Introduction

GeneCloudOmics is a web server for transcriptome data analysis and visualization. It supports the analysis of microarray and RNASeq data and performs ten different bio-statistical analyses that cover the common analytics for gene expression data. Furthermore, it gives the users access to several bioinformatics tools to perform 12 different bioinformatics analyses on gene/protein datasets.

GeneCloudOmics is designed as a one-stop server that helps the users perform all tasks through an intuitive graphical user interface (GUI) that waves the hassle of coding, installing tools, packages or libraries and dealing with operating systems compatibility and versioning issues, some of the complications that make data analysis tasks more challenging for biologists. GeneCloudOmics is an open-source tool and the website is free and open to all users and there is no login requirement.

Section I - Data upload, import and pre-processing

GeneCloudOmics supports two types of common data formats for gene expression analysis: RNA-Seq count matrix and Microarray CEL files.

1. RNA-Seq count matrix format

1.1. Data Upload

GeneCloudOmics requires all input files in comma-separated value (.csv) format. The data file in .csv should contain the gene names in rows and genotypes (e.g. wild type – mutants or control - treatments, …) in columns, which is similar to the standard transcriptome data file format of the NCBI GEO database. Supporting files (if applicable) include gene length, list of negative control genes, and metadata file.

1.2. Data import from NCBI GEO databases

GeneCloudOmics supports the import of transcriptomic data directly from GEO databases. The user is required to provide the GEO accession of the desired dataset and GeneCloudOmics will import the data. The user can specify the type of the transcriptomic data (RNA-Seq or microarray) or leave it to GeneCloudOmics.

Figure 1. The data upload and import menu of GeneCloudOmics

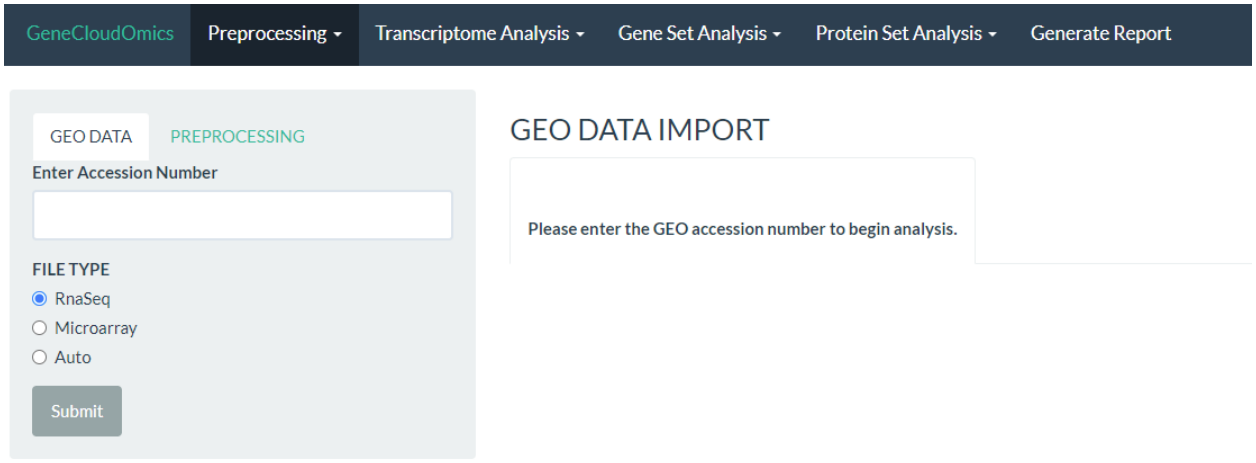


Figure 2. The GEO data import menu of GeneCloudOmics.

Once the data is imported successfully from GEO, the user is asked to select the data file(s) to be included in the analysis then move on to the **Preprocessing** menu.

