse" button, and select the three files- matrix.mtx, genes.tsv and barcodes.tsv, respectively. In our example, uploaded a 3k data set (2,700 single cells with 32,738 genes), a cellranger output of Peripheral Blood Mononuclear Cells (PBMC) from 10X Genomics (https://support.10xgenomics.com/single-cell-gene-expression/datasets/1.1.0/pbmc3k).

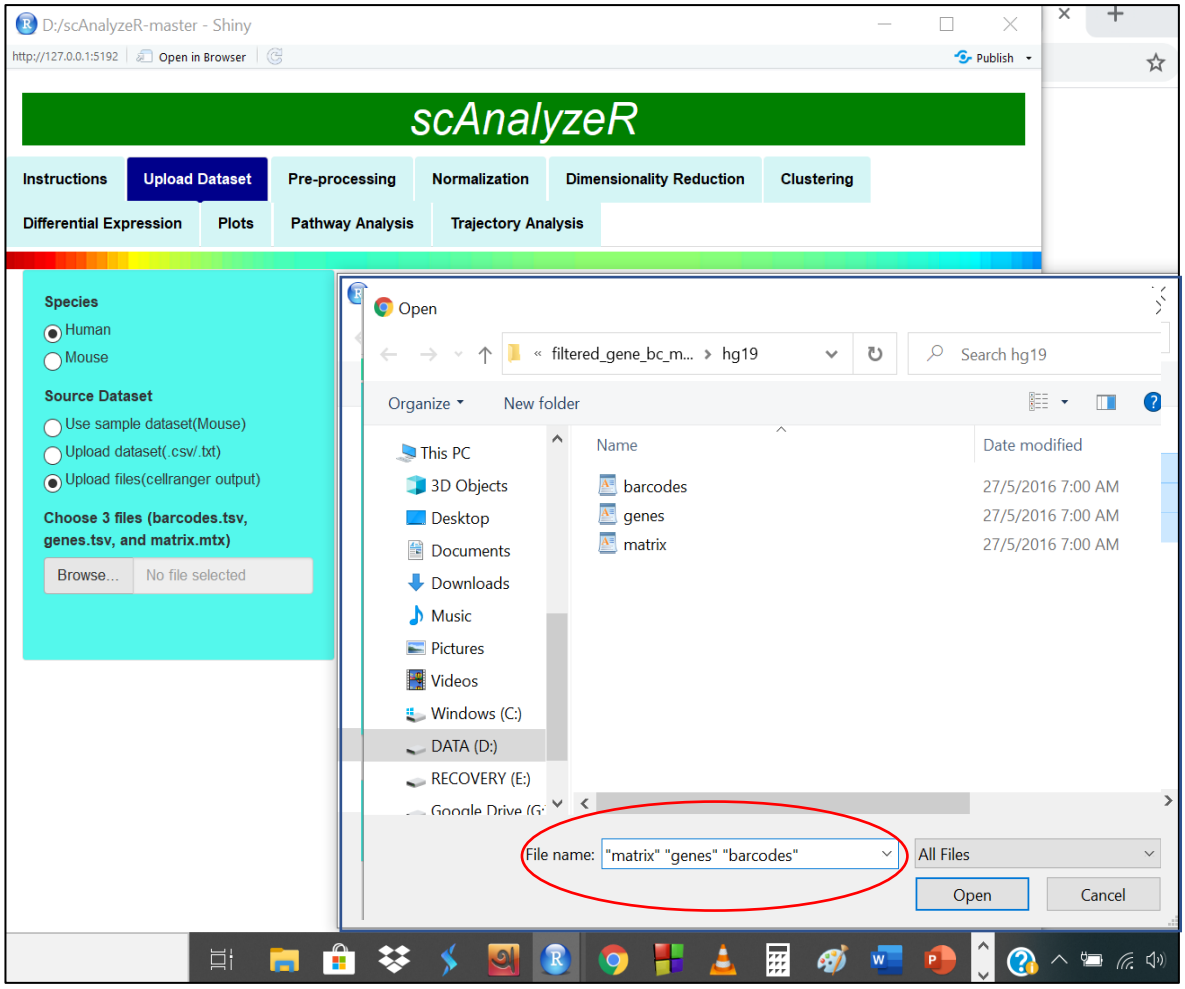


Figure 2. Upload cellranger's files. Follow the file name order (matrix, genes, barcodes) when you are using the Windows OS.

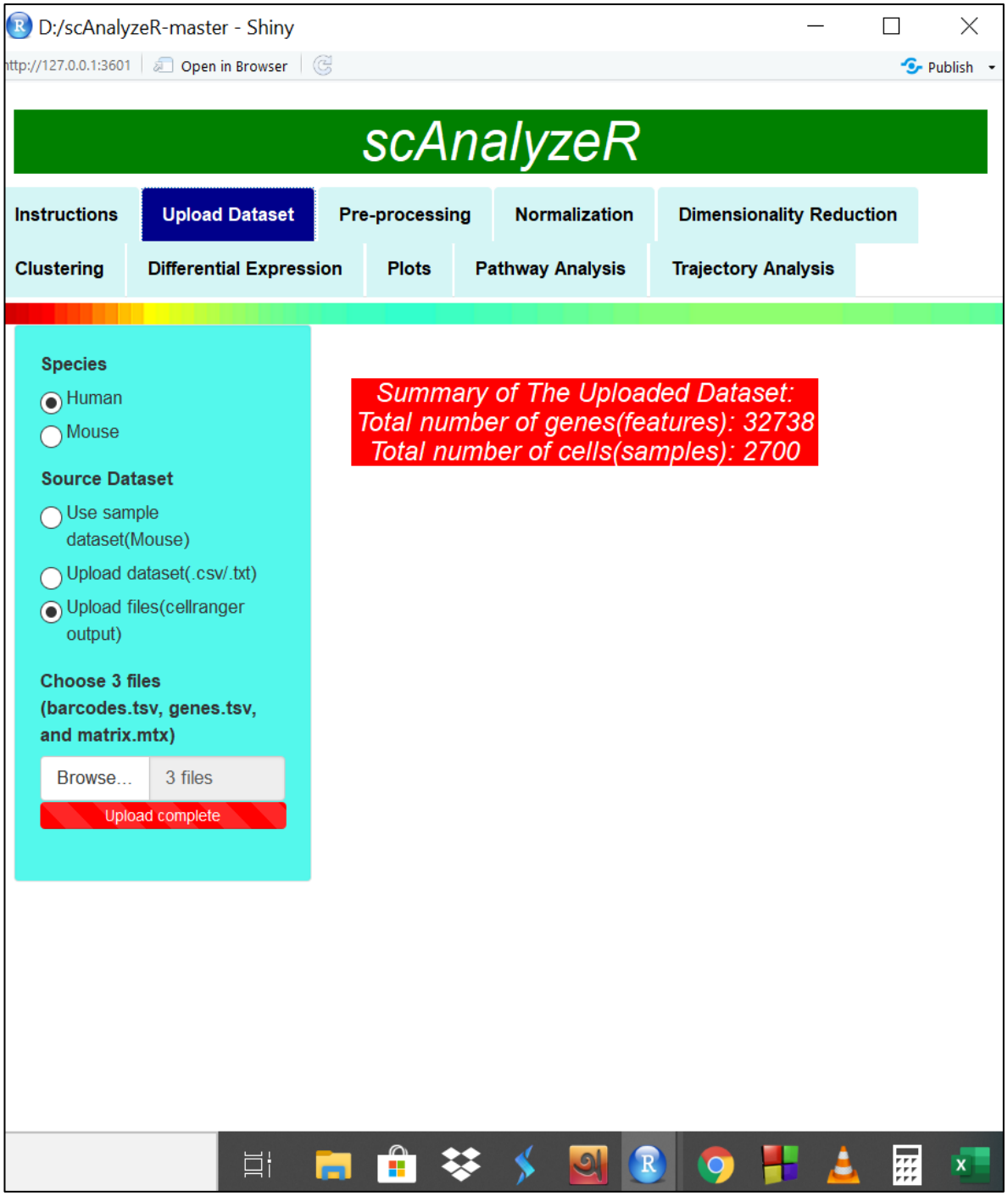


Figure 3. A text notification after successfully uploading the data set

4. Pre-processing

In this step, low-quality samples can be identified and discarded by setting some parameters. There are two-step filtering procedures – create dataset (**Fig. 4-6**), and filtering (**Fig. 7-8**). The dataset can be created by clicking the "Create dataset" button. The created dataset will keep all genes that are expressed in a minimum number of cells (3, by default) and contain those cells

with at least minimum genes (200, by default) detected. The default value(s) can be changed as required. Now, it is ready for showing the list of mitochondrial genes, plot metadata, gene summary plot, and download mitochondrial gene expression. To show a plot or list, select a particular checkbox via mouse clicking. Moreover, it can be copied and saved via right-click on the mouse. Deselect the checkbox to clear plot (s) or list.

Sometimes very few genes present in low-quality cells or empty droplets, and an abnormally high gene count may lie in cell doublets or multiplets. Even in, immense mitochondrial contamination points out in dying cells or low-quality cells. Now our target is discarding cells that have a paranormally high and low gene counts and a significant level of mitochondrial contamination. In the filtering step, check the "Filtering Cells" checkbox, then the filtering interface will be shown with parameters list along default values setting. The default values can be edited by mouse clicking or typing. After clicking the "Submit" button, a text notification will be shown, which indicates how many cells remain in the dataset after filtering. In this instance, discarded cells that have unique gene counts over 2500 or less than 200 and greater than 5% mitochondrial counts. If the notification shows a message "*Error: Cannot find cells provided*," it means that the dataset is empty (no cell exists) after filtered with these parameters setting. In this situation, it is necessary to change the parameter(s) value and click the "Submit" button again.

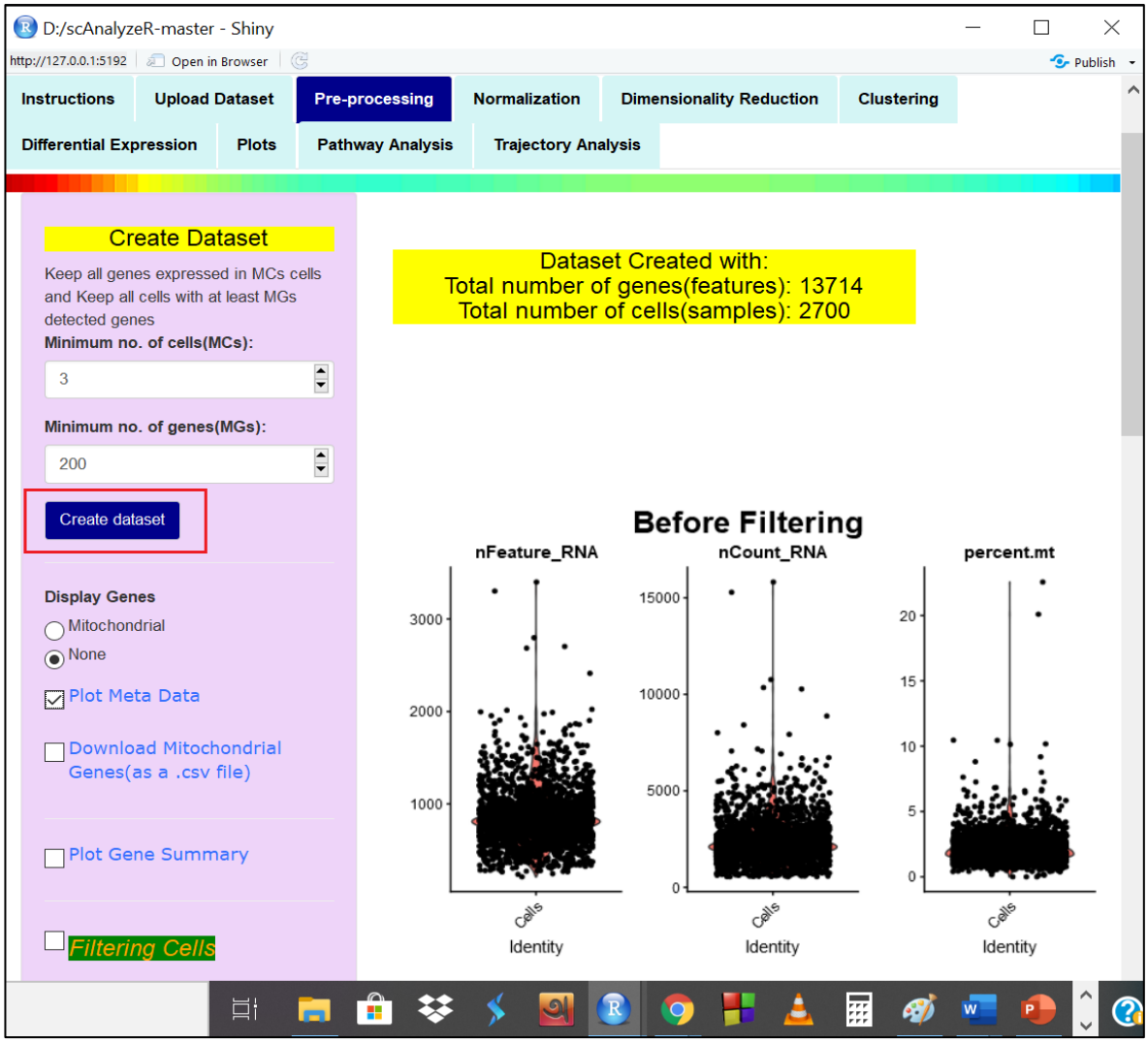


Figure 4. Create data: Meta data plot after creating the data set

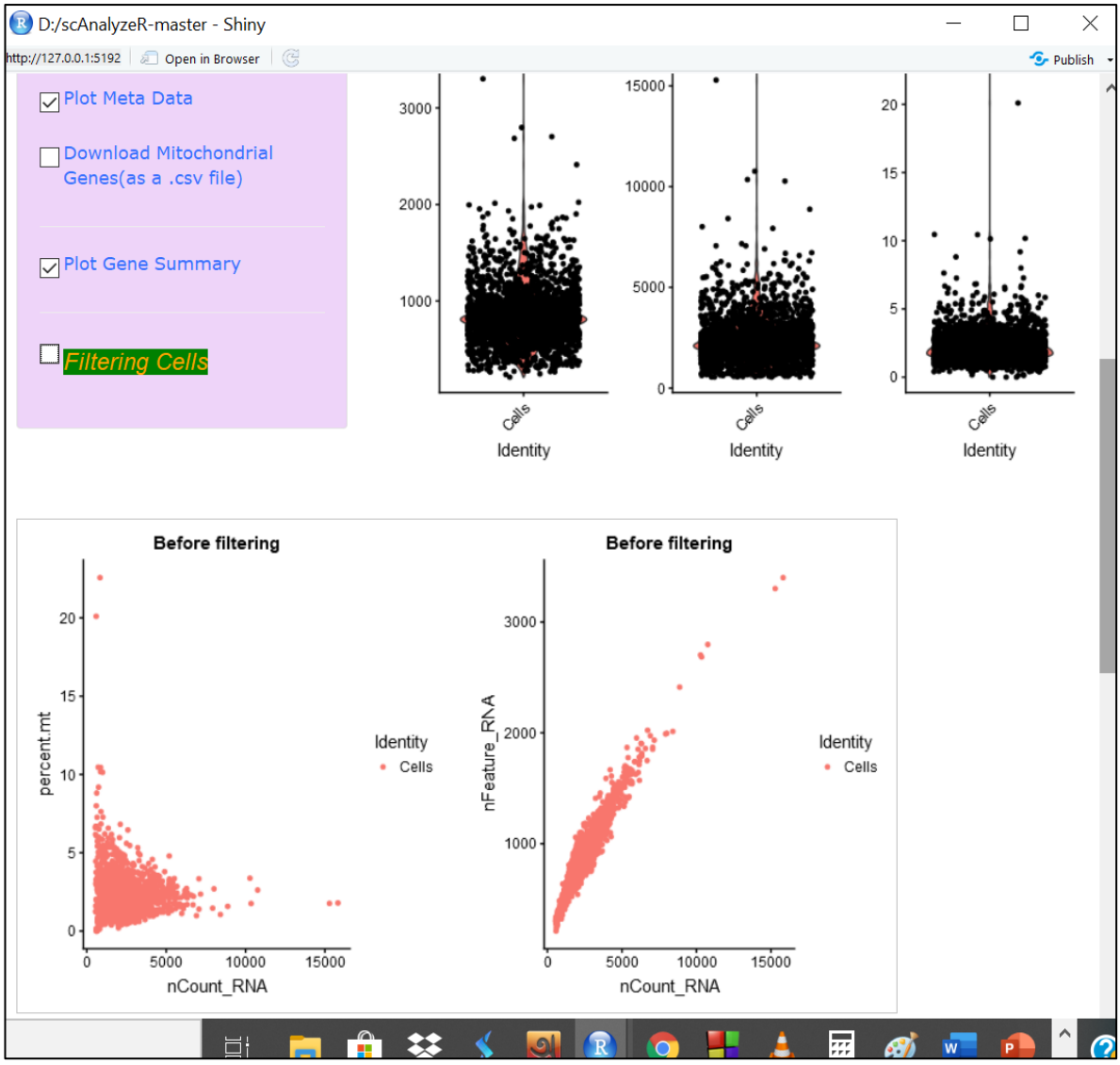


Figure 5. Create data: Gene summary plot (bottom) after creating the data set

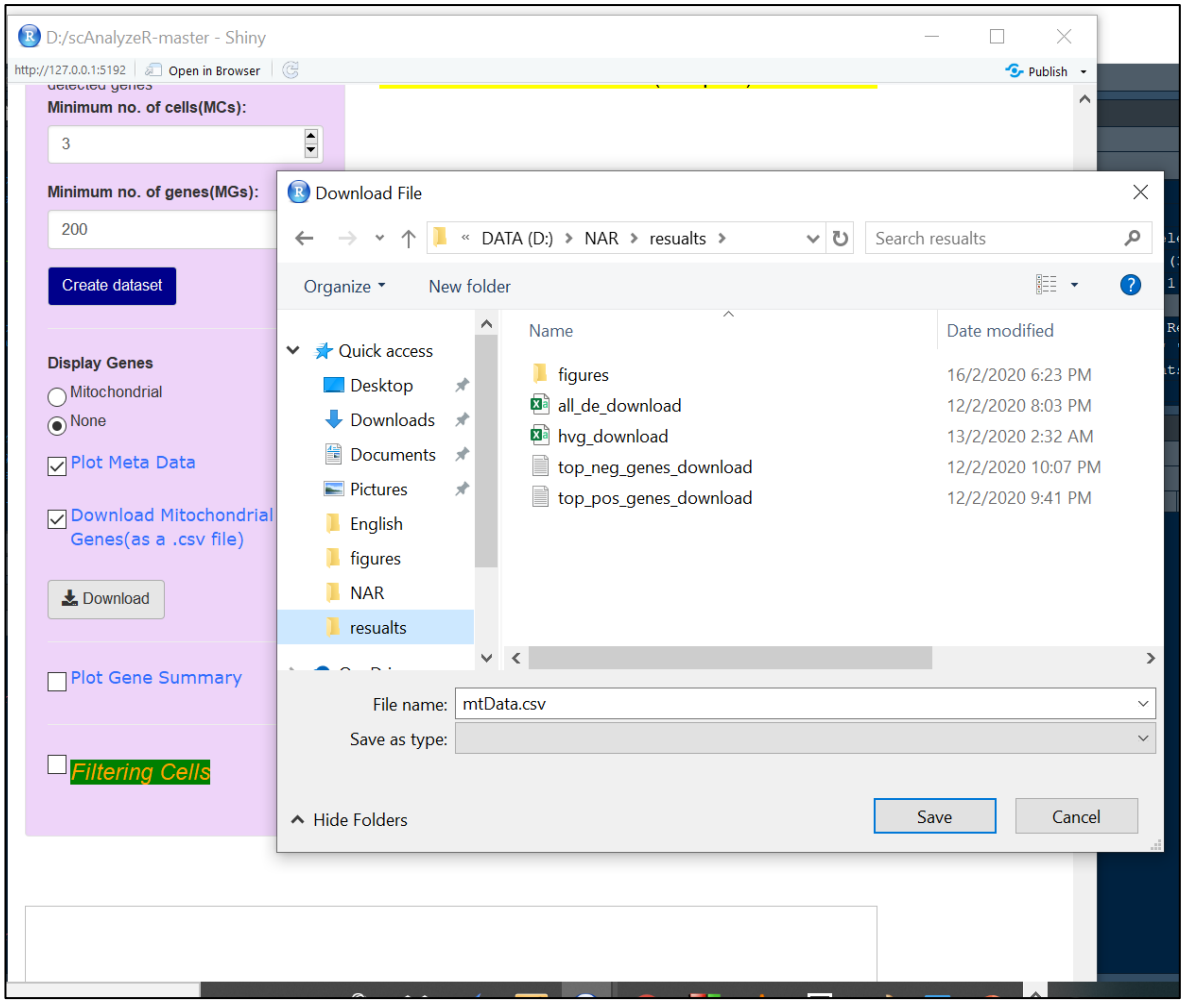


Figure 6. Create data: Download mitochondrial gene expressions after creating the data set