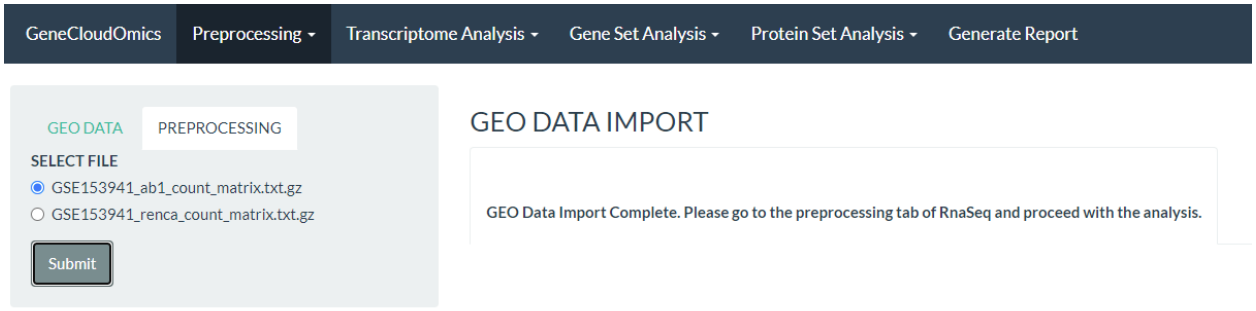
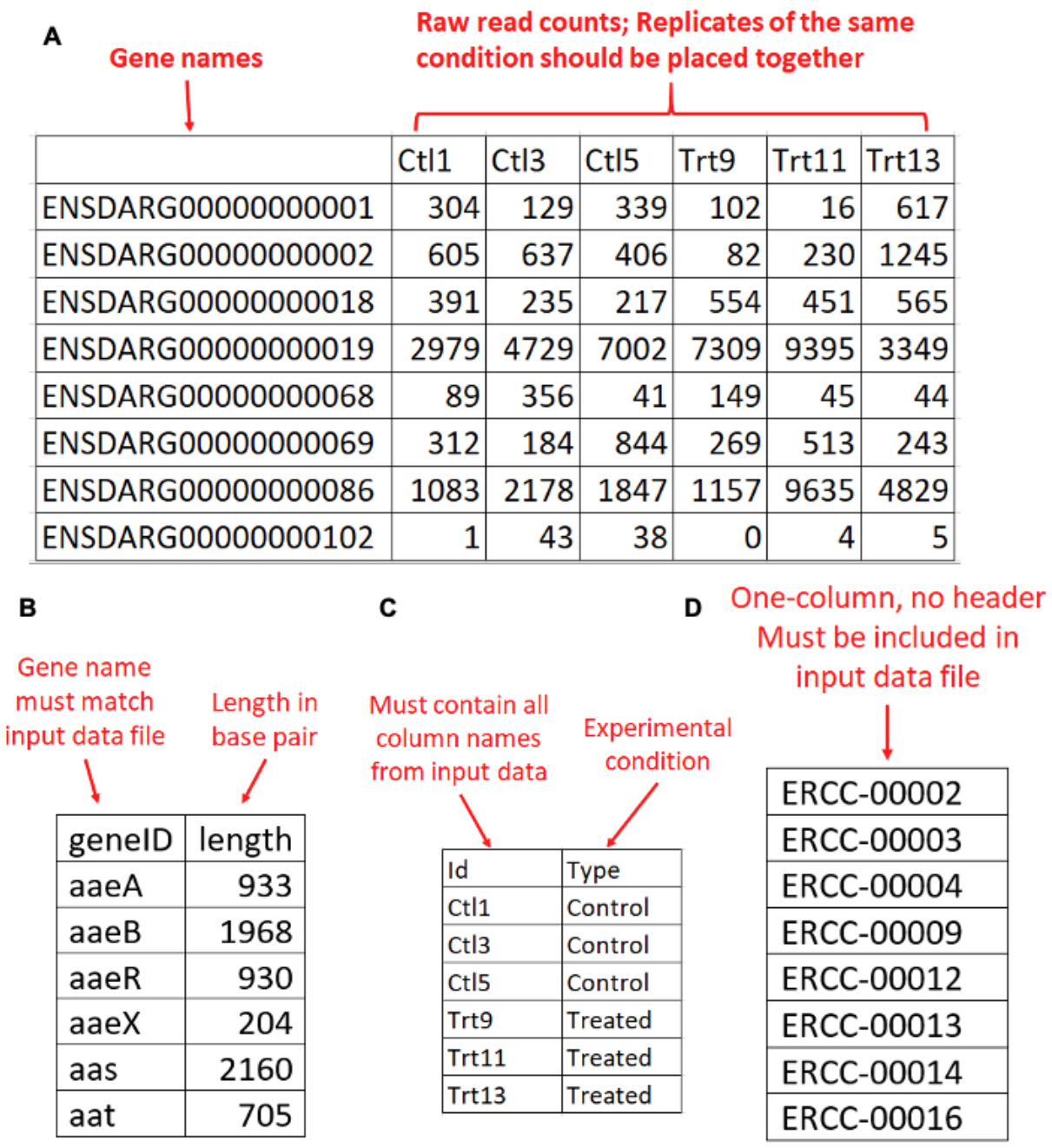


Figure 3. The user chooses from the successfully imported GEO data files to preprocess and perform further analysis.

Once data files and supporting files are loaded into GeneCloudOmics, the user can press **Submit** and GeneCloudOmics will automatically proceed to the next tab, the **Preprocessing**. In case the user accidentally uploaded a wrong data file or a wrong supporting file, the user can overwrite each of them by uploading new files or refresh the page to reset the whole program.

1.3. Normalization

A number of normalization options are provided in the preprocessing tab depending on the availability of supporting files: RPKM, FPKM, TPM (requiring gene length), RUV (requiring negative control genes), and Upper Quartile (no supporting file needed). The metadata file is required for differential expression analysis, and should specify experimental conditions (e.g. Control/Treated, time 1/time 2/ time3,...) for each genotype listed in the data file. Otherwise, if the data is normalized, the user can move to the next option to perform biostatistical analyses (such as scatter plot, distribution fitting, Pearson correlation) or differential expression (DE) analysis.



	Ctl1	Ctl3	Ctl5	Trt9	Trt11	Trt13
ENSDARG00000000001	304	129	339	102	16	617
ENSDARG00000000002	605	637	406	82	230	1245
ENSDARG00000000018	391	235	217	554	451	565
ENSDARG00000000019	2979	4729	7002	7309	9395	3349
ENSDARG00000000068	89	356	41	149	45	44
ENSDARG00000000069	312	184	844	269	513	243
ENSDARG00000000086	1083	2178	1847	1157	9635	4829
ENSDARG00000000102	1	43	38	0	4	5

geneID	length
aaeA	933
aaeB	1968
aaeR	930
aaeX	204
aas	2160
aat	705

Id	Type
Ctl1	Control
Ctl3	Control
Ctl5	Control
Trt9	Treated
Trt11	Treated
Trt13	Treated

ERCC-00002
ERCC-00003
ERCC-00004
ERCC-00009
ERCC-00012
ERCC-00013
ERCC-00014
ERCC-00016

Figure 4. The required format for A) raw counts data file, B) gene length file, C) metadata file and D) Negative control gene file

1.4. Pre-processing

Preprocessing involves two steps: 1) removing lowly expressed genes and 2) normalizing the remaining gene expression. First, the user needs to specify threshold expression values (which must be in the same units as the input data file - either raw read counts or normalized

expression), and the minimum number of data columns that must exceed the threshold value. Normalization methods are available upon the availability of supporting data files: normalization for sequencing depth, including TPM and RPKM, requires gene length and normalization for sample variation, including RUV, requires negative control genes. Users can download the filtered, normalized data in the **Data** tab.

Relative Log Expression (RLE) plots of raw and processed data are displayed to visualize the effects of normalization. The distribution of gene expression in each data column is visualized by a violin plot.