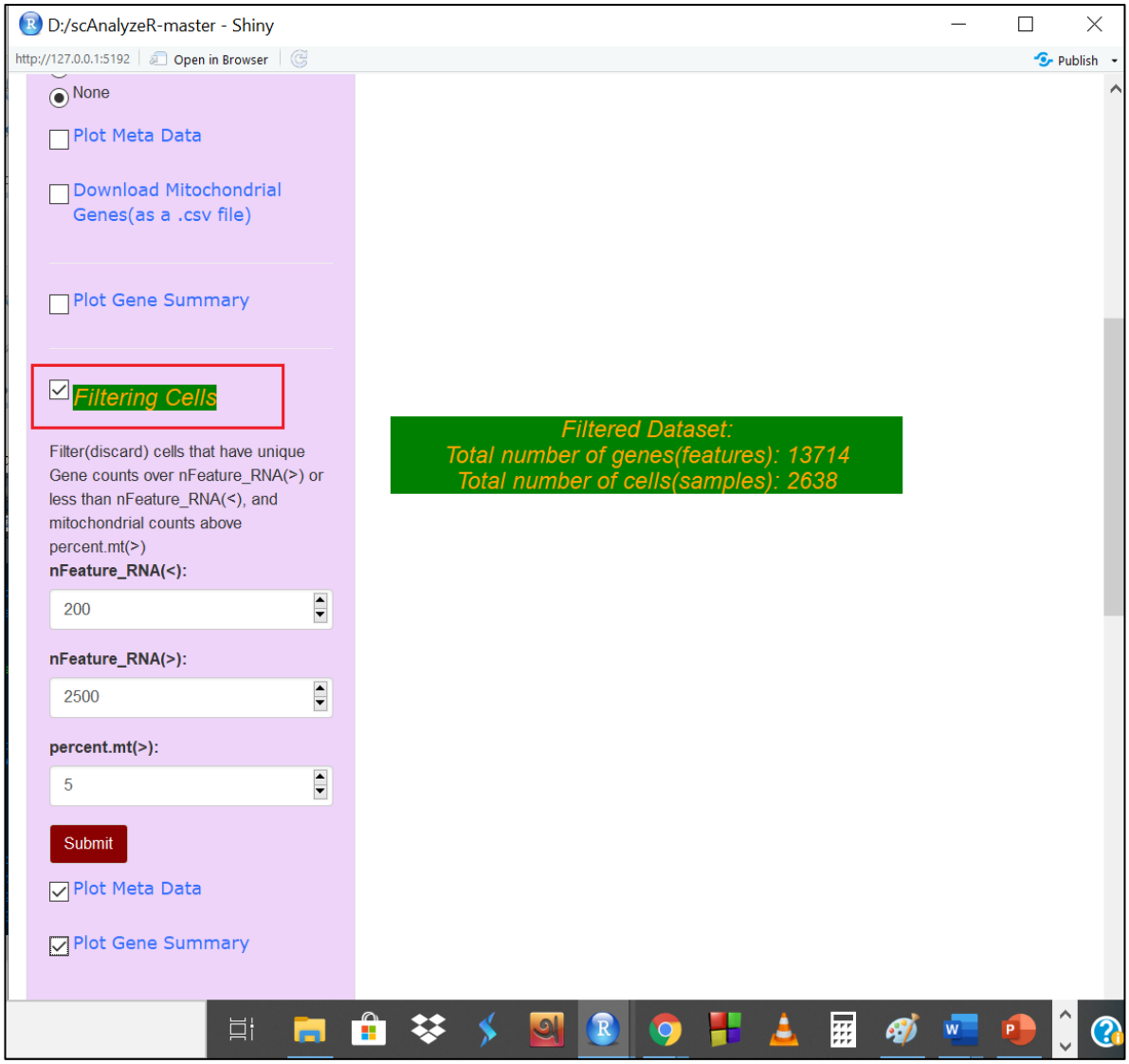


Figure 7. Filtering

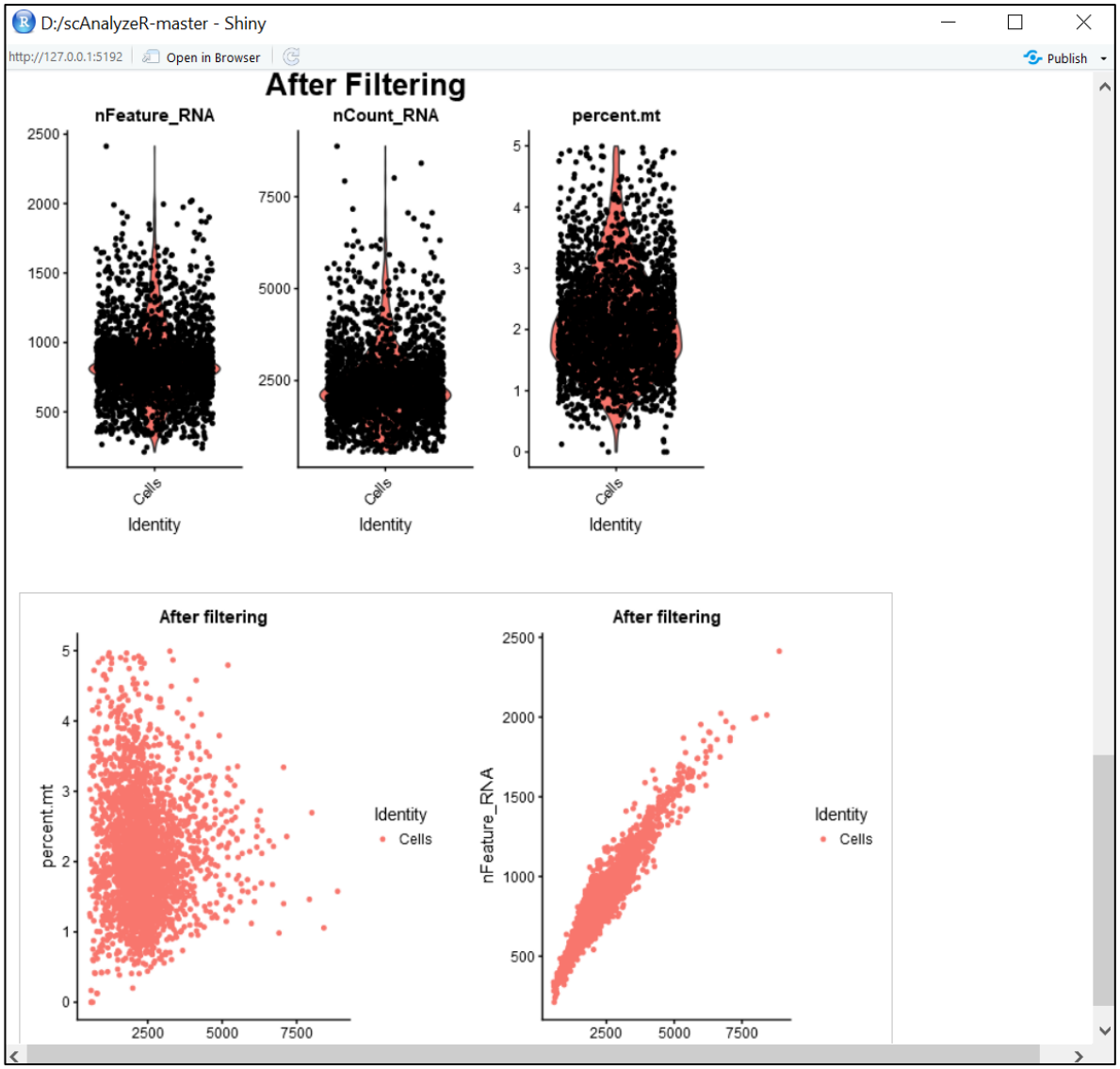


Figure 8. Filtering: Meta data plot(top), Gene summary plot(bottom) after filtering

5. Normalization

After discarding unwanted cells, in this module, the normalization method will be applied, and highly variable Genes will be identified from the normalized data. To normalize, choose a normalization method from the "Normalization Method" drop-down list (**Fig. 9**). The 'Log Normalization' method selected by default. The user can also change the scaling factor value. Now, press the "Submit" button, then a message will be shown after successfully normalized.

To find highly variable Genes (**Fig. 9**), set the number (2000 by default) for how many Genes will be considered as highly variable to all remained modules where it ('*High Variable Genes Only'*) is indicated (e.g., Dimensionality Reduction, and Trajectory Analysis). It is necessary to press the "Find" button to detect highly variable Genes. To see the computed list of highly

variable Genes, check the “Show Highly Variable Genes” checkbox. The gene list is sorted descending order according to standardized variance, and 10 entities are showing by default. To increase the number of genes for showing, select a number from the drop-down menu, and write a specific gene symbol in the search box for searching individual highly variable gene properties. Top (10 by default) highly variable Genes’ plot will be shown through checking on the “Variable Genes Plot” checkbox (**Fig. 10**). To download the expression matrix of detected high variable Genes, check the “Download Highly Variable Genes(as a .csv file)” box, then press the download button.

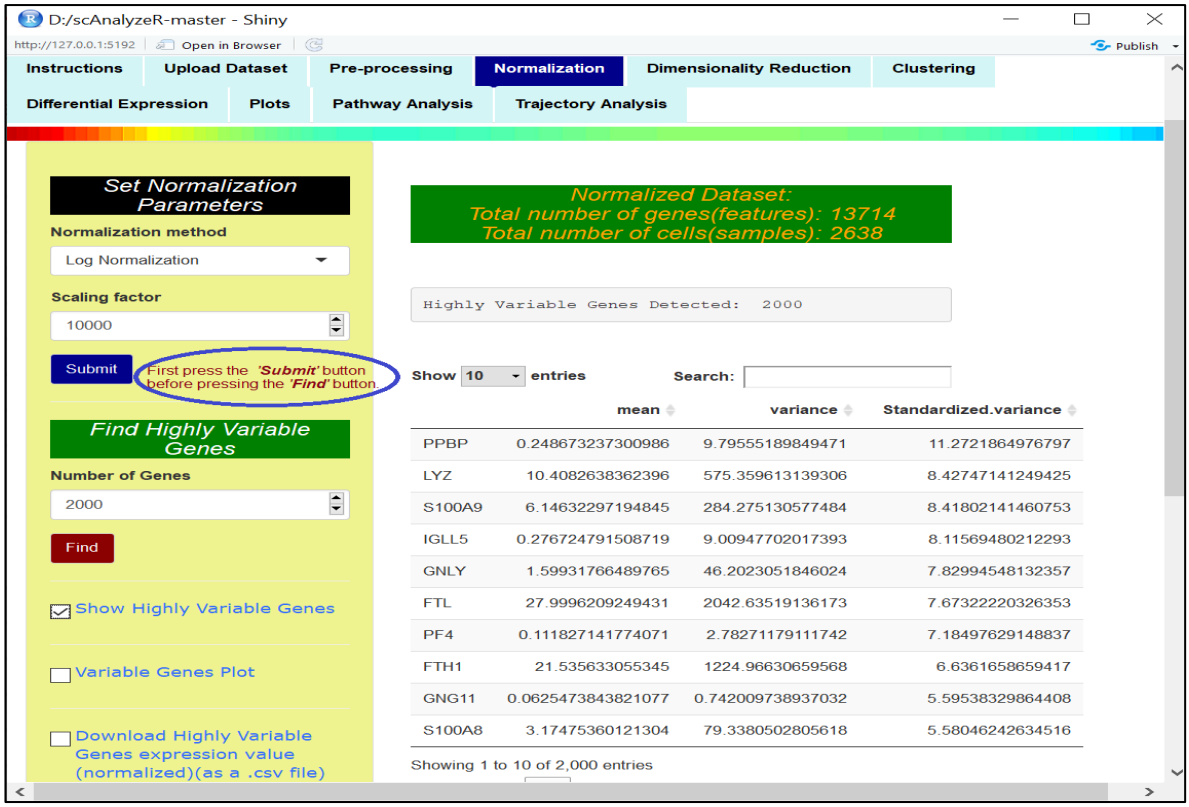


Figure 9. Normalization

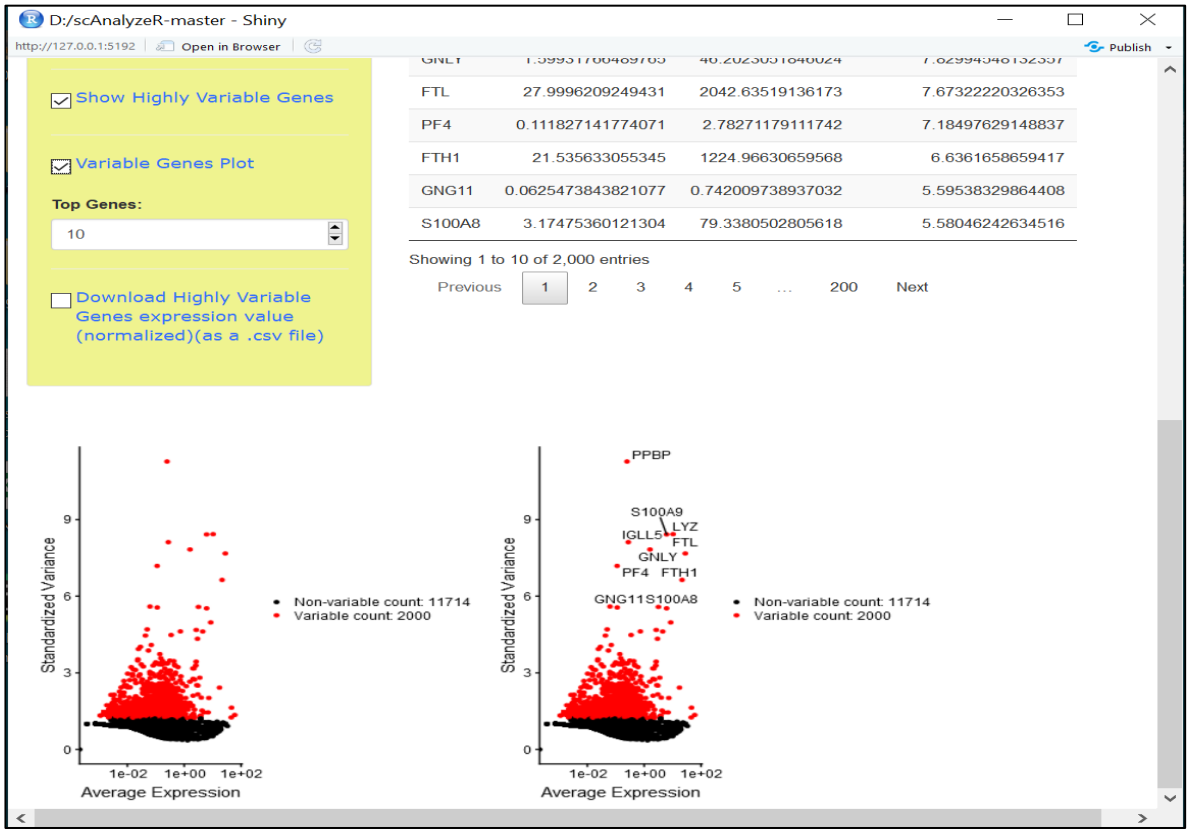


Figure 9. Normalization: Variable genes plot (top 10)

6. Dimensionality Reduction

Now we perform the Principal Component Analysis (PCA) on normalized data. To compute PCA, either use previously (on Normalization module) identified highly variable Genes or all Genes as input. The number of PCs (20 by default) will be computed via clicking the "Compute PCA" button (**Fig. 10**). Various plots can be shown if check the individual checkbox, e.g., PCA plot, t-SNE plot, Elbow plot, etc. However, the 'PCA plot' draws a 2D scatter plot with PC1 and PC2, where each dot represents a cell. The 't-SNE plot' graphs the output from applying the nonlinear dimensionality reduction technique (**Fig. 11**), and the 'Print PCA' presents all genes in an individual PC as well as remarks which genes are belonging to the positive and negative PC scores. It is recommended to choose the right number of PCs that will be processed to the clustering module. The 'Elbow plot' and 'Jack Straw plot' will help you to identify the correct number of PC(s). The 'Elbow plot' ranking PCs according to percentages of variance, each black dot represents a single PC. In our example, we can see there is no significant drop after the PC10; it is concluding that the first 10 PCs (PC1-10) are covering the maximum number of true signals (**Fig. 12**). On the other hand, the 'Jack Straw plot' shows a uniform distribution for each PC with their p-values. Curves (above the black dashed line) with low p-values indicate significant PCs. In our demonstration, there is a significant drop after the first 10-12 PCs. We recommend that repeat the PCA procedure with different number of PCs, and try to compute PCA with 10 or more number of PCs.

The PC heatmap helps to explore the primary sources of heterogeneity in the dataset. To draw the heatmap plot, check the "PC Heatmap" box, then the interface panel will be visible with the default parameters setting. The user can set different values, if necessary. After setting arguments, click the "Show plot" button to display the heatmap. The heatmap shows a correlation for an equal number of genes from +ve and -ve PC's sets. In this case, it shows 30 genes (15 with positive PC scores and another 15 with negative PC scores) in the first two principle components (PC1 and PC2) (**Fig. 13**).

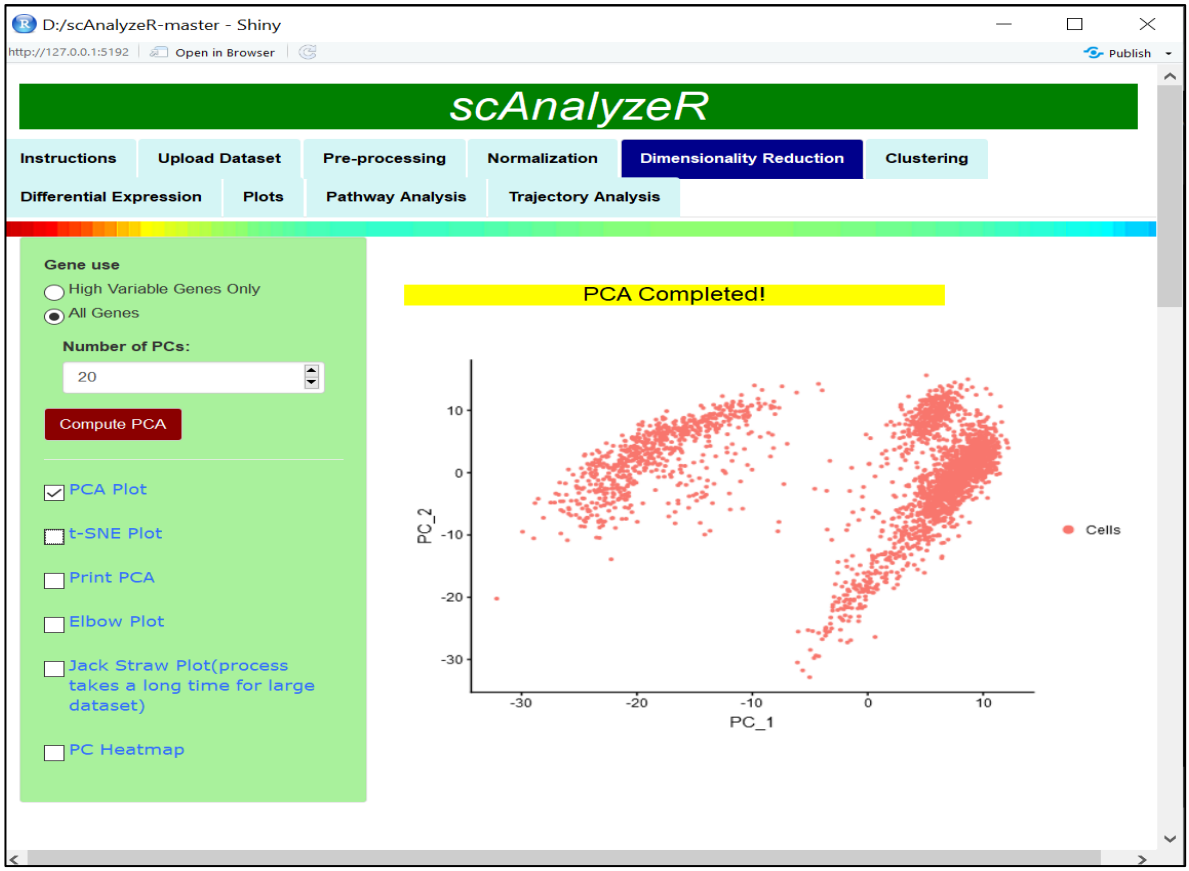


Figure 10. Dimensionality Reduction: Principal component analysis (PCA) for 20 PCs

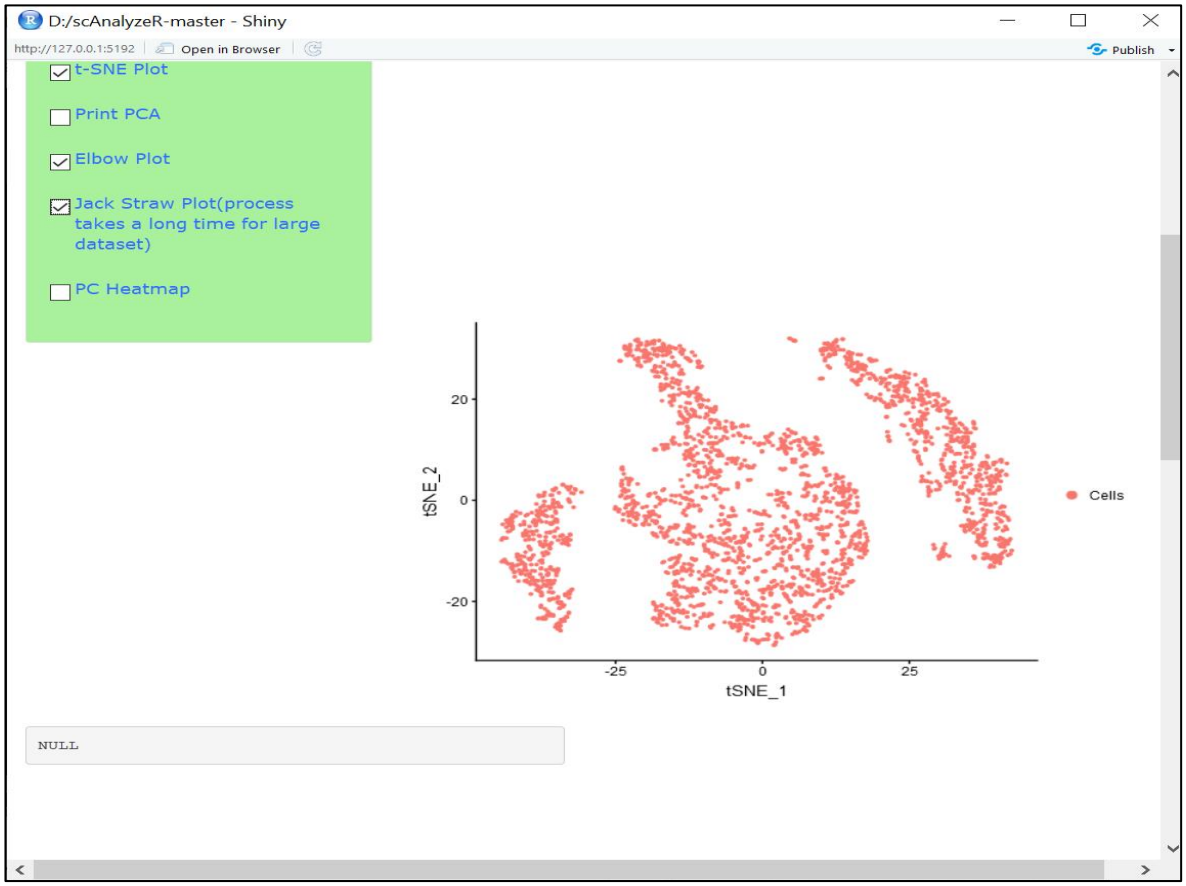


Figure 11. Dimensionality Reduction: t-SNE plot

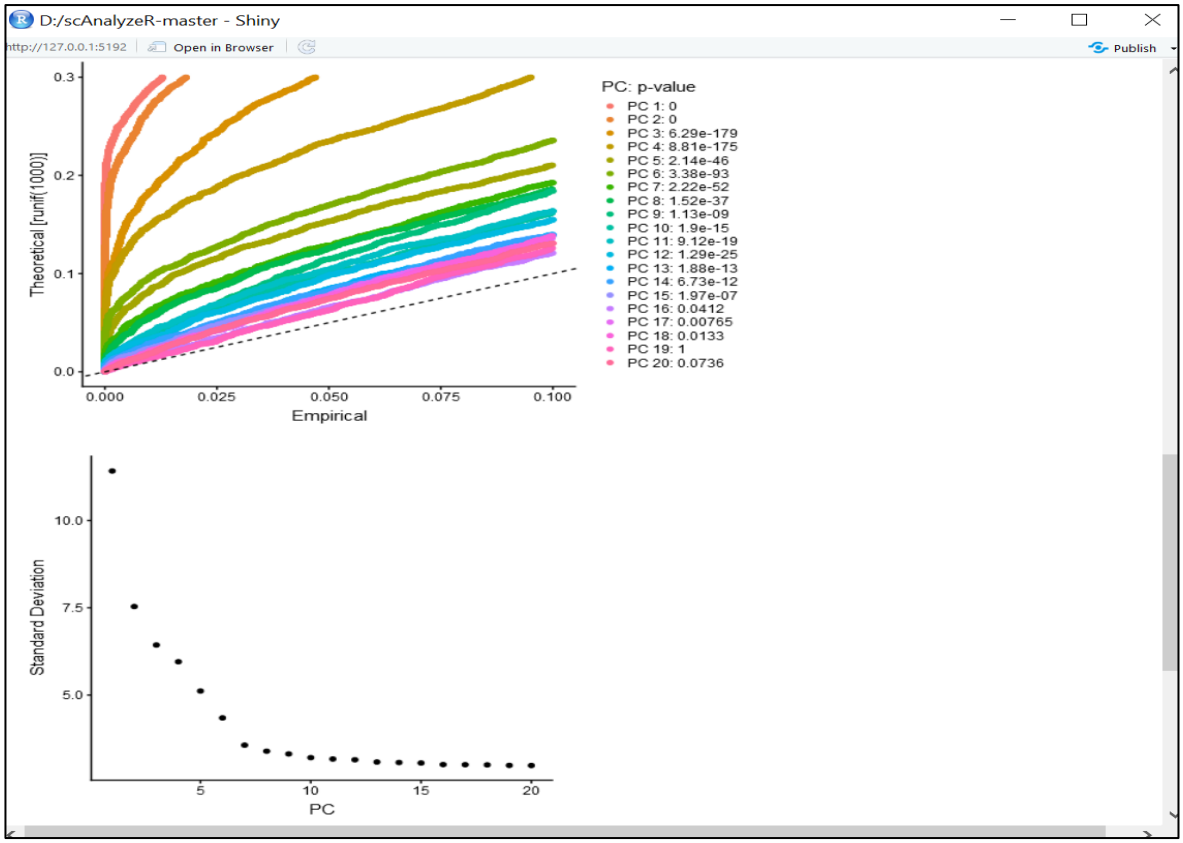


Figure 12. Dimensionality Reduction: Jack Straw plot(top), and Elbow plot(bottom) for 20 PCs