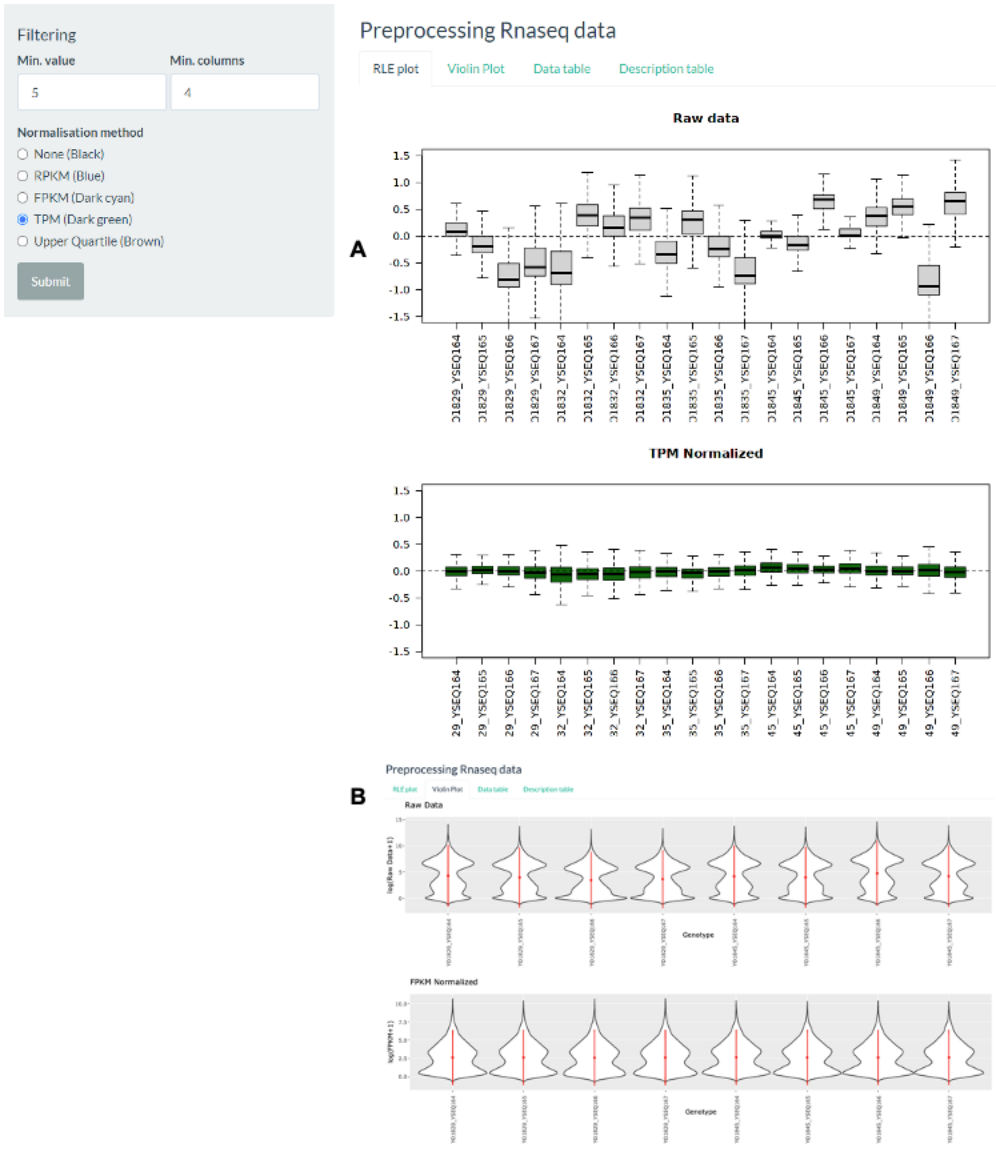


Figure 5. Data normalization panel with A) RLE plots of raw data (upper figure) and filtered, RUV-normalized data (lower figure). Gene expressions with a minimum of five counts in at least two columns are retained and B) the violin plots of raw and normalized data.

2. Micro-array CEL format

   For the DNA microarray data, GeneCloudOmics accepts files in CEL format. All .CEL files can be compressed into one compressed file and uploaded through the Microarray Data option in the **Preprocessing** menu.

Test data can be downloaded from
https://www.ebi.ac.uk/arrayexpress/experiments/E-MTAB-2967/

Section II - Transcriptome analysis

1. Scatter plot

A Scatter plot compares any two samples (or 2 replicates) by displaying the respective expression of all genes in 2D space. It is recommended to perform normalization for sequencing depth (TPM, RPKM, FPKM) for this step (and so does distribution fitting, correlation, hierarchical clustering, noise and entropy).

As gene expression data is naturally skewed towards very high expression level regions, we recommend applying log-transformation to capture the whole data range. Users can choose between log base 2, natural log, or log base 10. An option to add a linear regression line is also available.

Gene expression data is densely distributed in the lowly expressed region, making the dots usually indistinguishable in a regular scatter plot. GeneCloudOmics overlay a 2D kernel density estimation on the scatter plot to visualize the density of expression level.

The user can choose to download each single scatter plot, to download all pairs of samples scatter plot in one PDF file or to add them to the analysis report by clicking the **Add to Report** button. which may take some time to run.

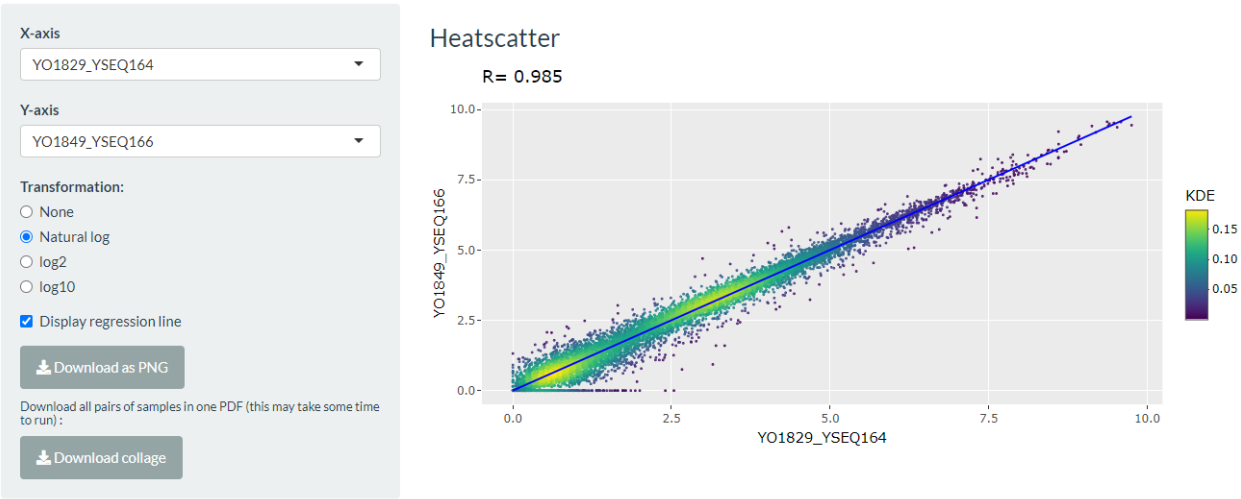


Figure 6. The Scatter plot

2. Distribution fitting

Distribution fitting compares the gene expression to a number of statistical continuous distributions, which can be used to validate the data. To visualize the comparison, GeneCloudOmics displays the Cumulative Distribution Function of the preprocessed gene expression data with the user-selected theoretical distributions. Once it is confirmed that the gene set follows a distribution, it would be safe to conclude the validity of the gene expression data. **A table** is also provided in the AIC table tab to show the best-fitted distribution in each sample