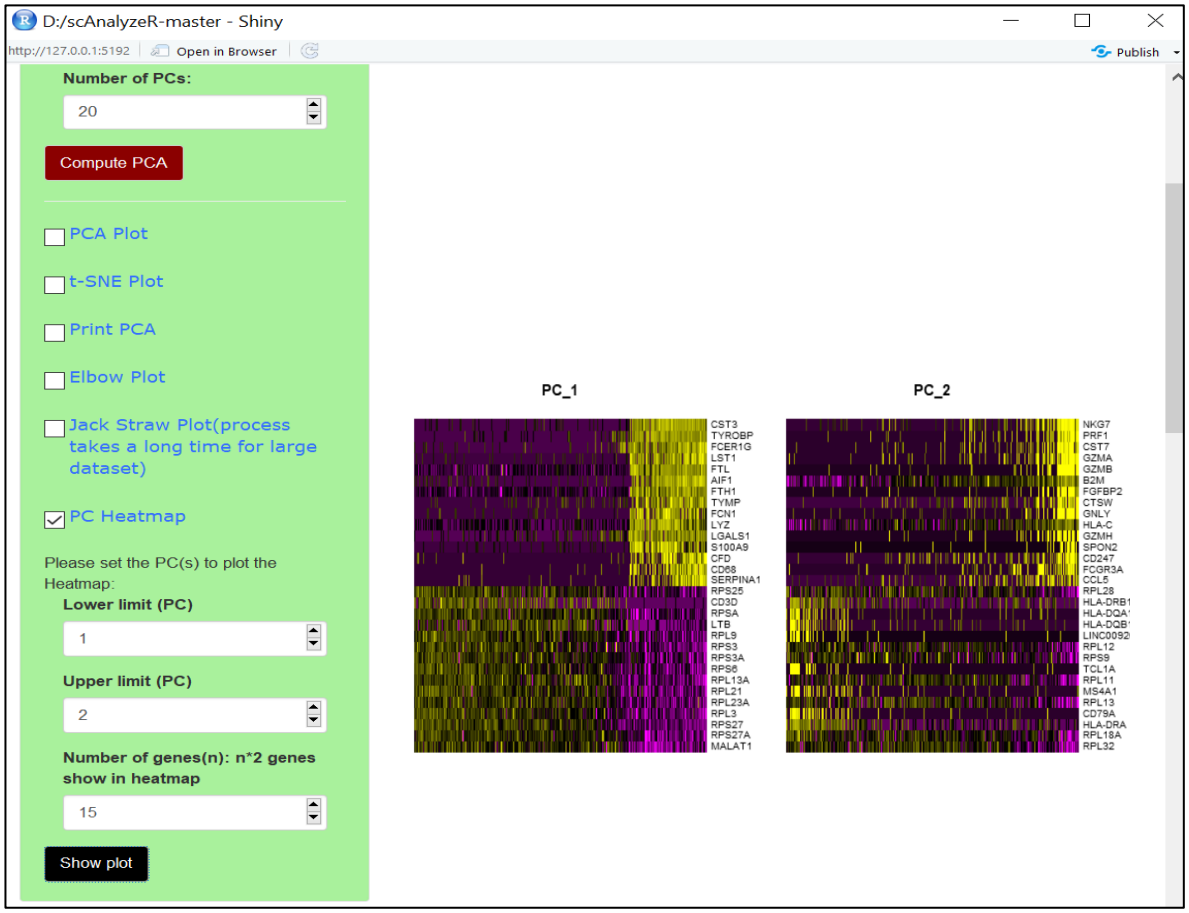


Figure 13. Dimensionality Reduction: Heatmap for top two PCs (PC1, PC2)

7. Clustering

In this module, cells are clustered based on pre-calculated PCs (PC1-PC10). The user can make clusters using either default parameter values or different values. It is highly recommended that use only the significant PCs for better results, for instance, we use first 10 PCs for performing clustering cells. The default value 0.5 is provided for the resolution field, and the minimum value of the resolution is 0 (zero), if you increase the resolution, it will also increase the number of clusters. Generally, setting this value between 0.4 -1.2 gives good results where the dataset contains about 3k cells. In this case, we set the resolution to 0.4, and the default clustering method (*Louvain algorithm*) was selected from three available clustering algorithms (**Fig. 14**). However, it is necessary to click the "Do cluster" button to perform the clustering. After pressing the button, then click checkboxes to show the clusters' t-SNE plot and bar plot (**Fig. 15**). It is mentioned that this clustering results will be applied to later modules.

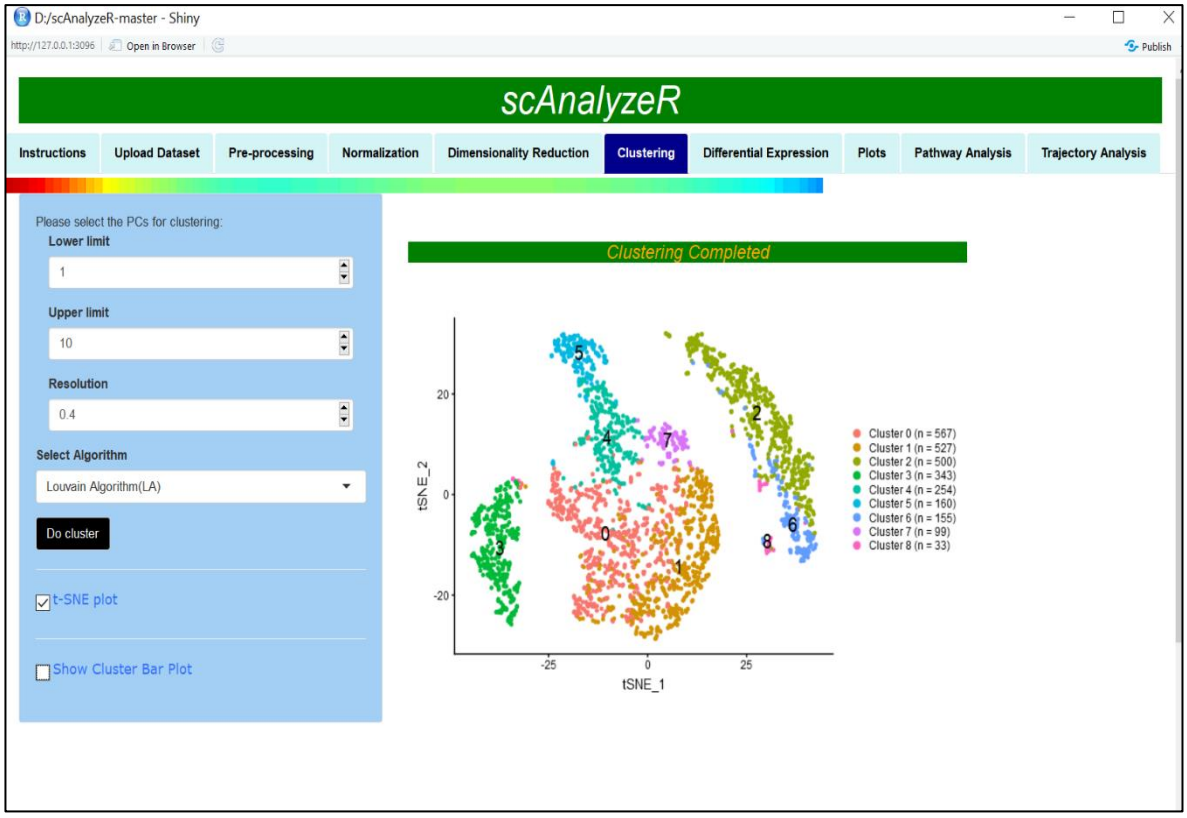


Figure 14. Clustering: t-SNE plot

Figure 15. Clustering: Clustering Bar plot

8. Differential Expression Analysis

In differential expression analysis, you can find markers (differentially expressed genes) that define clusters. The differential expression module is divided into three submodules, namely, 'find all markers,' 'find markers by cluster,' and 'find markers by cluster vs other clusters'.

To find differentially expressed genes for all clusters, click the "Find All markers" button (**Fig. 16-27**). You can choose a different test method from the "Select Test Method" drop-down menu, and then press the button to find markers. The "Save all markers as csv" button and markers bar plot will be shown when the marker finding process is completed. You can also save the full list of markers as a csv file. To show the full table of marker list, check the "Show All Markers" checkbox, and check the "Filtering Markers" checkbox for filtering the generated marker list. After checking the filtering checkbox, the filtering interface panel will be visible with necessary default setting parameters. If you choose "Positive only" from "Markers Selection," it means that both the "Show Filtered list" and "Show Top genes" also have the

same selection from the filtered list. Similar action will be done if the “Negative Only” is selected. You can see the top differentially expressed genes (10 by default) from each cluster clicking via “Show Top genes” as well as press a particular download button (‘Save All Top genes as a text file,’ ‘Save +ve Top genes as a text file’ or ‘Save -ve Top genes as a text file’) to save the top markers list.

In the ‘find marker by cluster’ submodule (**Fig. 28- 35**), a list of markers will be identified in a particular cluster (cluster ‘0’ is selected by default), compared to all remained clusters. For our example, we used cluster ‘0’, compared to all other clusters. The marker list will be shown and saved, which is similar to the ‘Find all markers’ submodule. Besides, a heatmap plot will be displayed clicking via the “Show Heatmap” checkbox for that particular cluster’s top markers as well as you can enlarge the heatmap plot for that specific cluster clicking on the “Zoom the cluster” checkbox.

The ‘find markers by cluster vs other clusters’ submodule (**Fig. 36-41**) can detect markers by cluster(s) versus cluster(s). First, write the cluster(s) number in both textboxes, i.e., “Select cluster/s” and “Select complementary cluster/s”. The cluster number must be separated by a comma, and no common cluster number is allowed as an input for both textboxes. If the same cluster number is written in both input boxes, it will be shown an error message, “*Error: No features pass logfc.threshold threshold*”. Other functionalities of this module are identical to the ‘find marker by cluster’ submodule. In this case, we used clusters ‘0’, and ‘1’, compared with clusters ‘2’, ‘6’, and ‘8’.

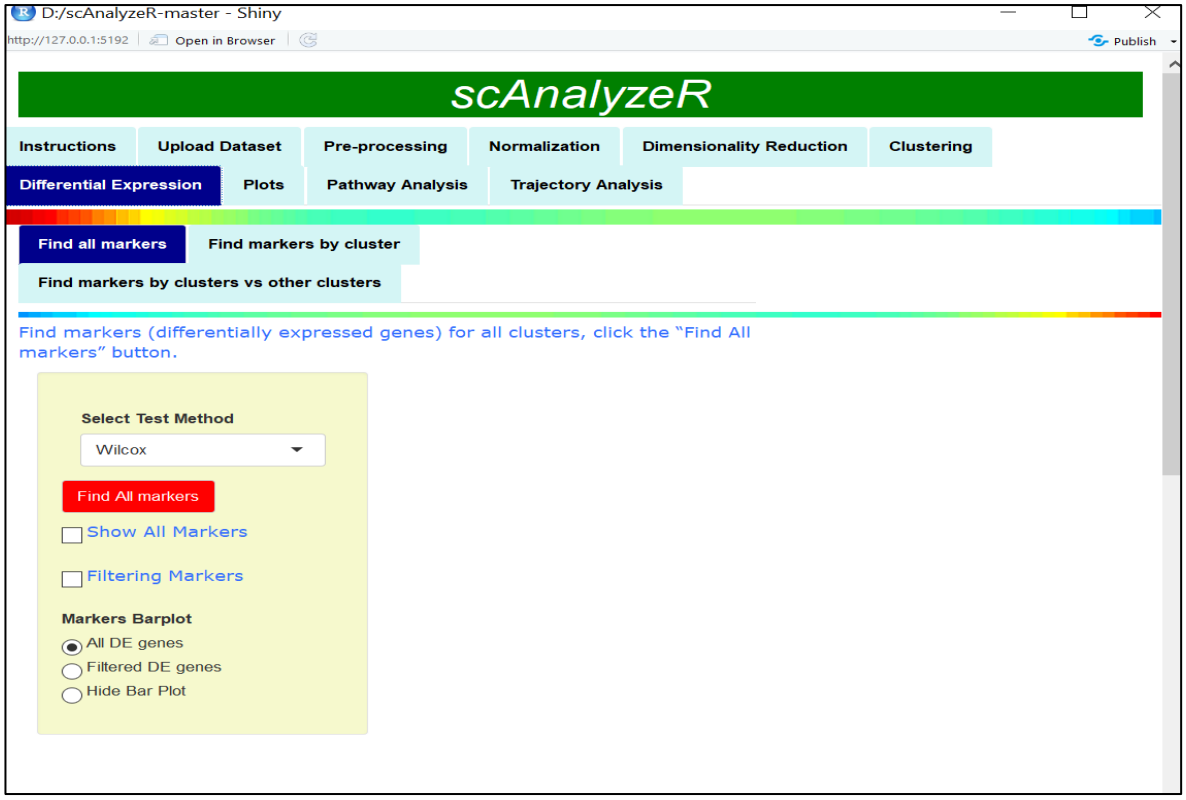


Figure 16. Differential Expression (Find all markers): the user interface

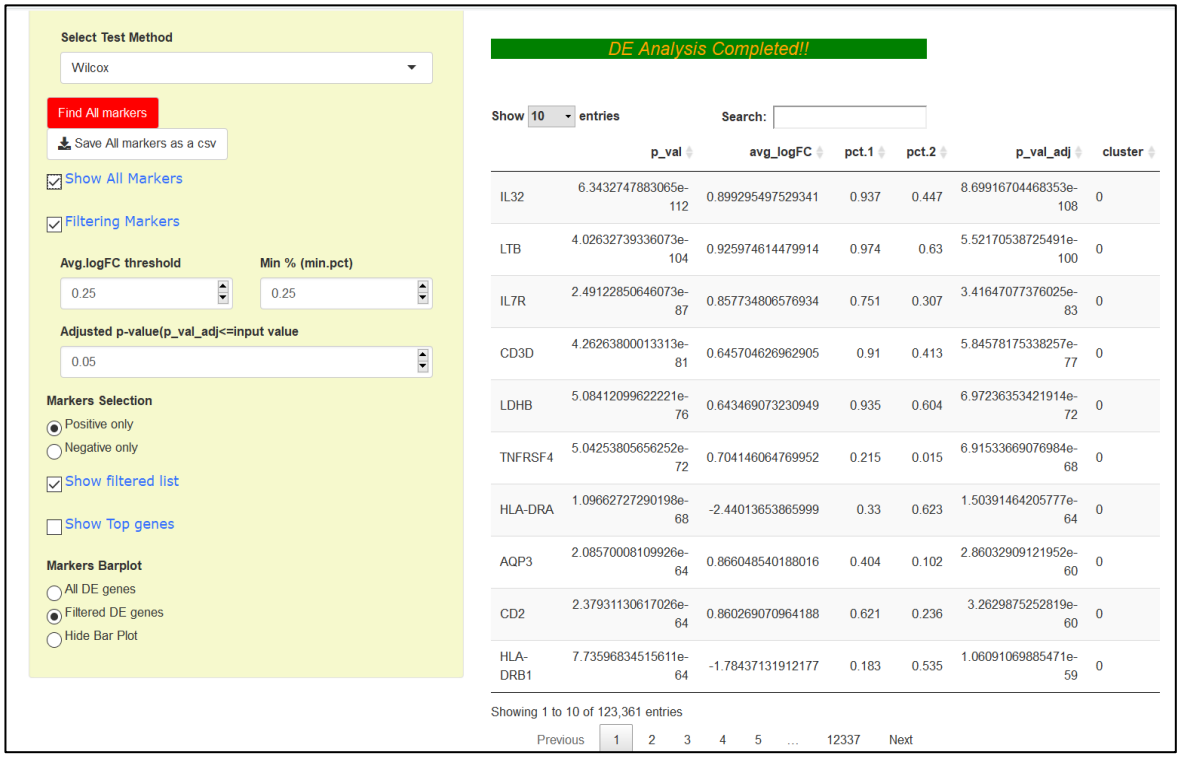


Figure 17. Differential Expression (Find all markers): The table shows all DE genes (up & down-regulated) before filtering

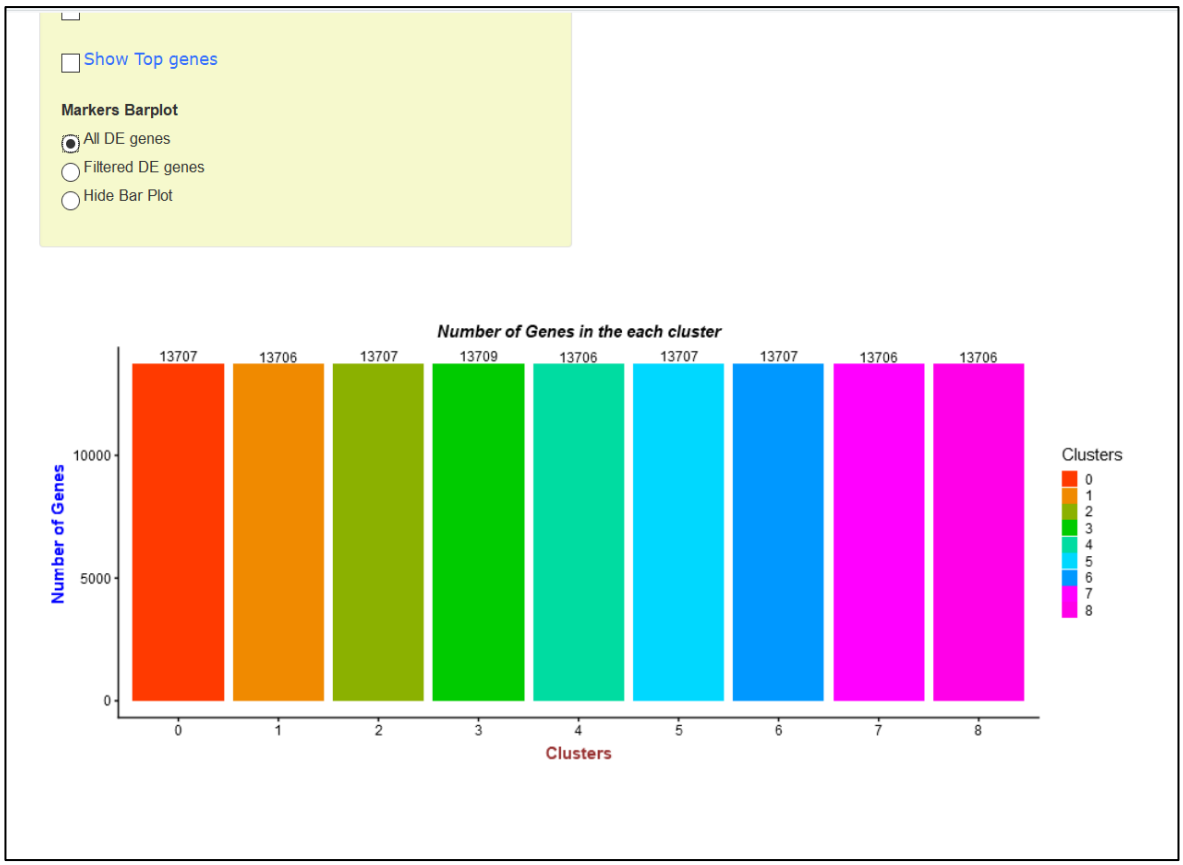


Figure 18. Differential Expression (Find all markers): The Bar plot shows number of DE genes identified in each cluster before filtering

Figure 19. Differential Expression (Find all markers): The table shows down-regulated DE genes after filtering

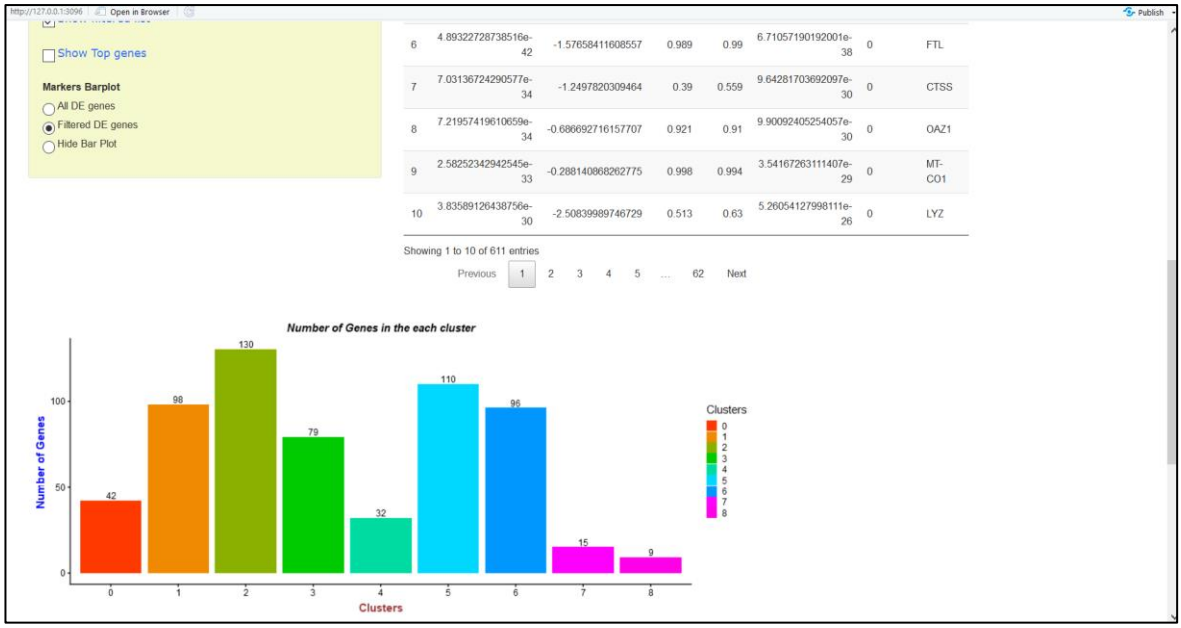


Figure 20. Differential Expression (Find all markers): The bar plot shows number of down-regulated DE genes in each cluster after filtering

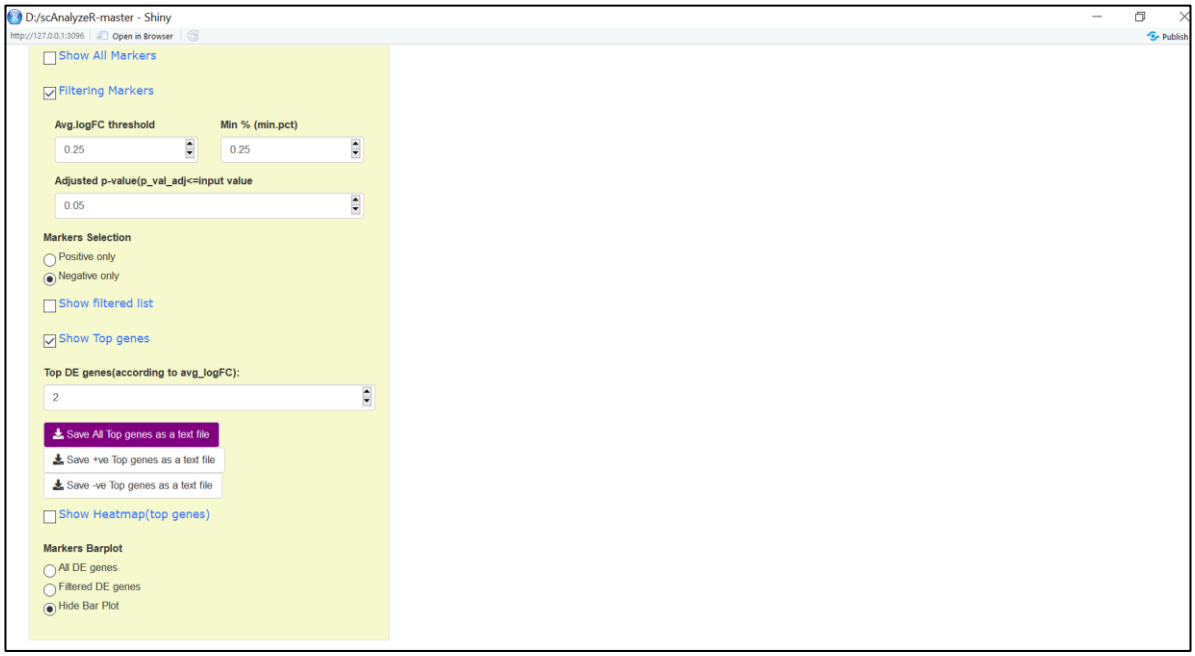
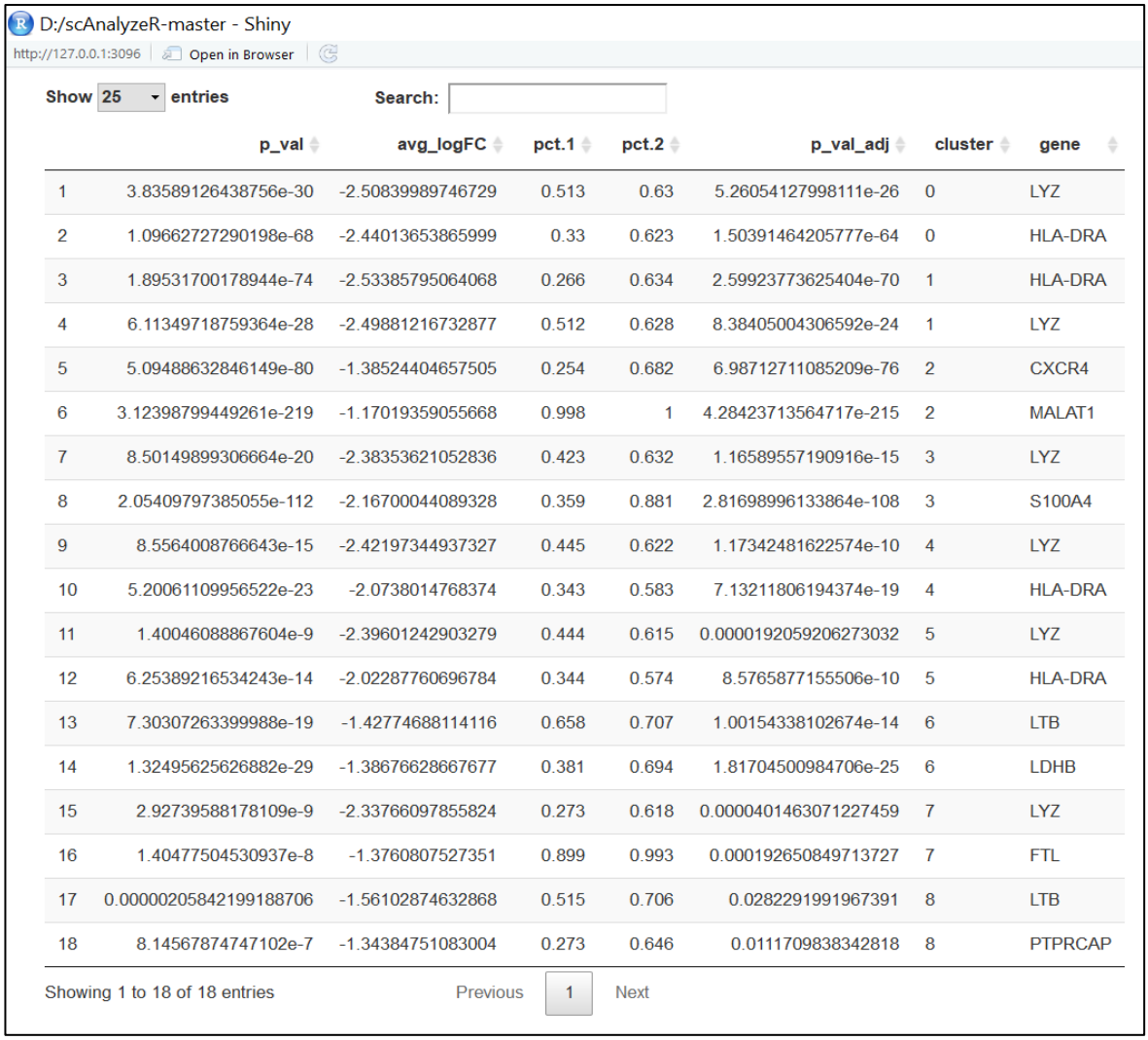


Figure 21. Differential Expression (Find all markers): DE genes downloading options after filtering



Show 25 entries Search:

	p_val	avg_logFC	pct.1	pct.2	p_val_adj	cluster	gene
1	3.83589126438756e-30	-2.50839989746729	0.513	0.63	5.26054127998111e-26	0	LYZ
2	1.09662727290198e-68	-2.44013653865999	0.33	0.623	1.50391464205777e-64	0	HLA-DRA
3	1.89531700178944e-74	-2.53385795064068	0.266	0.634	2.59923773625404e-70	1	HLA-DRA
4	6.11349718759364e-28	-2.49881216732877	0.512	0.628	8.38405004306592e-24	1	LYZ
5	5.09488632846149e-80	-1.38524404657505	0.254	0.682	6.98712711085209e-76	2	CXCR4
6	3.12398799449261e-219	-1.17019359055668	0.998	1	4.28423713564717e-215	2	MALAT1
7	8.50149899306664e-20	-2.38353621052836	0.423	0.632	1.16589557190916e-15	3	LYZ
8	2.05409797385055e-112	-2.16700044089328	0.359	0.881	2.81698996133864e-108	3	S100A4
9	8.5564008766643e-15	-2.42197344937327	0.445	0.622	1.17342481622574e-10	4	LYZ
10	5.20061109956522e-23	-2.0738014768374	0.343	0.583	7.13211806194374e-19	4	HLA-DRA
11	1.40046088867604e-9	-2.39601242903279	0.444	0.615	0.0000192059206273032	5	LYZ
12	6.25389216534243e-14	-2.02287760696784	0.344	0.574	8.5765877155506e-10	5	HLA-DRA
13	7.30307263399988e-19	-1.42774688114116	0.658	0.707	1.00154338102674e-14	6	LTB
14	1.32495625626882e-29	-1.38676628667677	0.381	0.694	1.81704500984706e-25	6	LDHB
15	2.92739588178109e-9	-2.33766097855824	0.273	0.618	0.0000401463071227459	7	LYZ
16	1.40477504530937e-8	-1.3760807527351	0.899	0.993	0.000192650849713727	7	FTL
17	0.00000205842199188706	-1.56102874632868	0.515	0.706	0.0282291991967391	8	LTB
18	8.14567874747102e-7	-1.34384751083004	0.273	0.646	0.0111709838342818	8	PTPRCAP

Showing 1 to 18 of 18 entries Previous 1 Next

Figure 22. Differential Expression (Find all markers): The table shows top-two down-regulated DE genes in each cluster after filtering

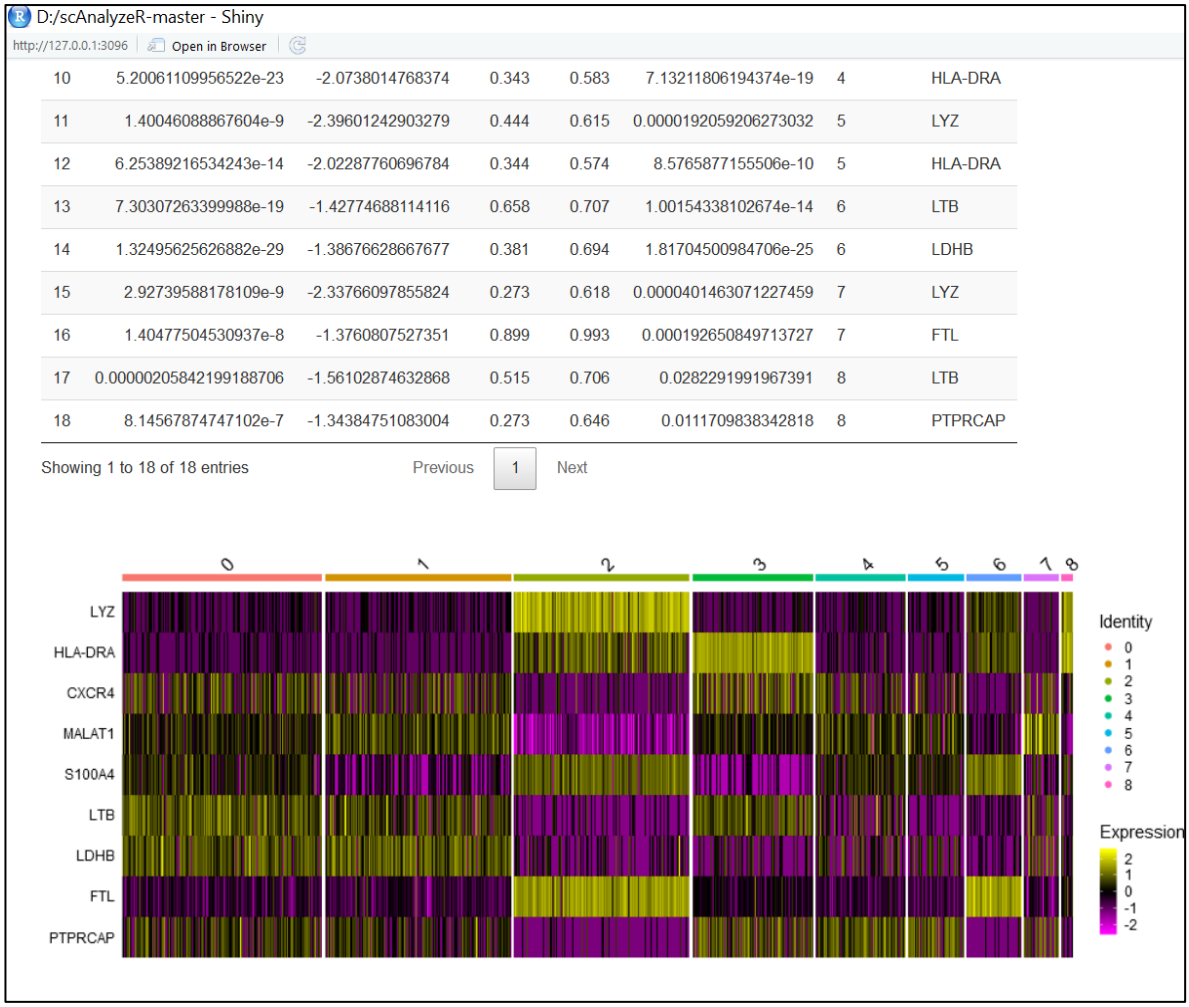


Figure 23. Differential Expression (Find all markers): The heatmap illustrates expressions for top-two down-regulated DE genes (unique genes only) in each cluster after filtering

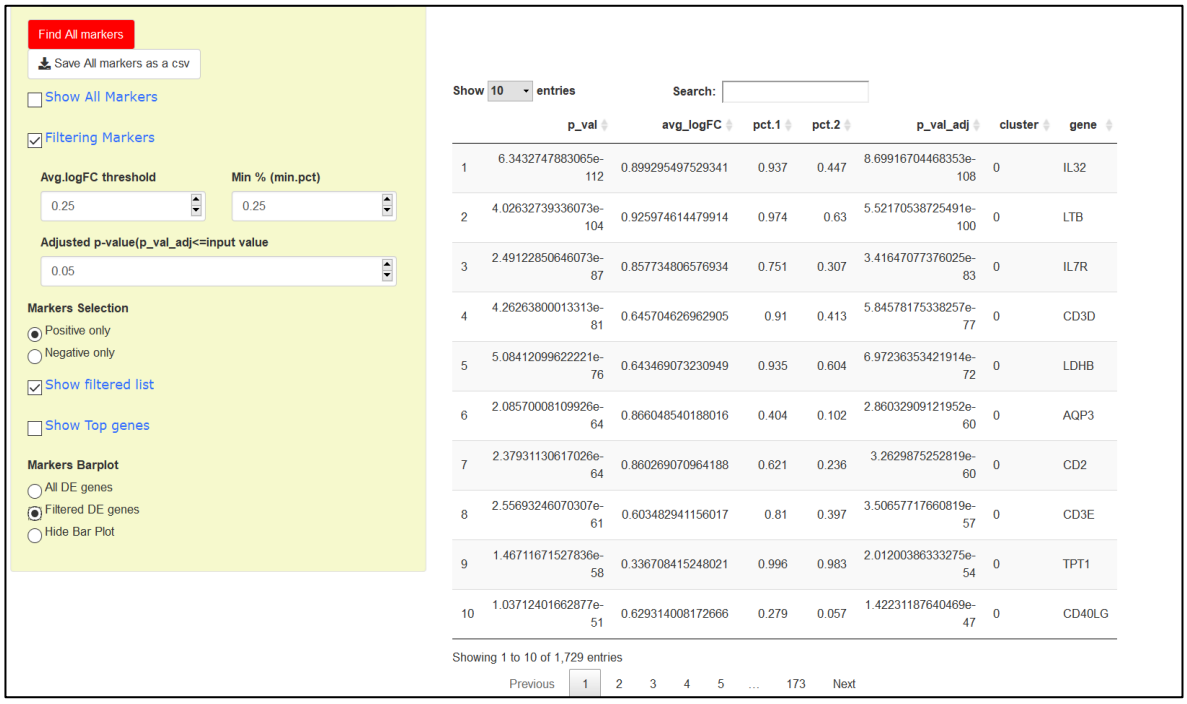


Figure 24. Differential Expression (Find all markers): The table shows up-regulated DE genes after filtering

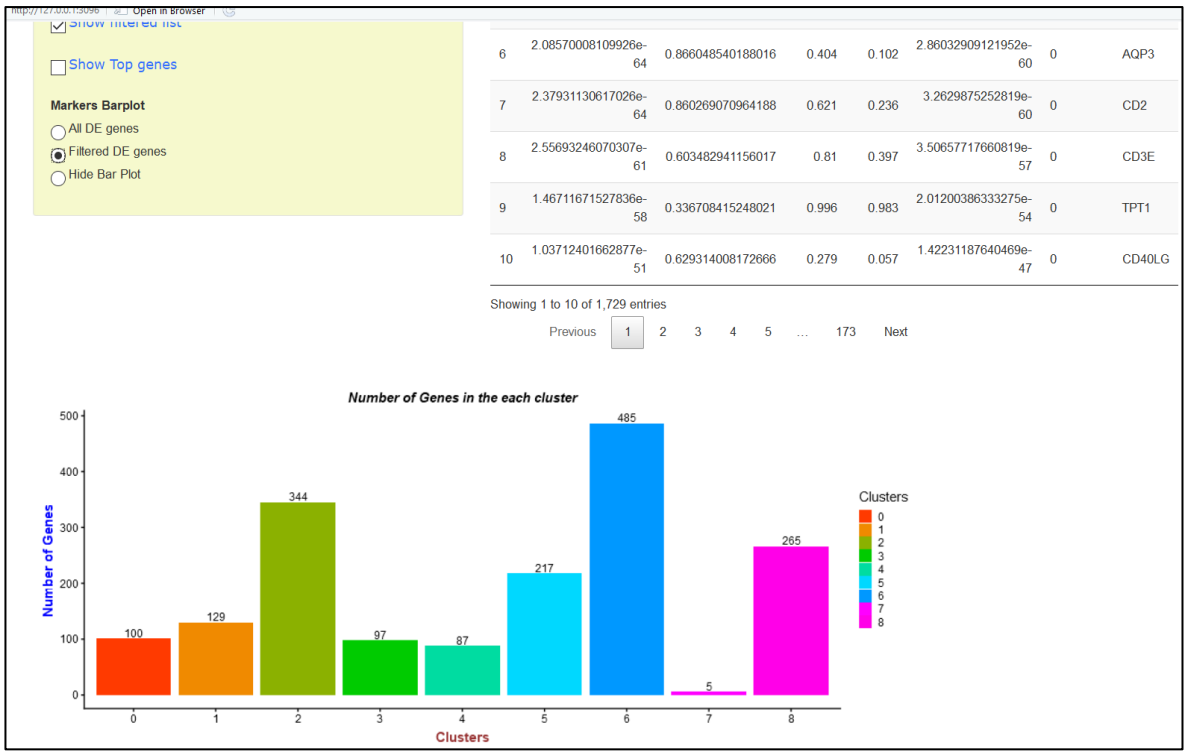
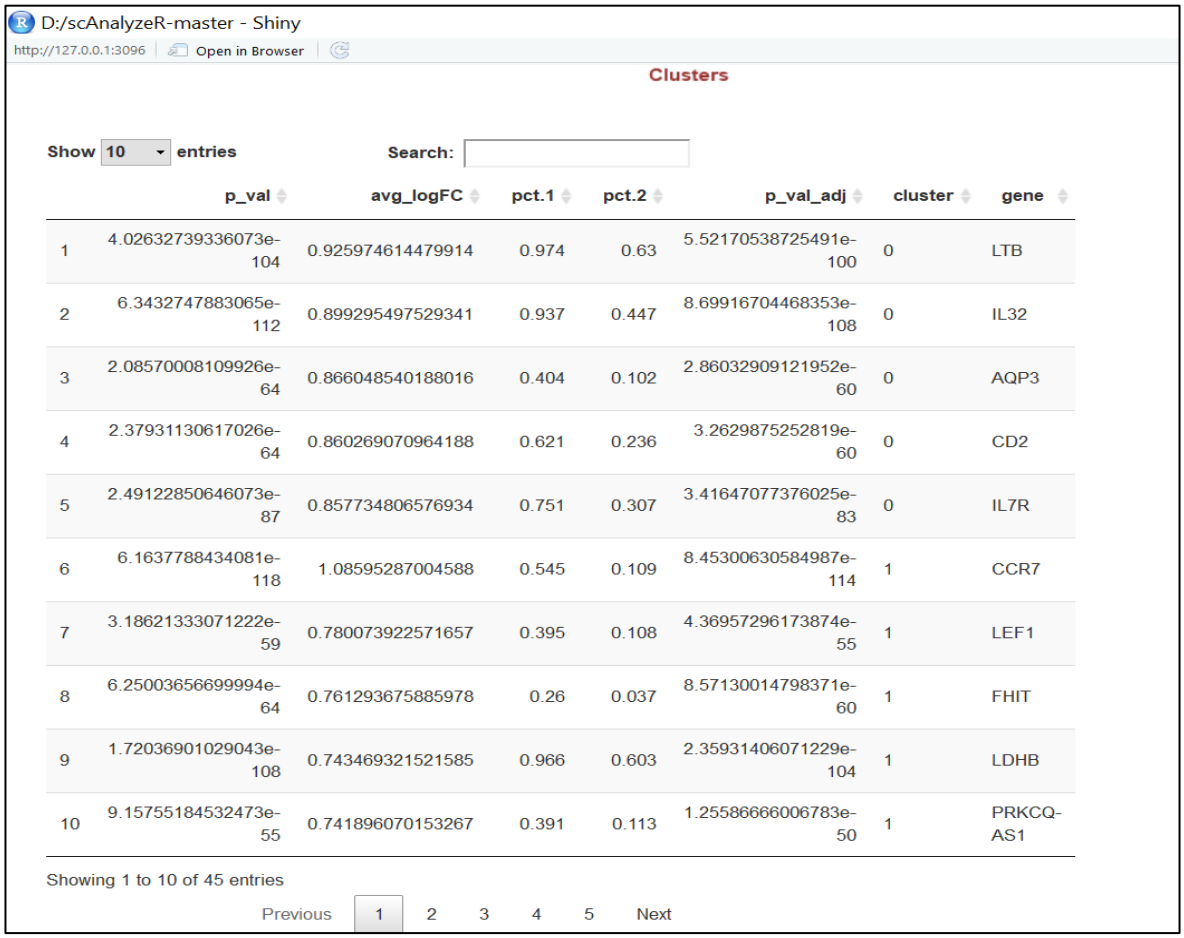


Figure 25. Differential Expression (Find all markers): The bar plot shows number of up-regulated DE genes in each cluster after filtering



	p_val	avg_logFC	pct.1	pct.2	p_val_adj	cluster	gene
1	4.02632739336073e-104	0.925974614479914	0.974	0.63	5.52170538725491e-100	0	LTB
2	6.3432747883065e-112	0.899295497529341	0.937	0.447	8.69916704468353e-108	0	IL32
3	2.08570008109926e-64	0.866048540188016	0.404	0.102	2.86032909121952e-60	0	AQP3
4	2.37931130617026e-64	0.860269070964188	0.621	0.236	3.2629875252819e-60	0	CD2
5	2.49122850646073e-87	0.857734806576934	0.751	0.307	3.41647077376025e-83	0	IL7R
6	6.1637788434081e-118	1.08595287004588	0.545	0.109	8.45300630584987e-114	1	CCR7
7	3.18621333071222e-59	0.780073922571657	0.395	0.108	4.36957296173874e-55	1	LEF1
8	6.25003656699994e-64	0.761293675885978	0.26	0.037	8.57130014798371e-60	1	FHIT
9	1.72036901029043e-108	0.743469321521585	0.966	0.603	2.35931406071229e-104	1	LDHB
10	9.15755184532473e-55	0.741896070153267	0.391	0.113	1.25586666006783e-50	1	PRKCQ-AS1

Figure 26. Differential Expression (Find all markers): The table shows top-five up-regulated DE genes in each cluster after filtering