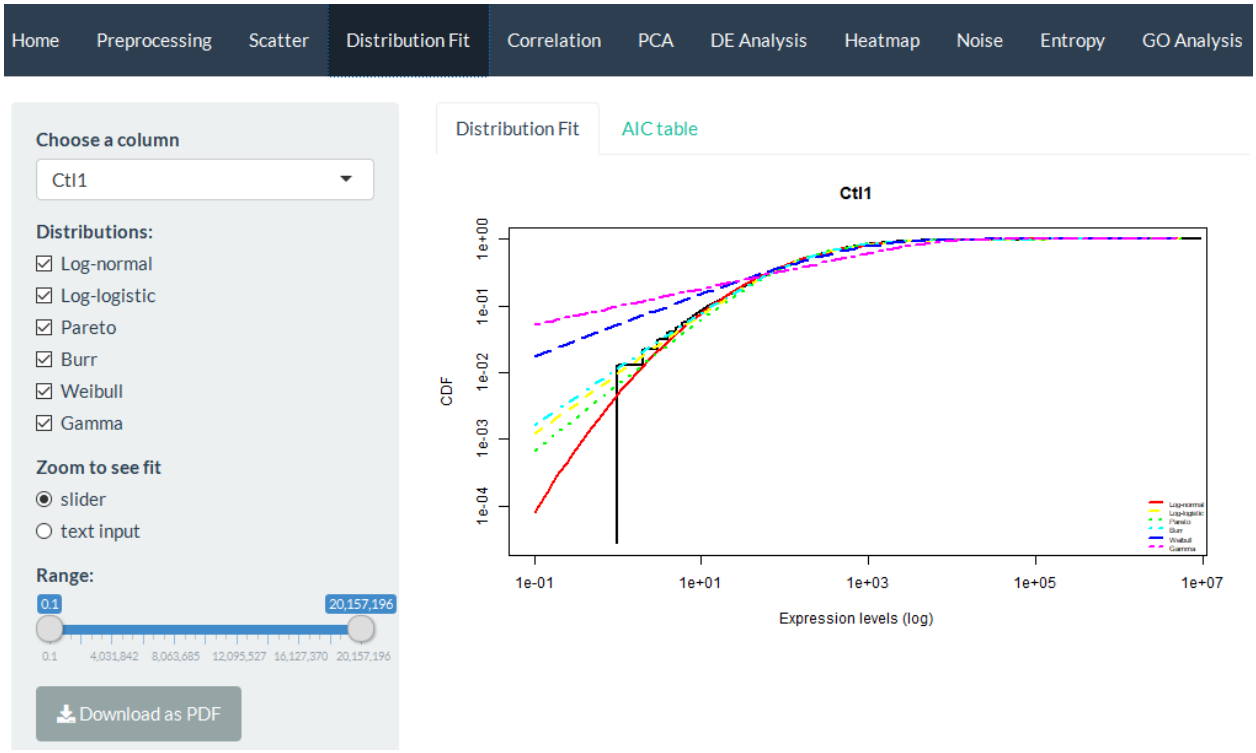


Figure 7. Comparing Cumulative Distribution Functions of raw count data (black) with lognormal (red colour), log-logistic (yellow colour), Pareto (green colour), Burr (cyan colour), Weibull (blue colour) and gamma (purple colour) distributions in Ctrl 1 replicate.

3. Correlation

2.1. Pearson correlation

The Pearson correlation coefficient r between two vectors (e.g. transcriptome in two different samples), containing n observations (e.g. gene expression values), is defined by (for large n):

$r(X,Y)=\sum ni=1(xi-\mu X)(yi-\mu Y)/\sigma X\sigma Y$

where xi and yi are the ith observation in the vectors X and Y, respectively, μX and μY, the average values of each vector, and σx and σy, the corresponding standard deviations. Pearson correlation measures the linear relationship between two vectors, where r = 1 if the two vectors are identical, and r = 0 if there are no linear relationships between the vectors.

2.2. Spearman correlation

Spearman rank correlation is a non-parametric test that is used to measure the degree of association between two vectors (e.g. transcriptome in two different samples). The Spearman rank correlation test does not carry any assumptions about the distribution of the data and is the appropriate correlation analysis when the variables are measured on a scale that is at least ordinal.

The following formula is used to calculate the Spearman rank correlation:

$$\rho(X,Y) = 1 - \frac{6\sum_{i=1}^{n}(r_{x,i} - r_{y,i})^2}{n(n^2-1)}$$

ρ= Spearman rank correlation

r_x,i , r_y,i = the ranks of the ith gene xi and yi, in vectors X and Y respectively.

n= number of genes in each gene expression vector (X, Y)