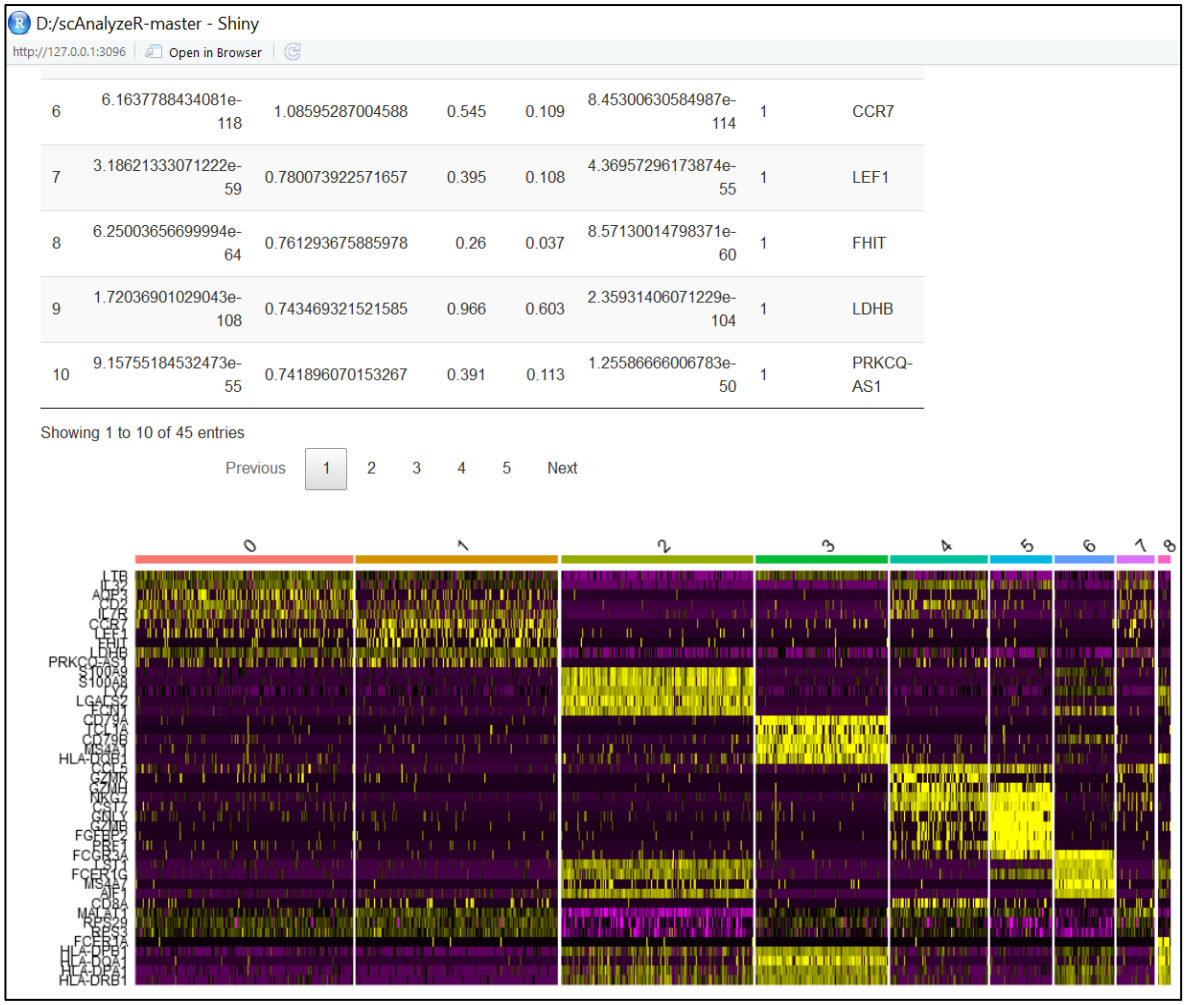


Figure 27. Differential Expression (Find all markers): The heatmap illustrates expressions for top-five up-regulated DE genes (unique genes only) in each cluster after filtering

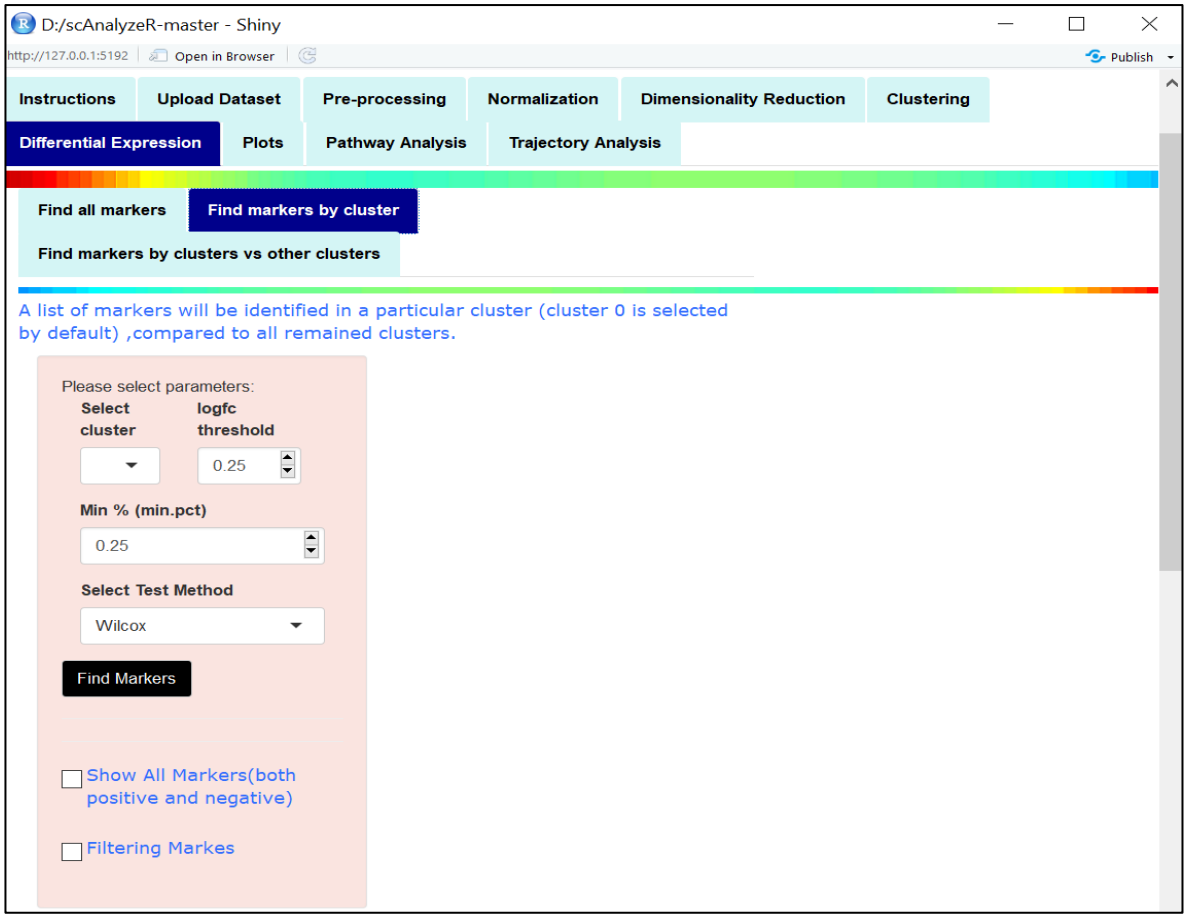
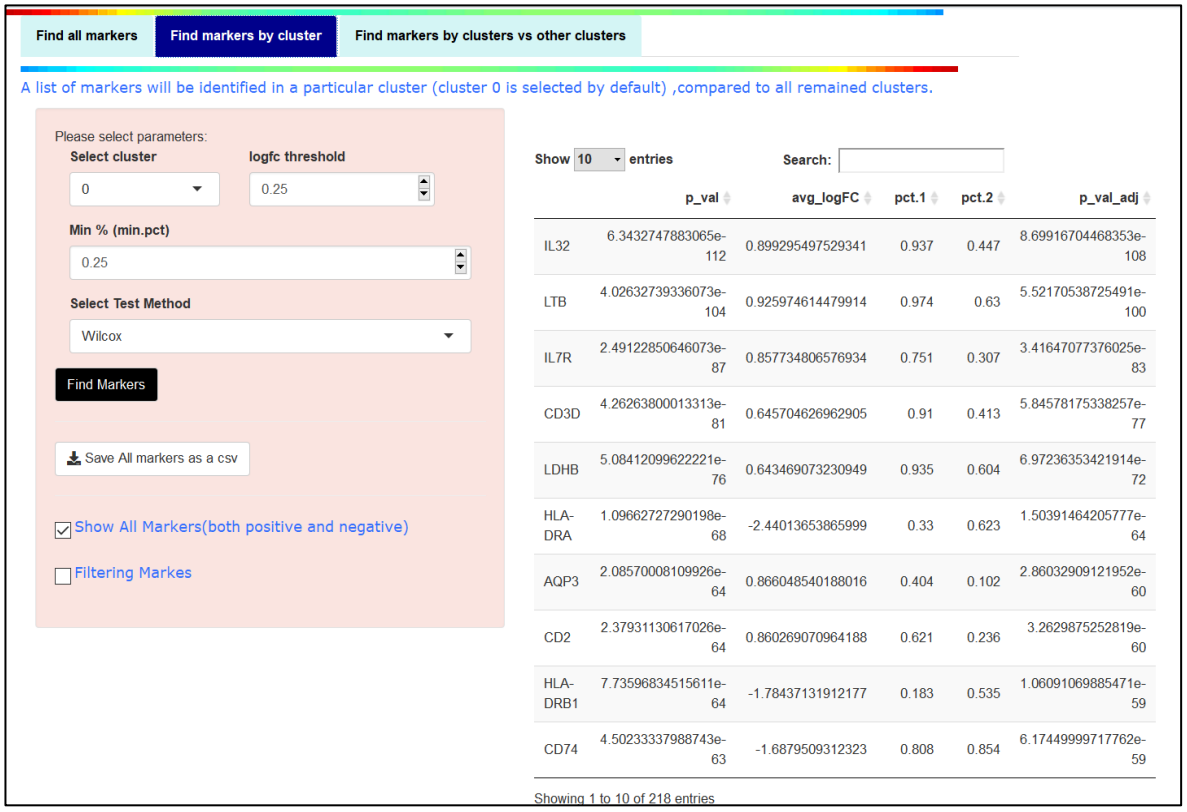


Figure 28. Differential Expression (Find markers by cluster): the user interface



	p_val	avg_logFC	pct.1	pct.2	p_val_adj
IL32	6.3432747883065e-112	0.899295497529341	0.937	0.447	8.69916704468353e-108
LTB	4.02632739336073e-104	0.925974614479914	0.974	0.63	5.52170538725491e-100
IL7R	2.49122850646073e-87	0.857734806576934	0.751	0.307	3.41647077376025e-83
CD3D	4.26263800013313e-81	0.645704626962905	0.91	0.413	5.84578175338257e-77
LDHB	5.08412099622221e-76	0.643469073230949	0.935	0.604	6.97236353421914e-72
HLA-DRA	1.09662727290198e-68	-2.44013653865999	0.33	0.623	1.50391464205777e-64
AQP3	2.08570008109926e-64	0.866048540188016	0.404	0.102	2.86032909121952e-60
CD2	2.37931130617026e-64	0.860269070964188	0.621	0.236	3.2629875252819e-60
HLA-DRB1	7.73596834515611e-64	-1.78437131912177	0.183	0.535	1.06091069885471e-59
CD74	4.50233337988743e-63	-1.6879509312323	0.808	0.854	6.17449999717762e-59

Figure 29. Differential Expression (Find markers by cluster): The table shows DE genes (both up & down-regulated) in cluster '0'

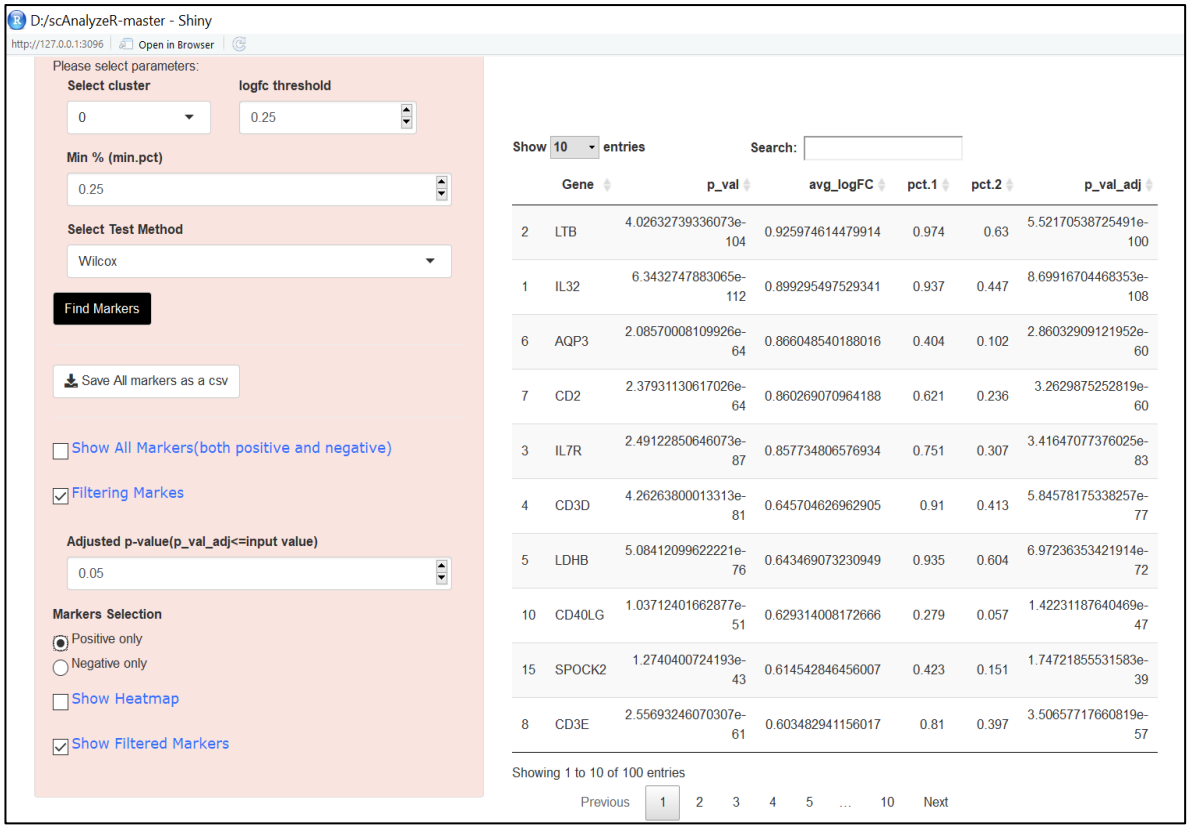


Figure 30. Differential Expression (Find markers by cluster): The table shows up-regulated DE genes (in cluster '0') after filtering

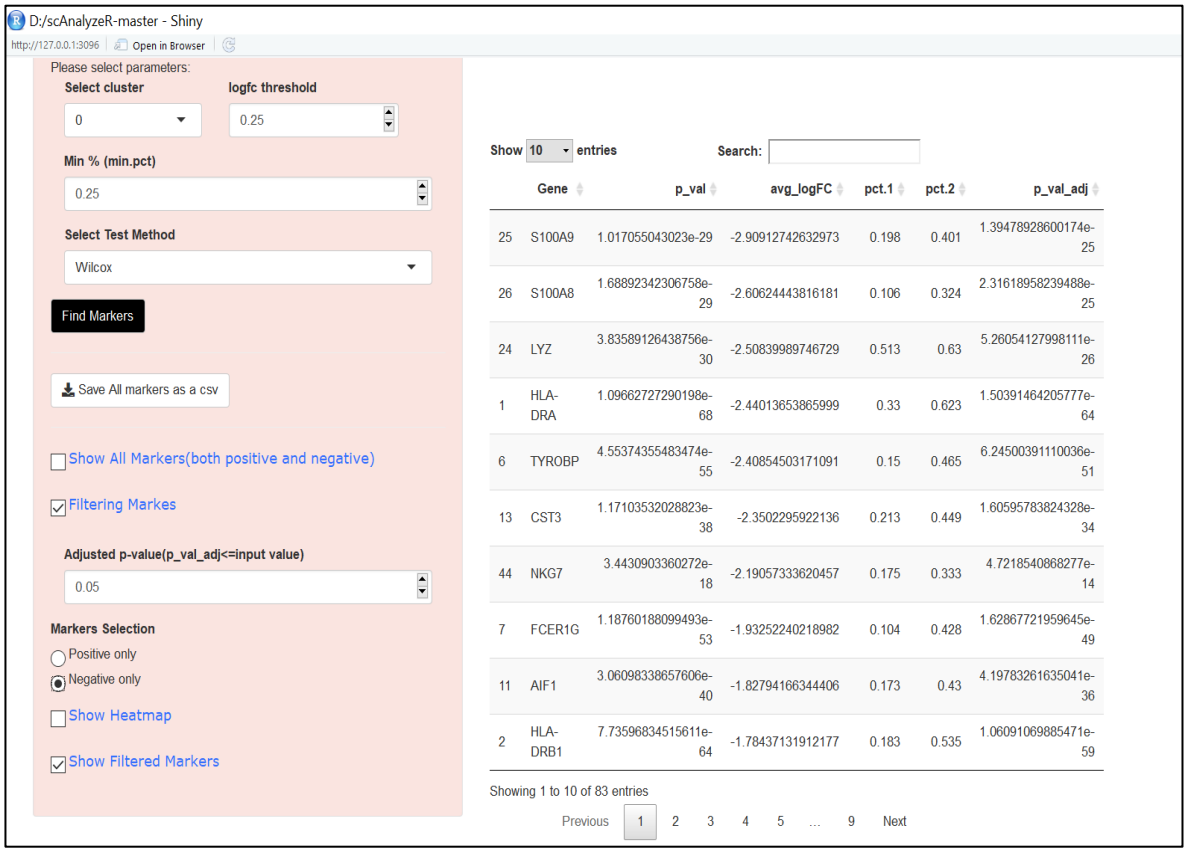


Figure 31. Differential Expression (Find markers by cluster): The table shows down-regulated DE genes (in cluster '0') after filtering

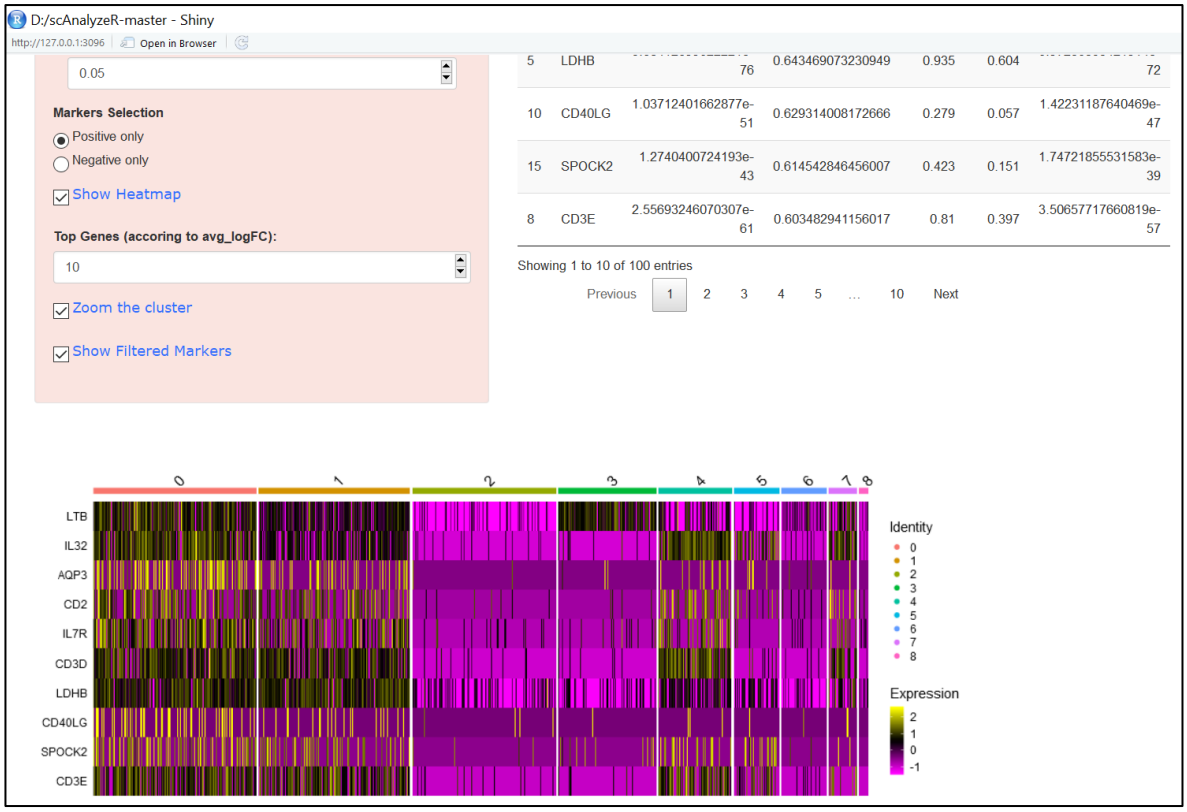


Figure 32. Differential Expression (Find markers by cluster): The heatmap illustrates expressions for top-ten up-regulated DE genes after filtering

Figure 33. Differential Expression (Find markers by cluster): The heatmap(bottom) illustrates expressions for top-ten up-regulated DE genes in the cluster '0' only (zooming the cluster '0'), after filtering

Figure 34. Differential Expression (Find markers by cluster): The heatmap illustrates expressions for top-ten down-regulated DE genes after filtering

Figure 35. Differential Expression (Find markers by cluster): The heatmap(bottom) illustrates expressions for top-ten down-regulated DE genes in the cluster '0' only (zooming the cluster '0'), after filtering