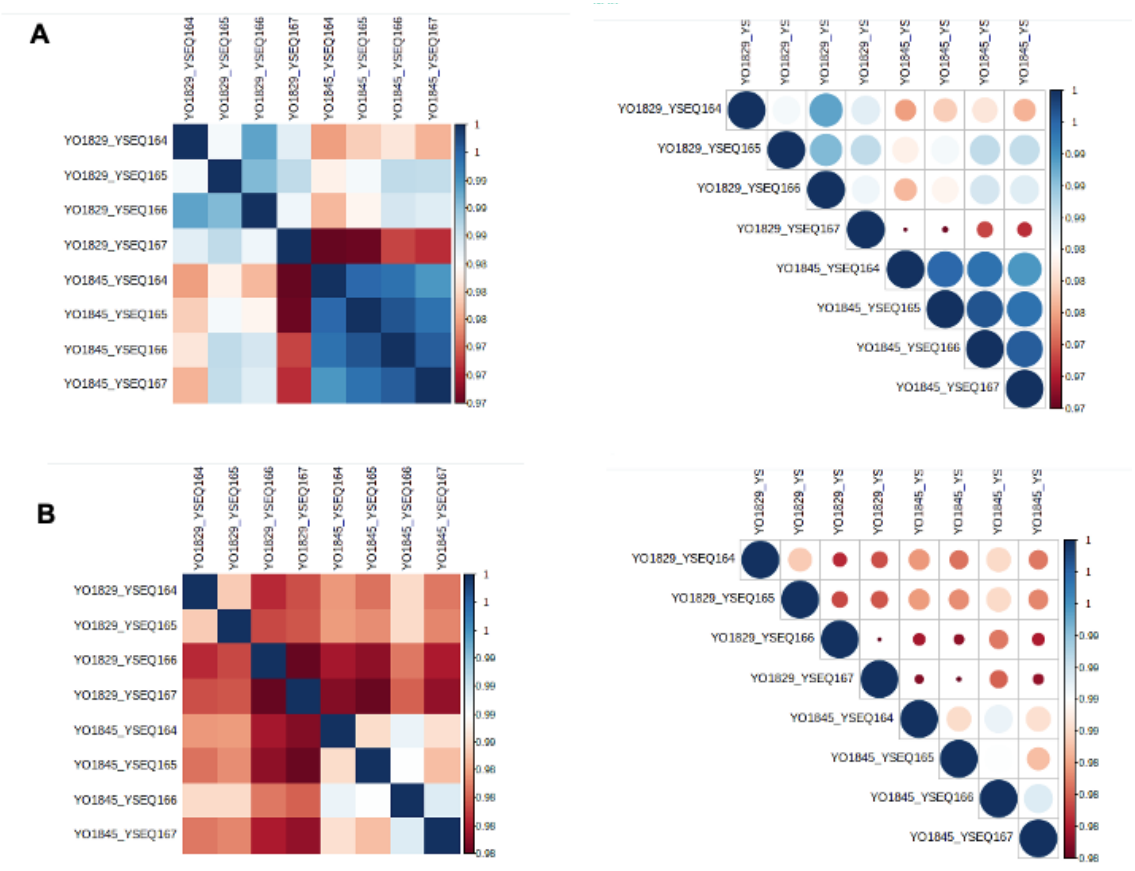


Figure 8. Correlation analysis in GeneCloudOmics. A) heatmap and correlation plot of Person correlation and B) heatmap and correlation plot of Spearman correlation.

4. PCA

Principal Components Analysis (PCA) is a multivariate statistical technique for simplifying high-dimensional data sets (Basilevsky 1994). Given *m* observations on *n* variables, the goal of PCA is to reduce the dimensionality of the data matrix by finding *r* new variables, where *r* is less than *n*. Termed principal components, these *r* new variables together account for as much of the variance in the original *n* variables as possible while remaining mutually uncorrelated and orthogonal. Each principal component is a linear combination of the original variables, and so it is often possible to ascribe meaning to what the components represent.

A PCA analysis of transcriptomic data considers the genes as variables, creating a set of "principal gene components" that indicate the features of genes that best explain the experimental responses they produce.

To compute the principal components, the *n* eigenvalues and their corresponding eigenvectors are calculated from the *n*×*n* covariance matrix of conditions. Each eigenvector defines a principal component. A component can be viewed as a weighted sum of the conditions, where the coefficients of the eigenvectors are the weights. The projection of gene *i* along the axis defined by the *j*th principal component is:

$$a'_{ij} = \sum_{t=1}^{n} a_{it} v_{tj}$$

Where *vtj* is the *t*th coefficient for the *j*th principal component; *a*it is the expression measurement for gene *i* under the *t*th condition. *A′* is the data in terms of principal components. Since *V* is an orthonormal matrix, *A′* is a rotation of the data from the original space of observations to a new space with principal component axes.

The variance accounted for by each of the components is its associated eigenvalue; it is the variance of a component over all genes. Consequently, the eigenvectors with large eigenvalues are the ones that contain most of the information; eigenvectors with small eigenvalues are uninformative.

Figure 9. PCA analysis in GeneCloudOmics. A) 2D PCA with K means clustering and sample names displayed and B) 3D PCA with K means clustering and sample names displayed.