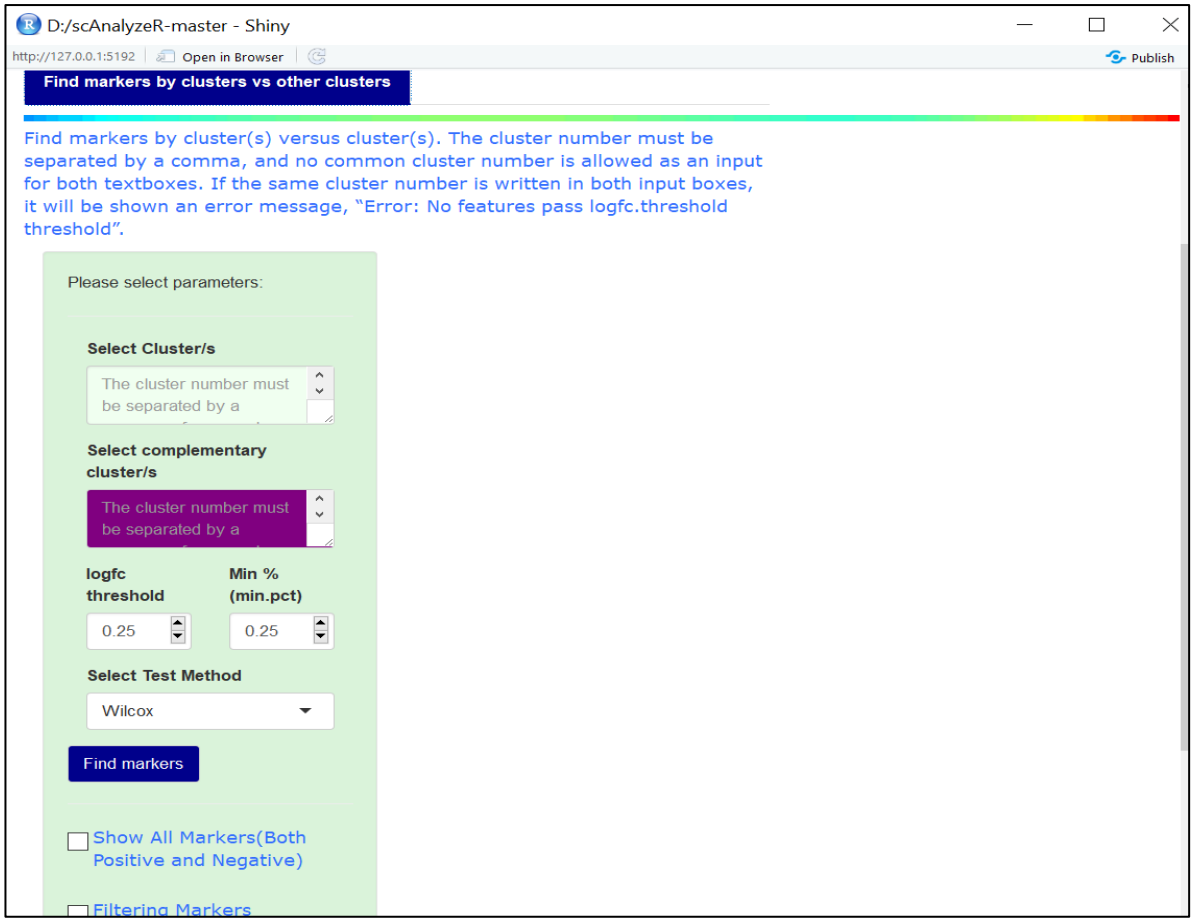


Figure 36. Differential Expression (Find markers by clusters vs other clusters): the user interface

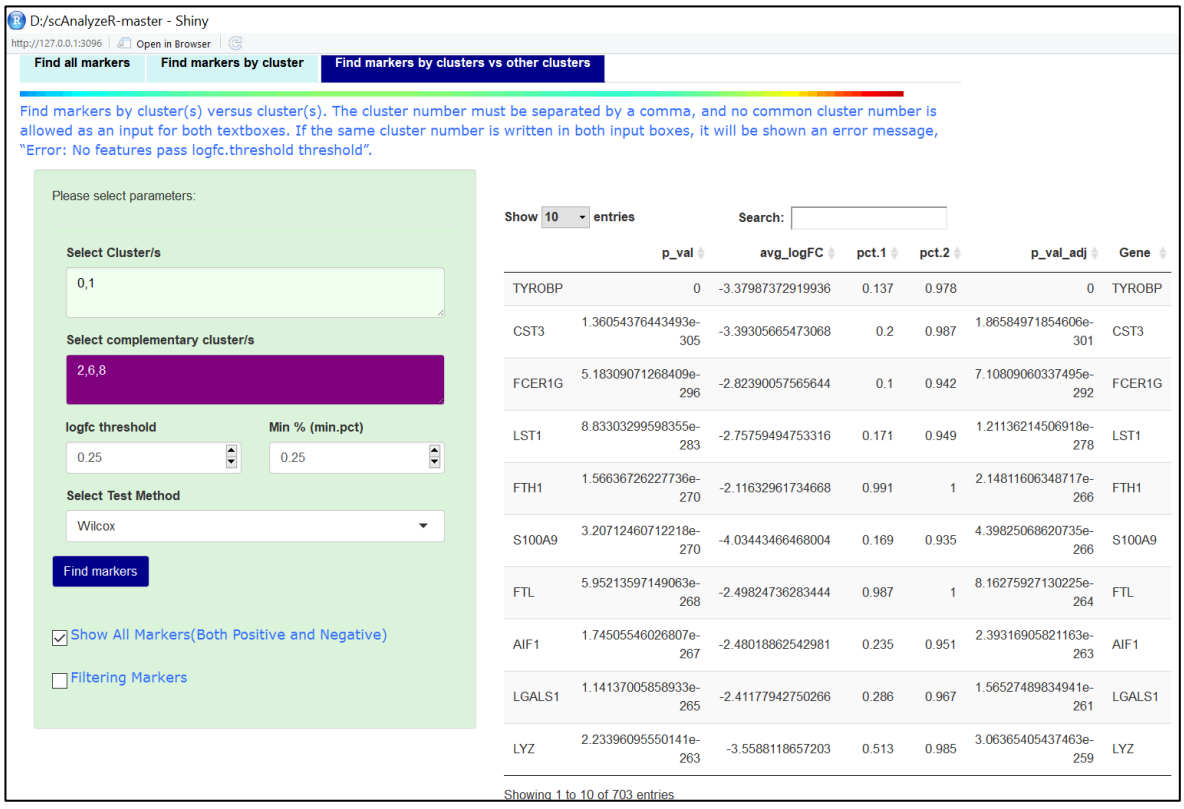


Figure 37. Differential Expression (Find markers by clusters vs other clusters): The table shows both up & down-regulated DE genes

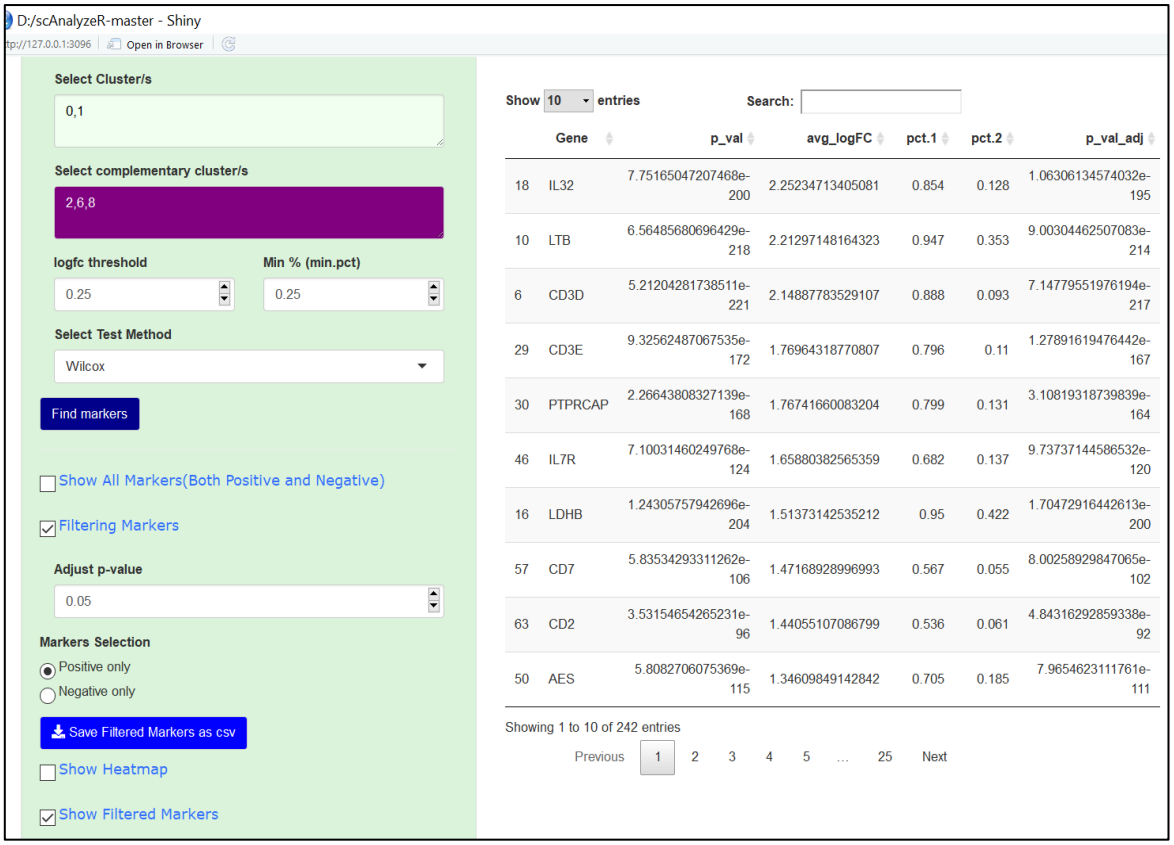


Figure 38. Differential Expression (Find markers by clusters vs other clusters): The table shows only up-regulated DE genes after filtering

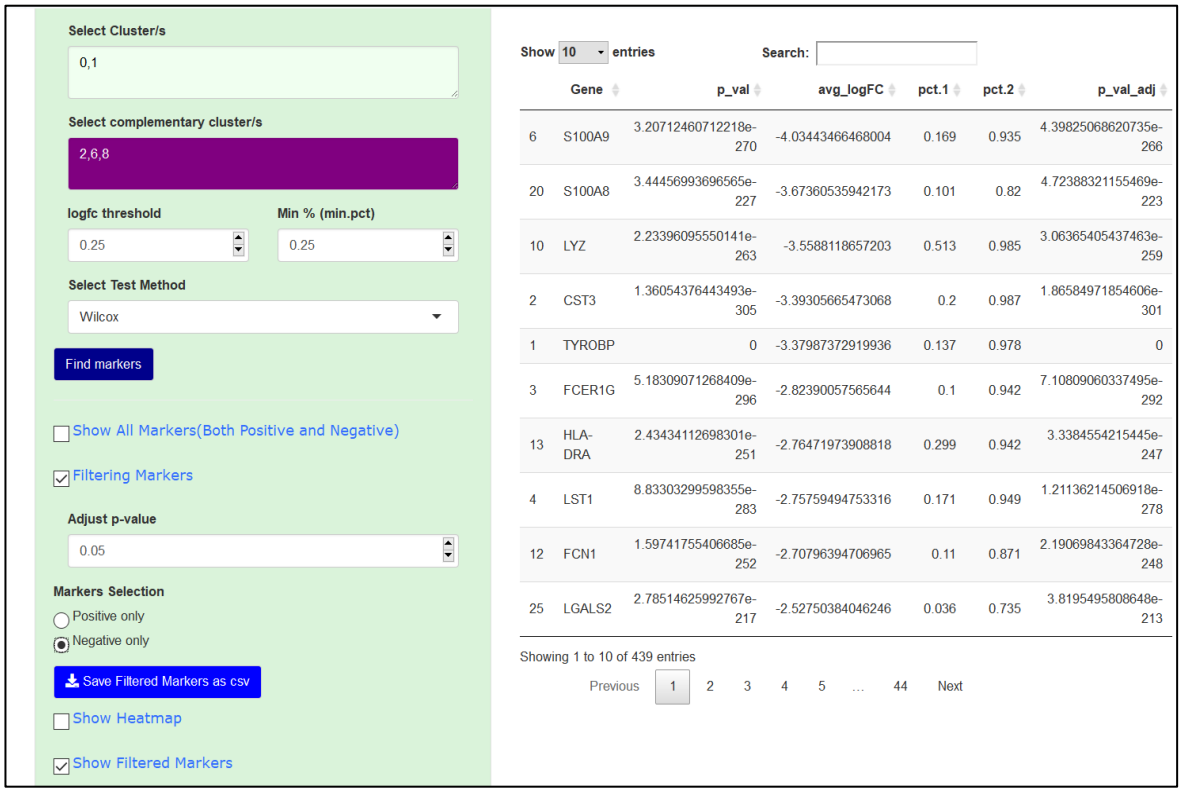


Figure 39. Differential Expression (Find markers by clusters vs other clusters): The table shows only down-regulated DE genes after filtering

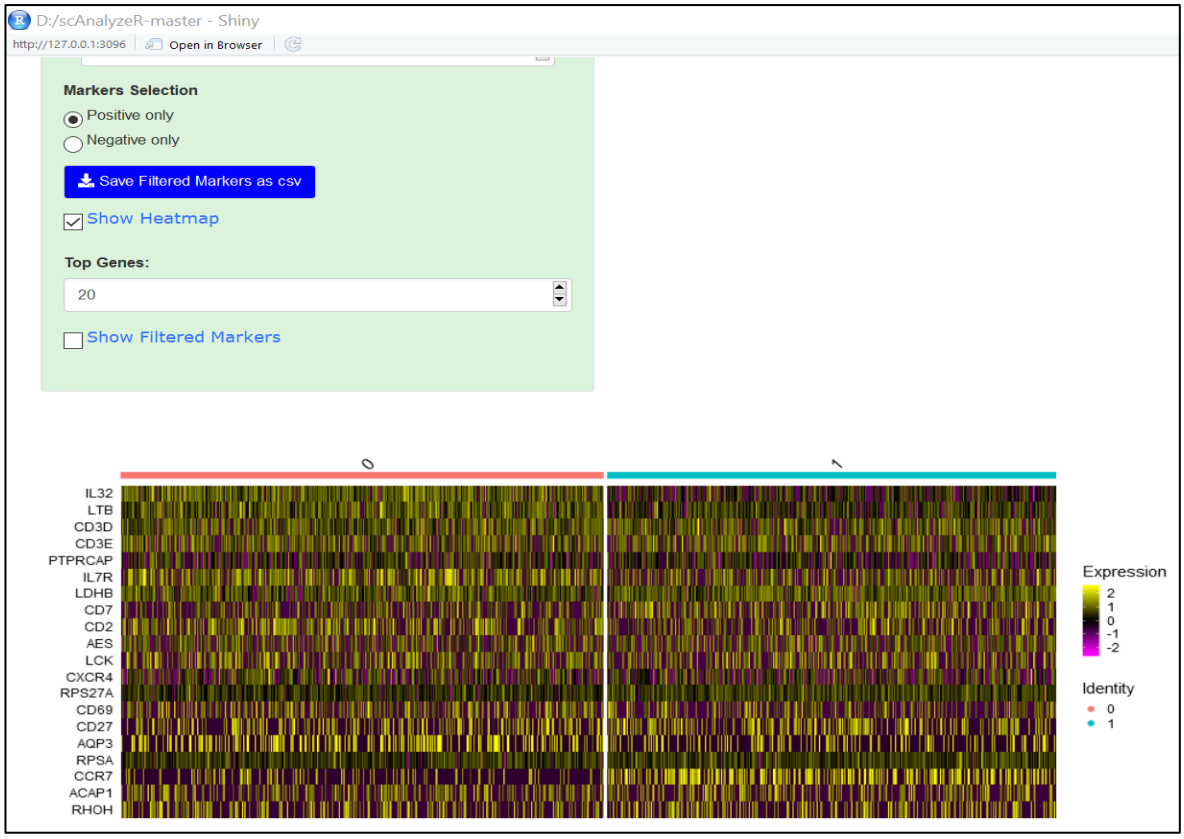


Figure 40. Differential Expression (Find markers by clusters vs other clusters): The heatmap illustrates expressions for top-twenty up-regulated DE genes after filtering

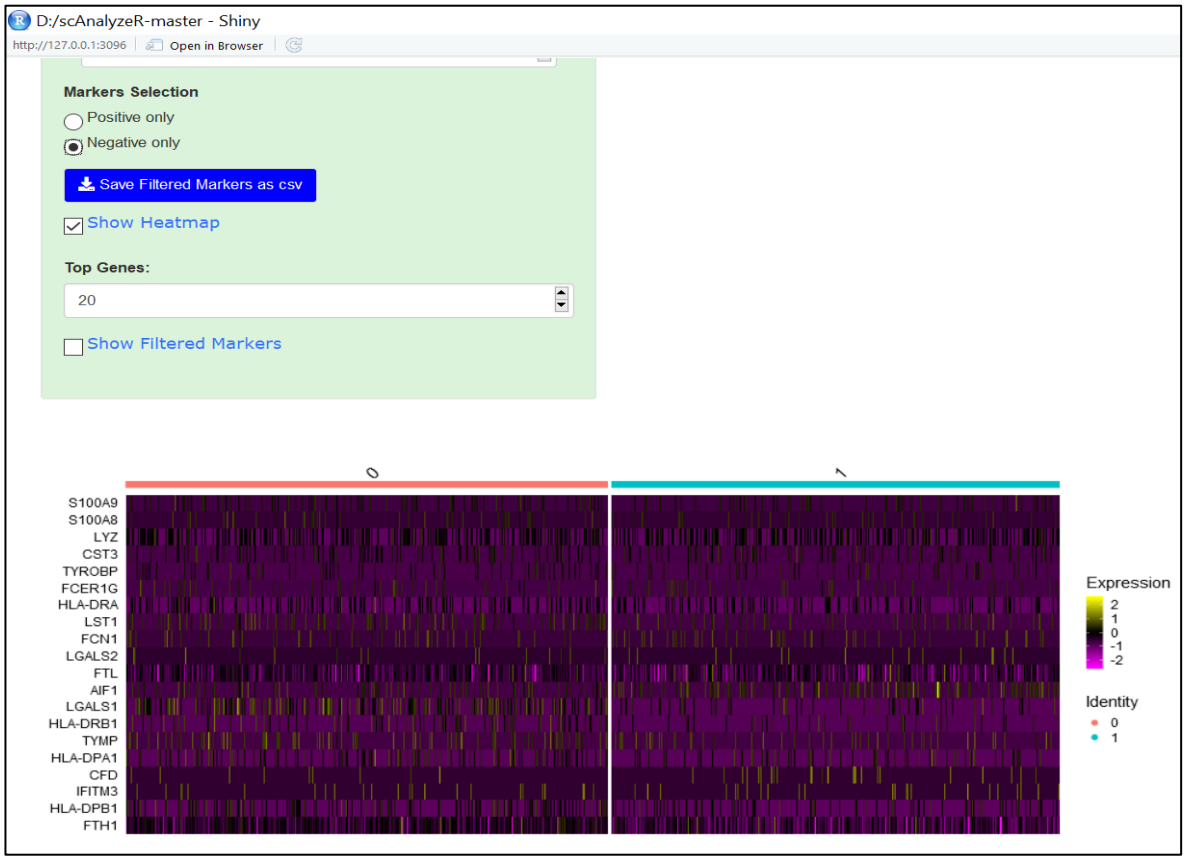


Figure 41. Differential Expression (Find markers by cluster): The heatmap illustrates expressions for top-twenty down-regulated DE genes after filtering

9. Plots

The ‘Plots’ module is divided into five essential plotting functions, i.e., Violin Plot, Feature Plot, Heatmap, and Correlation Plot. In the “Violin Plot” textbox, write one or more gene symbol(s) (one gene per line), then the violin plot will be shown automatically, which represents the gene expression across clusters (**Fig. 42-43**). The “Feature Plot” visualizes the gene expression on a tSNE plot (**Fig. 44**). The input style is the same as the “Violin Plot.”

There are two different input options for the “Heatmap,” you can either upload a text file that contains a set of gene symbol (one gene per line) or write gene symbols in the textbox that is provided on the interface, then the expression heatmap plot (by default Heatmap with clusters) will be shown for provided genes with all cells (after filtering, in pre-processing module) (**Fig. 45**). You may choose the “Heatmap without clusters” radio button that generates an expression heatmap for the same input genes without cluster indication (**Fig. 46**). To draw the dynamic PCA plot (**Fig. 47**), upload a set of genes in a text file (one gene symbol per line), then the PCA plot will be displayed. There are several advantages of the dynamic PCA plot, e.g., zooming a particular area (**Fig. 48-49**), zoom out, box and lasso selection, etc. The “Correlation Plot” submodule shows a correlation between two given genes. The ‘GAPDH’ and ‘TP53’ genes are provided, the ‘pearson’ correlation method and the linear regression line is drawn, by default setting. You can choose different gene names and by mouse clicking. If you check the “Apply Log2” checkbox, the log2 function is applied to the normalized expression data for given gene symbols, and then it draws the correlation plot between the two genes (**Fig. 50**).

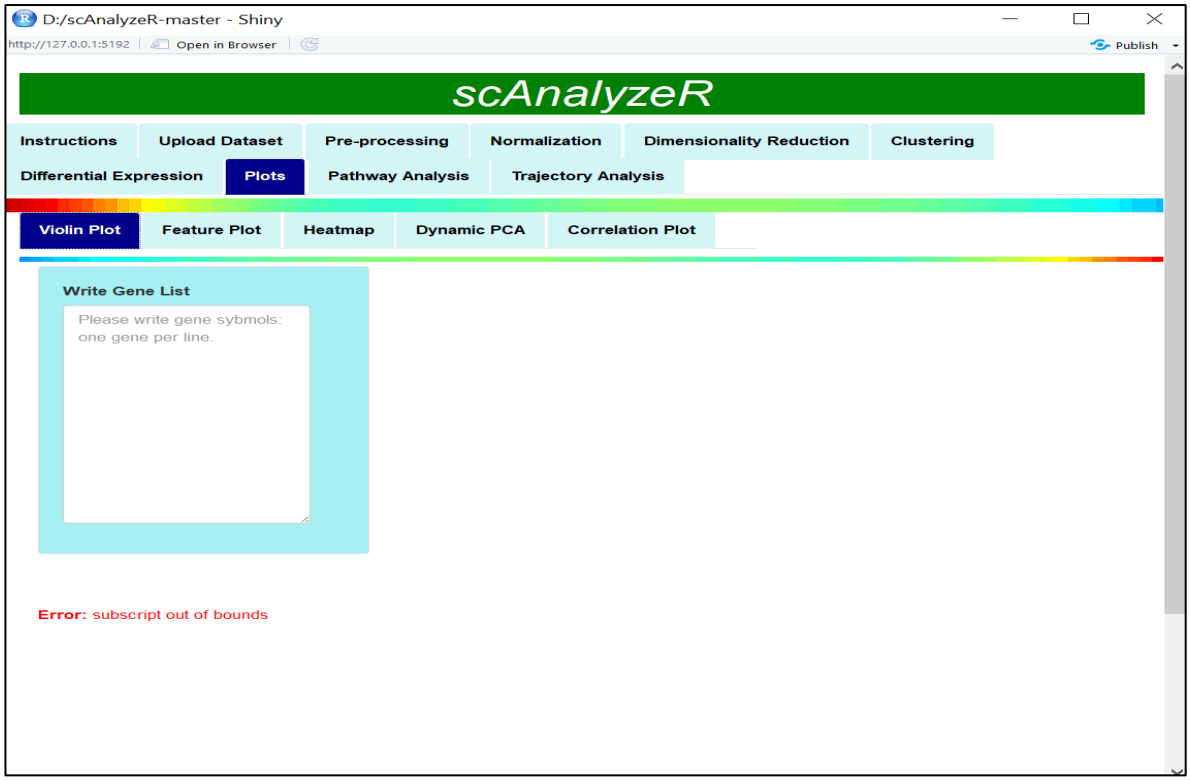


Figure 42. Violin plot: user interface

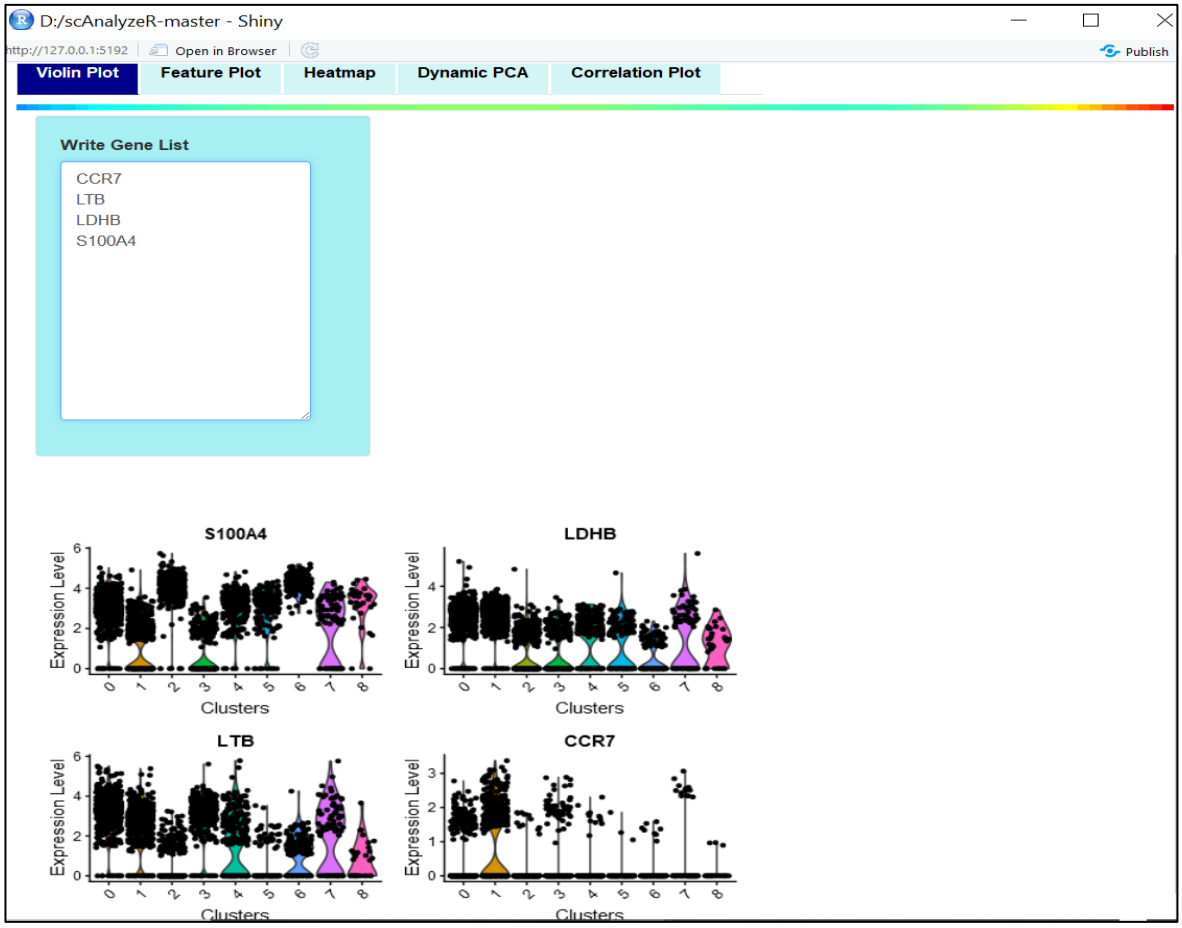


Figure 43. Violin plot for four individual genes

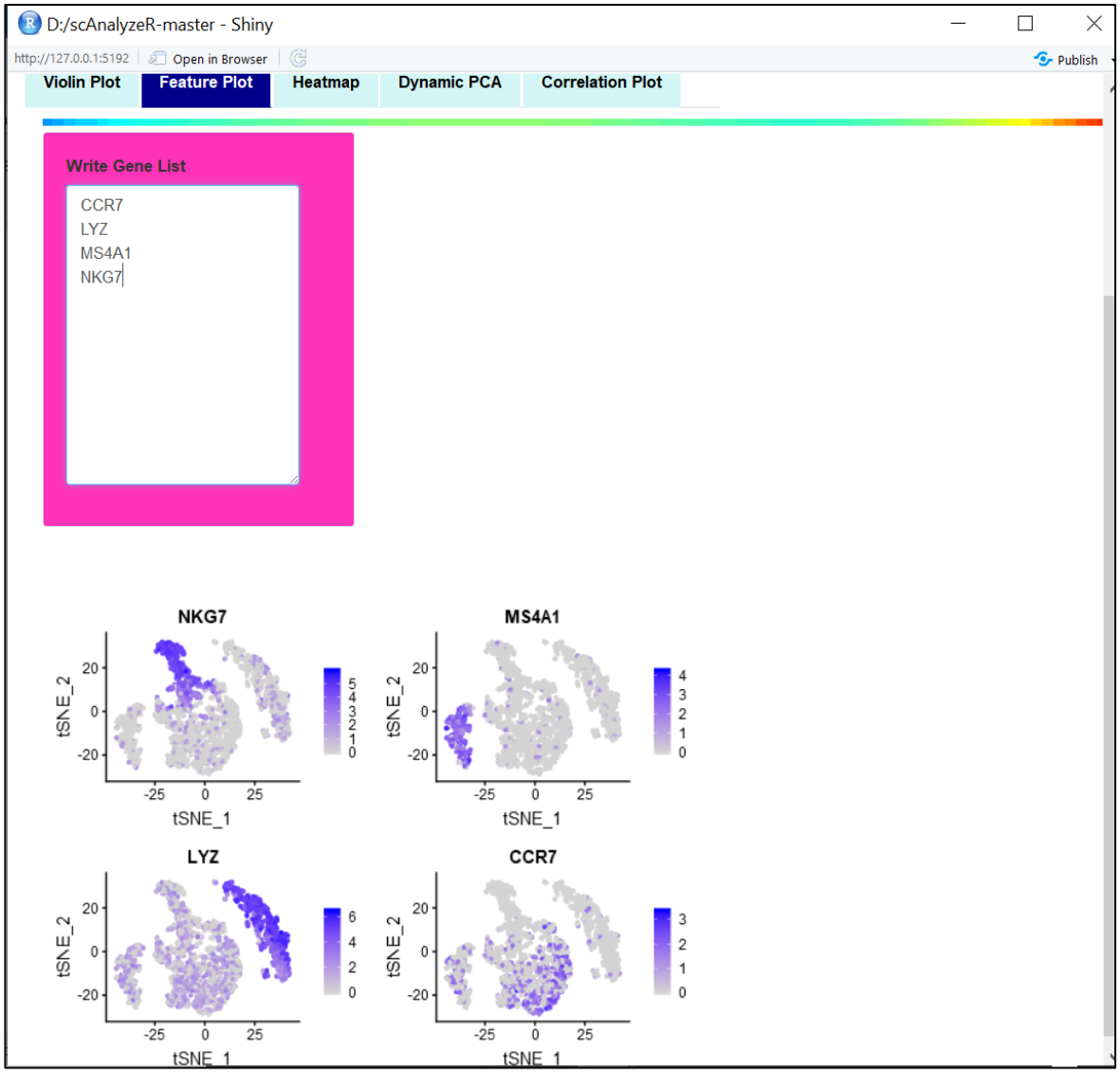


Figure 44. Feature plot for four individual genes

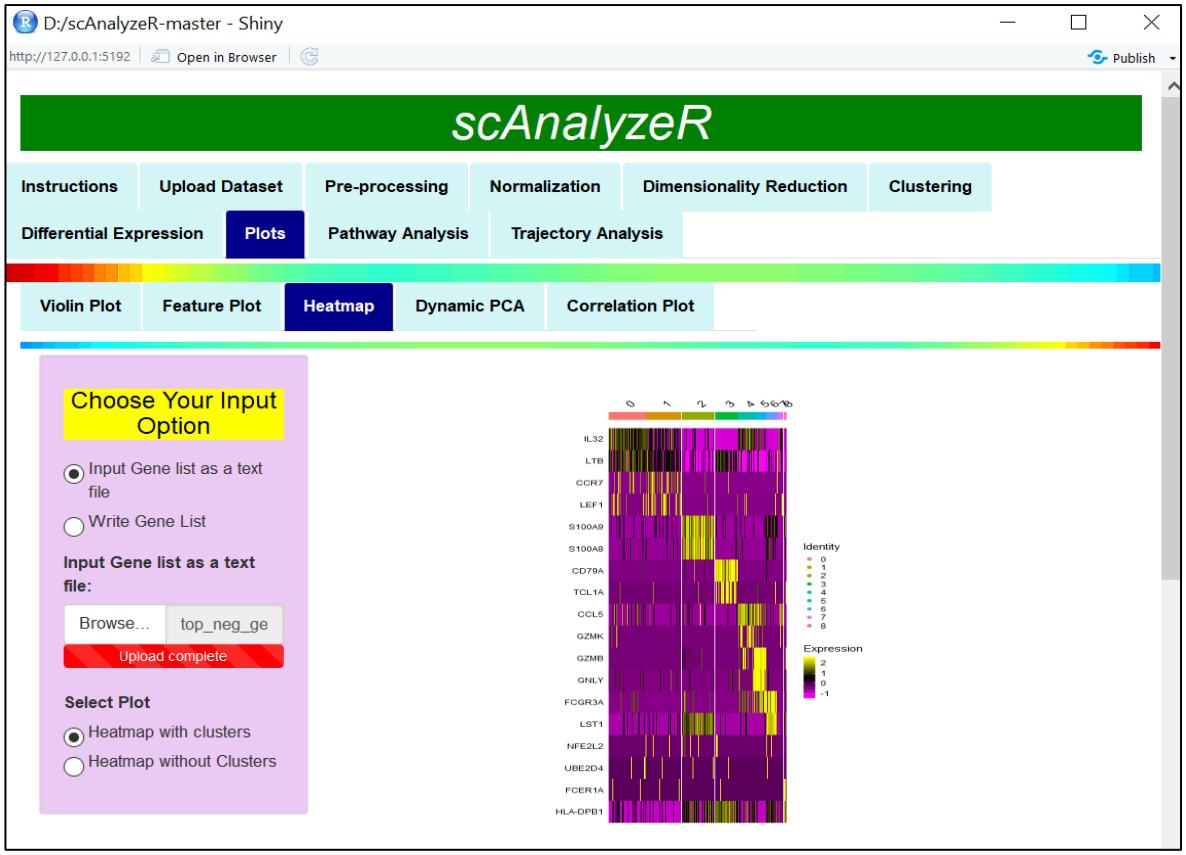


Figure 45. Heatmap plot for a set of genes(uploaded) with clusters indication

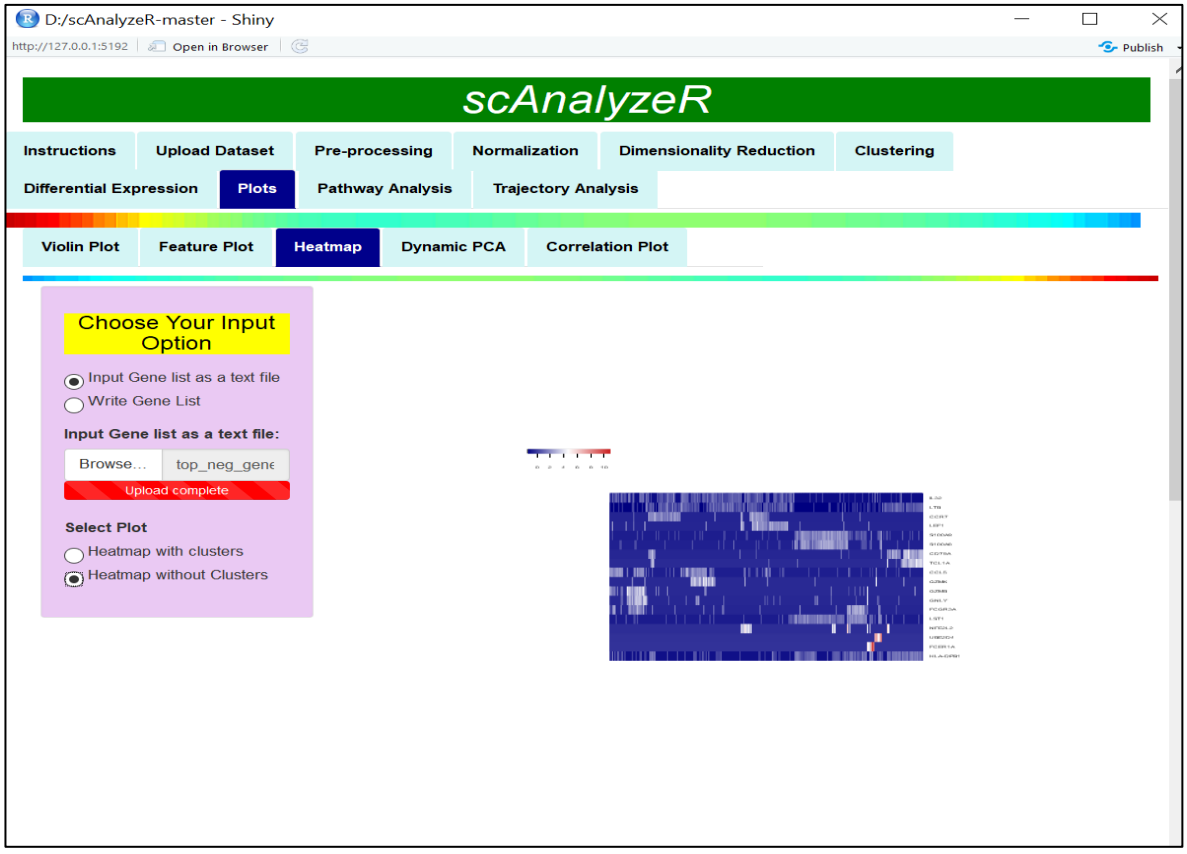


Figure 46. Heatmap plot for a set of genes(uploaded) without clusters indication

Figure 47. Dynamic PCA: PCA plot for a set of genes(uploaded) with dynamic customization options

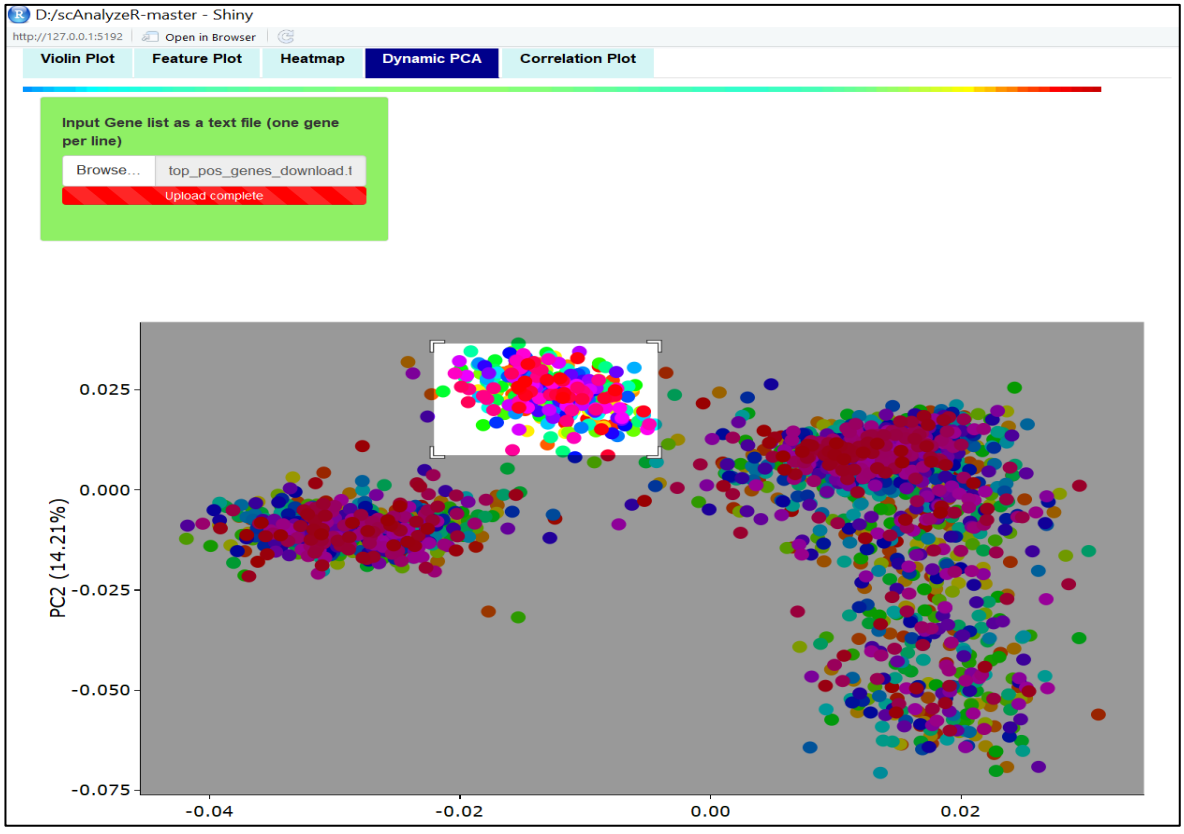


Figure 48. Dynamic PCA: Selected an area on the PCA plot for zooming cells(dots)

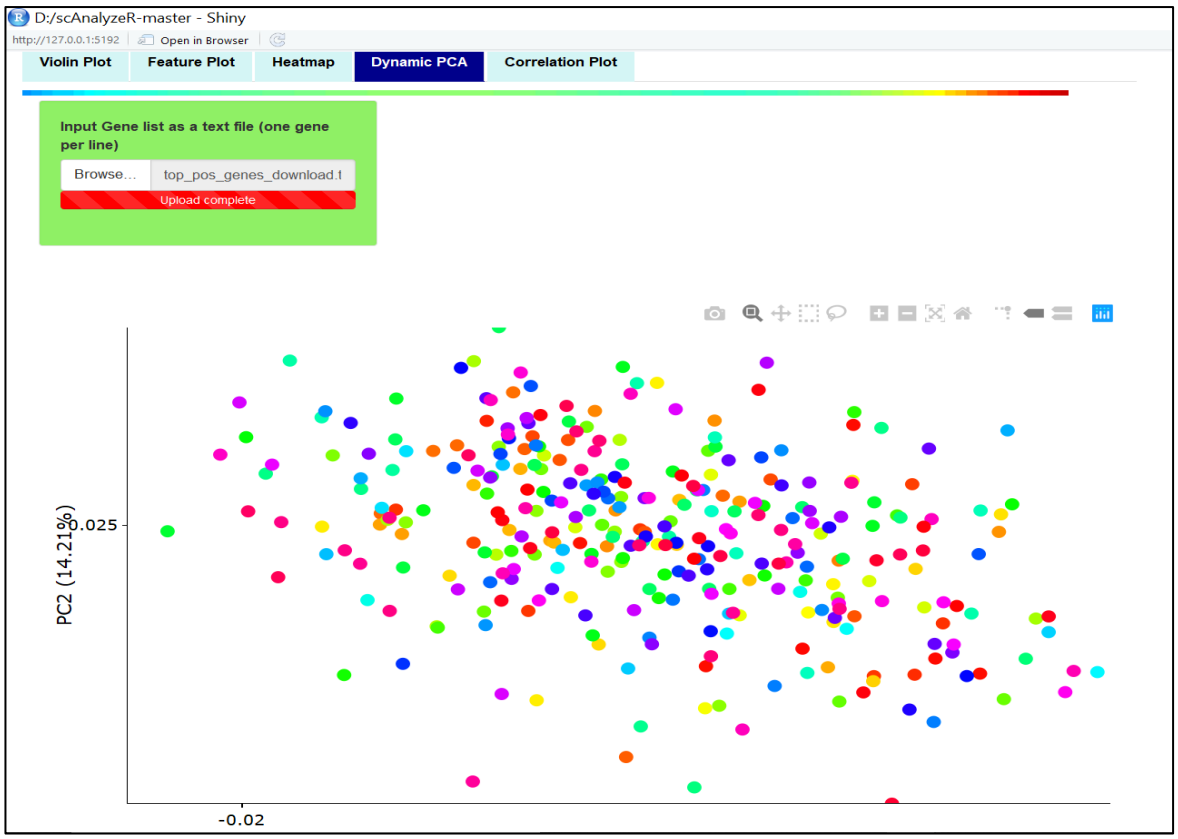


Figure 49. Dynamic PCA: PCA plot after zooming the selected area on Fig. 43

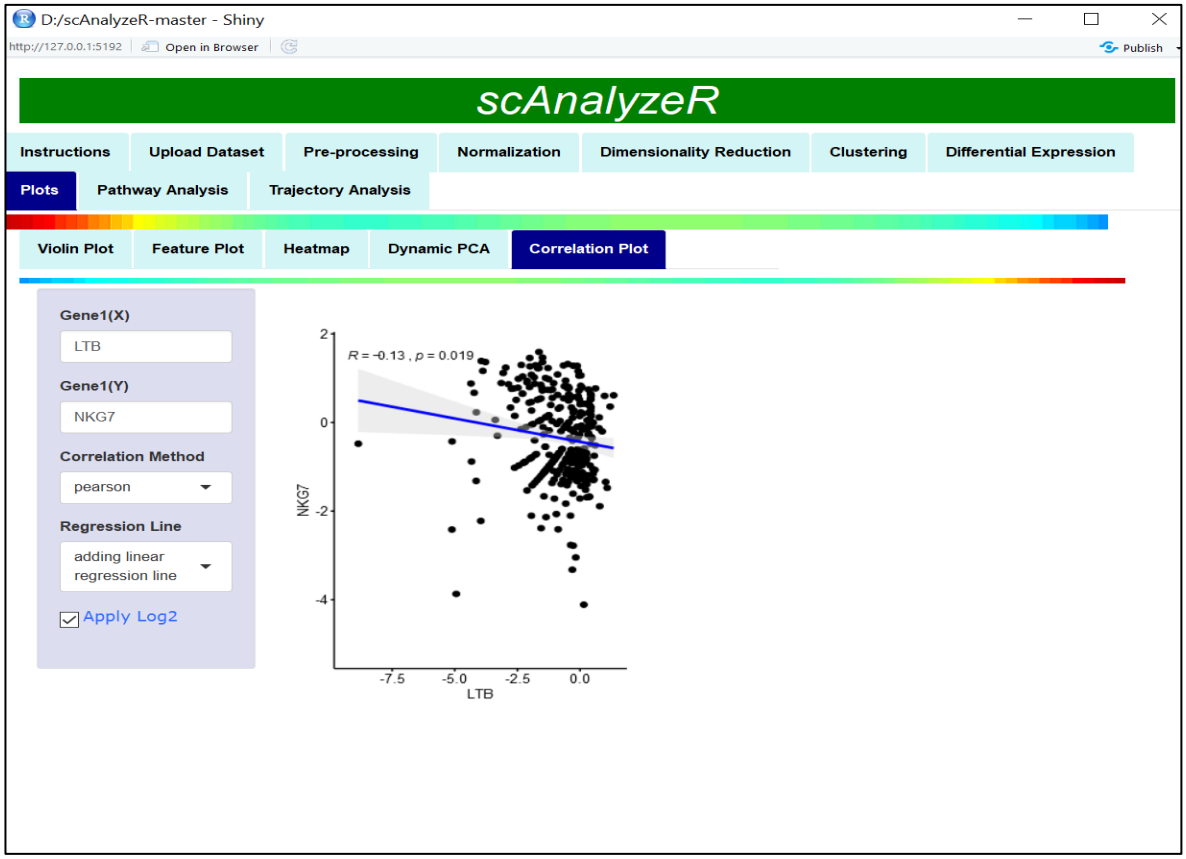


Figure 50. Correlation plot between two given genes

10. Pathway Analysis

The gene set enrichment analysis (GSEA) of previously (on 'Differential Expression' module) identified differentially expressed genes is performed in this analysis module. The "Cluster Specific DE Genes" is selected by default as an input gene list, and the 'KEGG pathways' is chosen as a source of pathway gene sets. You can either keep default selection or select different choices and click the "Compute Pathways" to proceed results. In our example, we choose default setting for computing pathways, i.e., 'KEGG pathways' and 'Cluster Specific DE Genes' (cluster 0's DE genes) were selected (**Fig. 51**). In the "Select Pathways" tab, if you choose the 'Positive Pathways,' then only positive pathways are selected for all later parts of results generation, and the reverse action will be applied if the 'Negative Pathways' is chosen. However, the full list (both positive and negative) of pathways and filtered (filtered by padj) list of pathways can be downloaded via clicking the "Download All Pathways as a csv" and "Download Filtered Pathways as a csv" buttons, respectively.