5. Differential expression analysis

DE analysis identifies the genes that are statistically different in expression levels between the 2 selected conditions. Two important thresholds are:

- The lower bound of expression **fold change** between the 2 selected conditions
- The upper bound of hypothesis test p-value

GeneCloudOmics implements 3 popular methods to identify DE genes:

- DESeq2
- EdgeR
- NOISeq

For data with a single replicate in all experiment conditions, the NOISeq method can simulate technical replicates to carry out the DE analysis. The metadata file is required for DE Analysis.

Please make sure metadata contains all column names from the input data file and matches them with the experimental condition

For edgeR and DESeq2, a raw read counts data file must be provided. For NOISeq, gene expression should be normalized for sequencing depths (by select normalization method in Preprocessing tab if raw counts file is inputted, or by directly providing normalized gene expression)

To carry out the analysis, first, the user needs to specify DE methods, two conditions to compare (condition 2 is compared against condition 1), fold change cut-off value and False Discovery Rate (FDR or adjusted p-value) threshold, then hit the "Submit" button. By convention, DE genes are thresholded at FDR = 0.05 and 2-fold change. When the computation finishes, table of DE genes, volcano plot of DE result and dispersion plot of input data are displayed in their respective tabs. Please note that volcano plot and dispersion plot are only available for edgeR and DESeq2 methods.

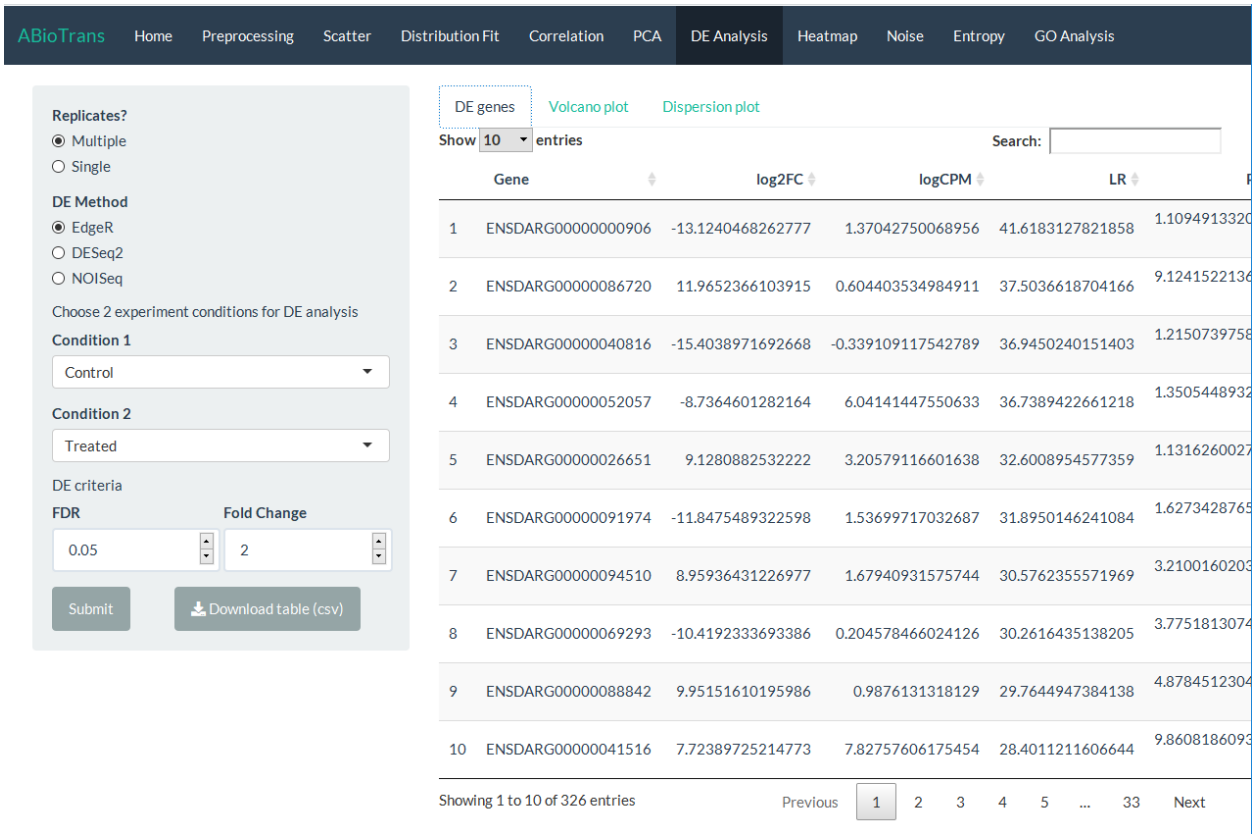


Figure 10. Table of DE genes computed by edgeR analysis method between Treated and Control conditions.