To show the pathway plot and pathway list for filtered pathways, click the "Show Pathways(plot)" and the "Show Pathways list" checkboxes, respectively. Moreover, gene symbols of gene sets (either significant genes or all genes) and their heatmap for a particular pathway (1 by default) can be seen by clicking the "Show Gene sets" checkbox (**Fig. 52-53**).

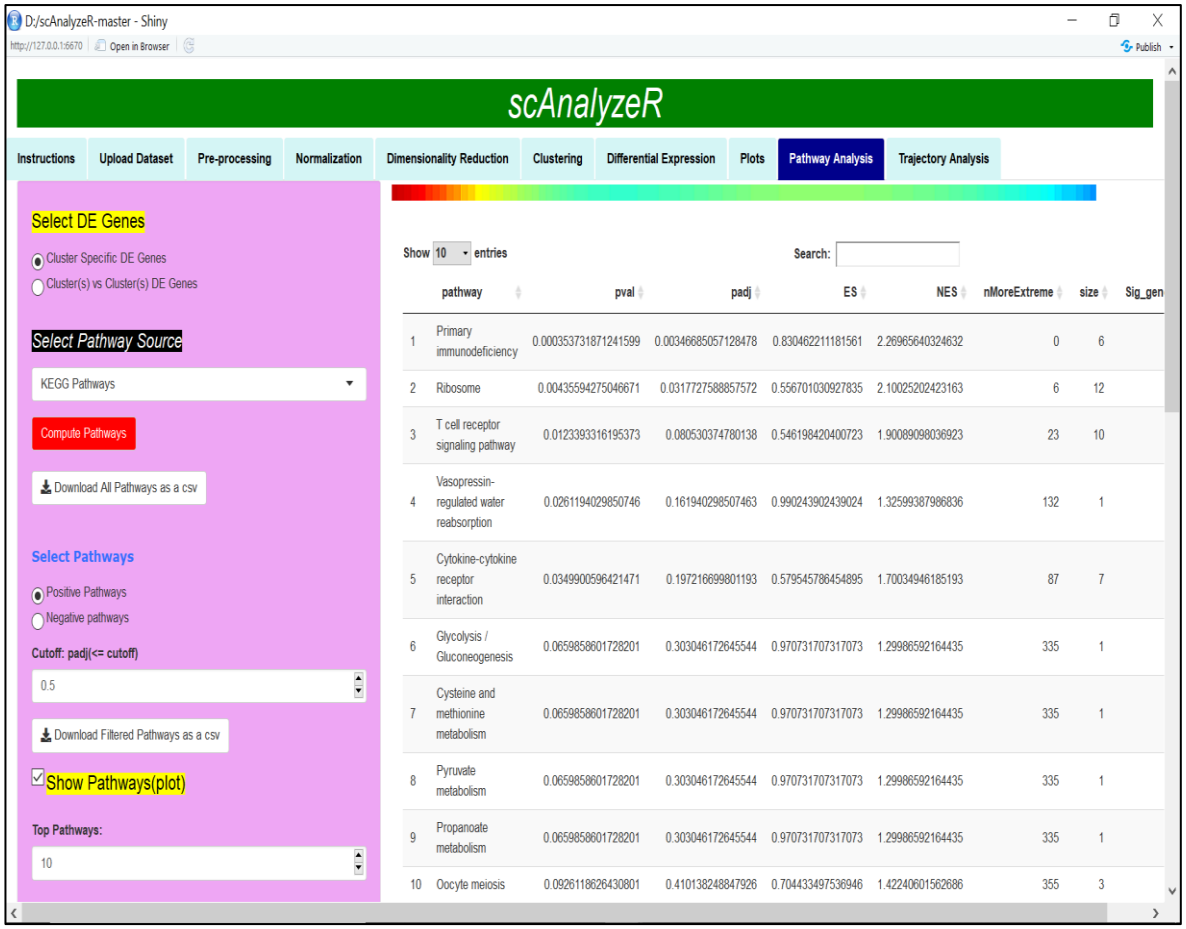


Figure 51. Pathway Analysis: The table shows the top-ten up-regulated pathways after filtering

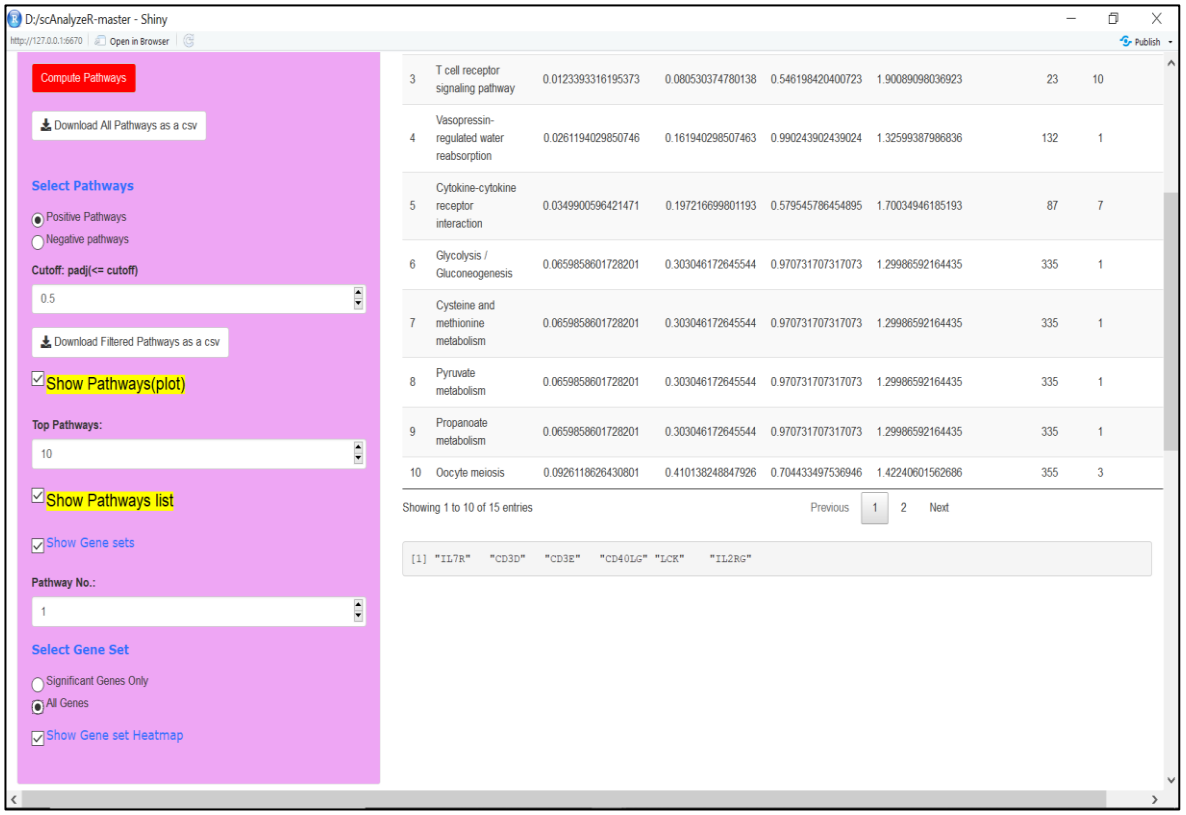


Figure 52. Pathway Analysis: Shows all gene list (bottom) in the top up-regulated pathway after filtering

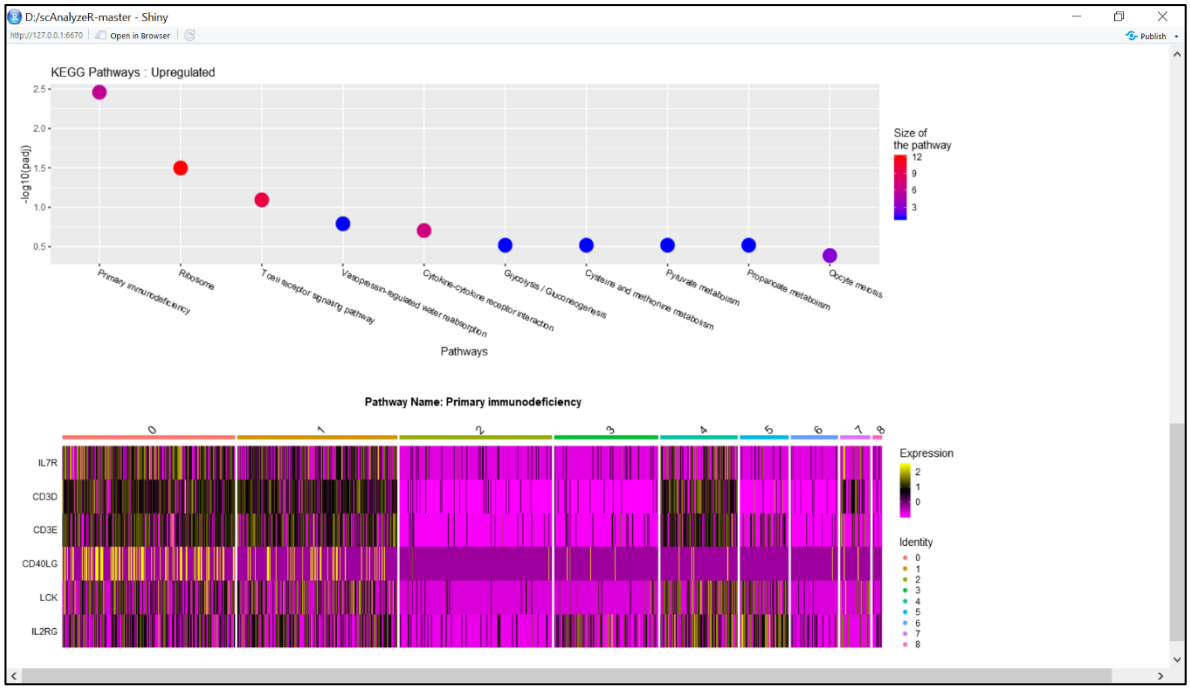


Figure 53. Pathway Analysis: Bubble plot shows the top-ten up-regulated pathways, and the heatmap illustrates expressions for the top up-regulated pathway's gene set (all genes)

11. Trajectory Analysis

In the last module, cells are ordered in pseudo-time according to their expression profiles such as cell differentiation stage. Select either 'All genes' or 'Highly Variable Genes Only' (default option) from the "Gene use" tab before clicking the "Cell trajectory plot," then cell trajectory plot will be shown with three different plot styles, 'Clusters' is selected by default. To see the pseudotemporal heatmap, click the "Pseudotemporal heatmap" checkbox, then the interface of the input panel will be visible. The input gene list can either be uploaded as a text file (one gene per line) or written in the input box, and can change the number of clusters or keep the default value before clicking the "OK" button. Once calculating is completed, the heatmap will be seen on the interface. In this example, the highly variable genes were used for both cell trajectory plot and pseudo-temporal heatmap (top 10 highly variable genes only) analysis (**Fig. 54-56**).

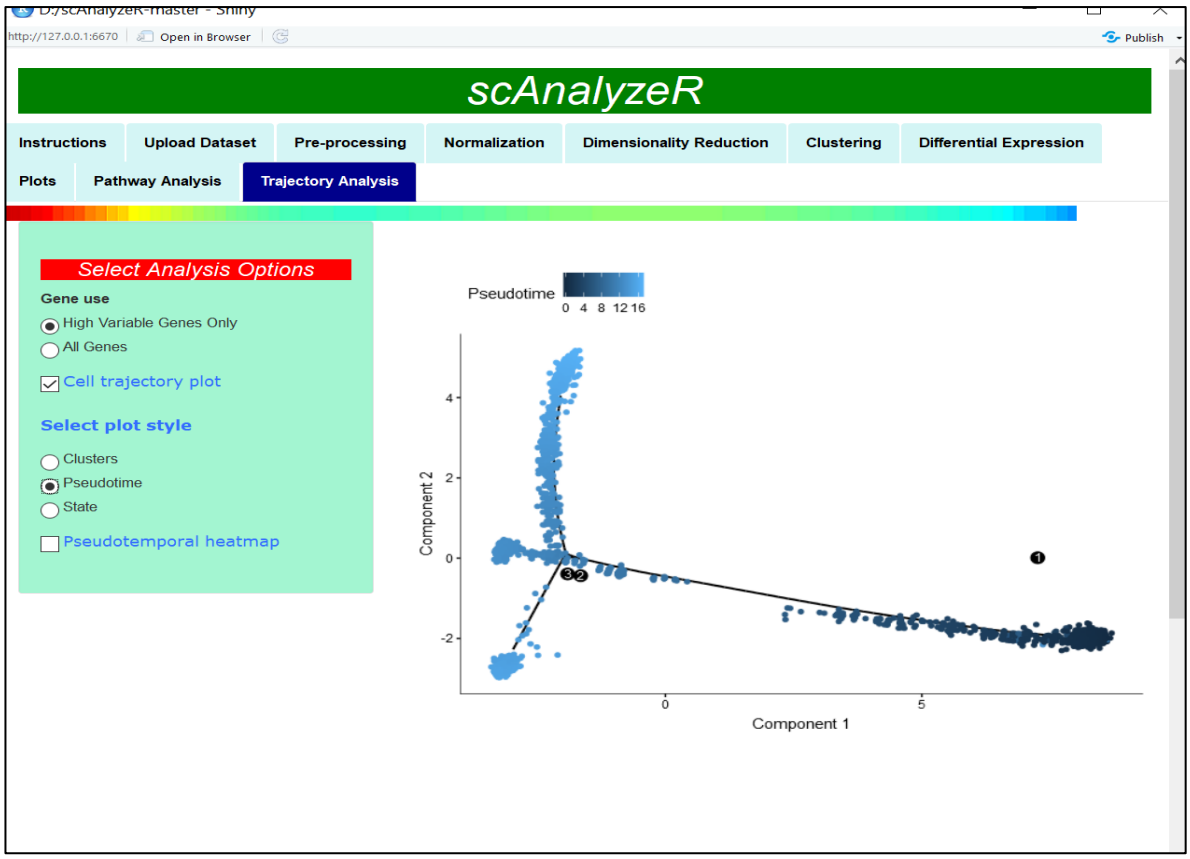


Figure 54. Trajectory Analysis: Cell trajectory plot with pseudotime

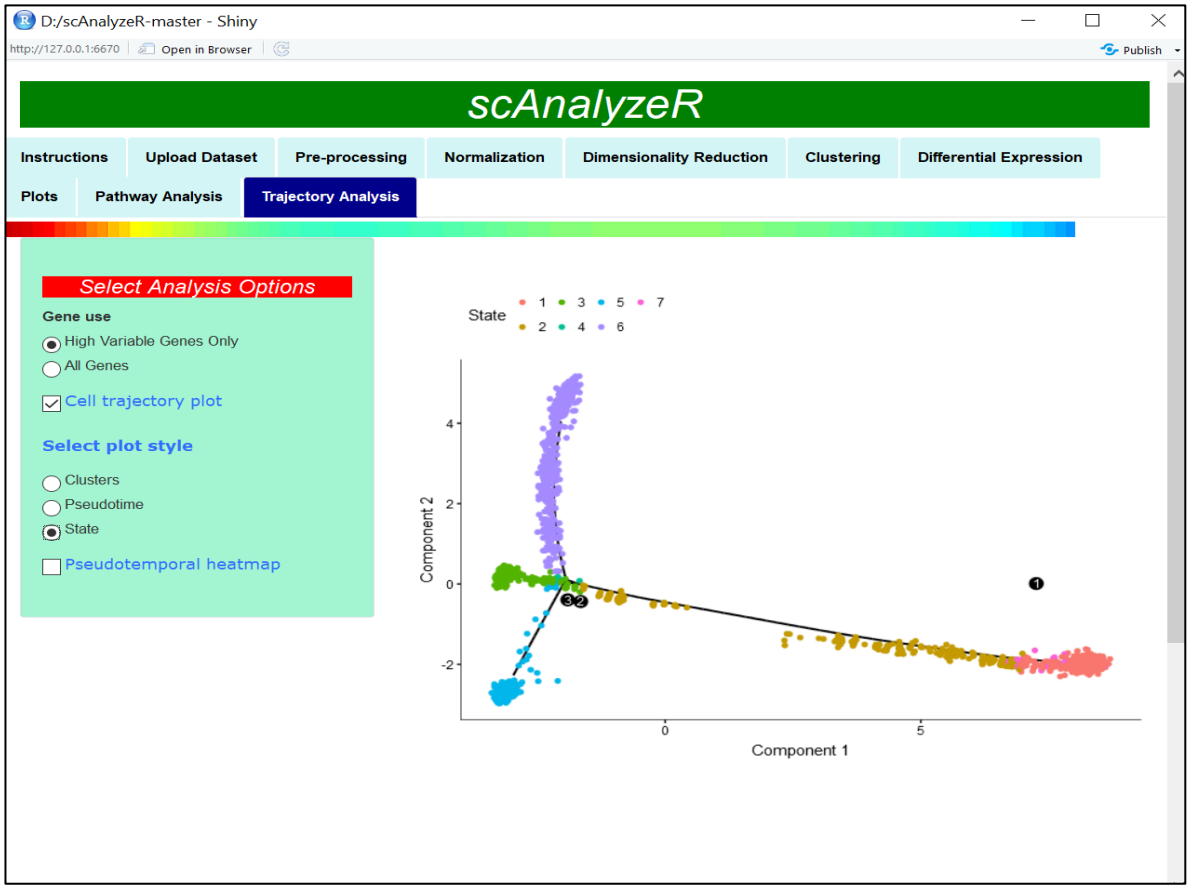


Figure 55. Trajectory Analysis: Cell trajectory plot with states

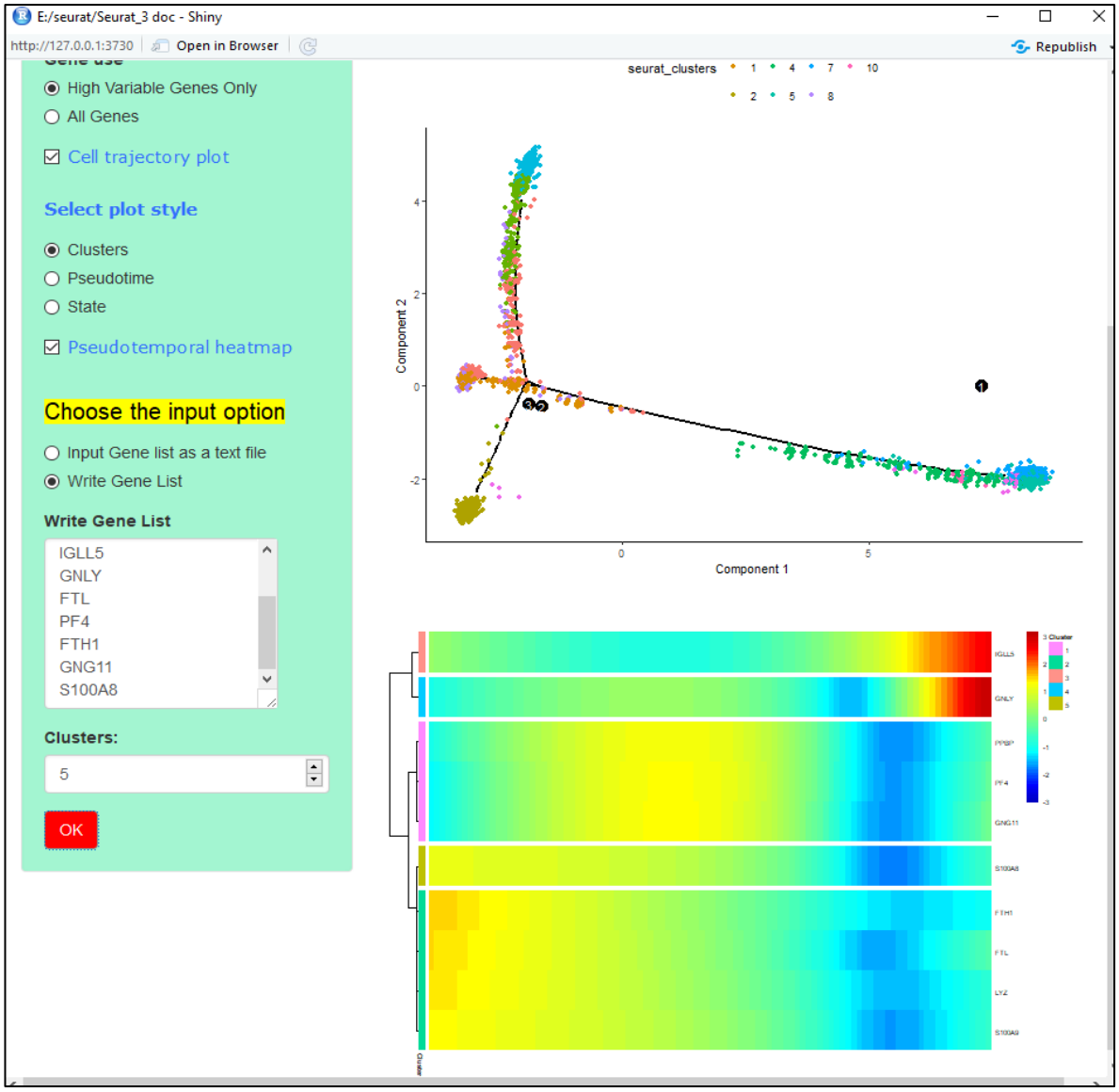


Figure 56. Trajectory Analysis: Cell trajectory plot with clusters(top), and pseudotemporal heatmap plot(bottom)

12. References

1. Nguyen, Q.H., et al., *Profiling human breast epithelial cells using single cell RNA sequencing identifies cell diversity.* Nat Commun, 2018. **9**(1): p. 2028.
2. Aizarani, N., et al., *A human liver cell atlas reveals heterogeneity and epithelial progenitors.* Nature, 2019. **572**(7768): p. 199-204.
3. Zheng, H., et al., *Single-cell analysis reveals cancer stem cell heterogeneity in hepatocellular carcinoma.* Hepatology, 2018. **68**(1): p. 127-140.
4. Su, X., et al., *Single-cell RNA-Seq analysis reveals dynamic trajectories during mouse liver development.* BMC Genomics, 2017. **18**(1): p. 946.
5. Rizvi, A.H., et al., *Single-cell topological RNA-seq analysis reveals insights into cellular differentiation and development.* Nat Biotechnol, 2017. **35**(6): p. 551-560.
6. Vuong, N.H., et al., *Single-cell RNA-sequencing reveals transcriptional dynamics of estrogen-induced dysplasia in the ovarian surface epithelium.* PLoS Genet, 2018. **14**(11): p. e1007788.