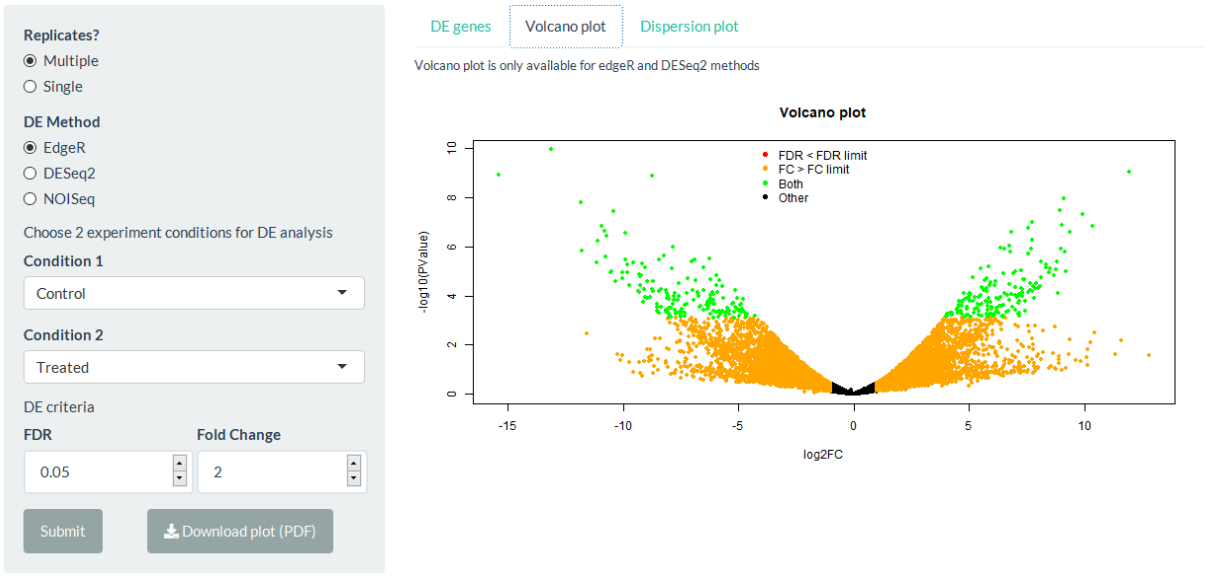


Figure 11. Volcano plot from edgeR DE analysis between Treated and Control conditions

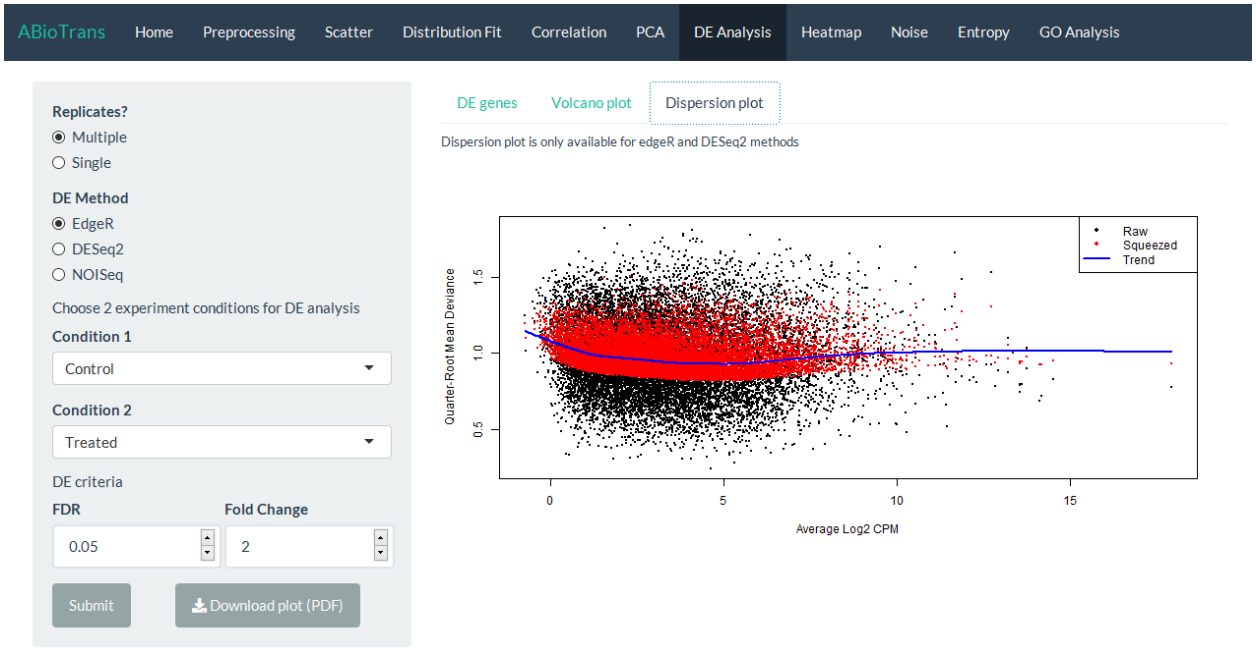


Figure 12. The plot of dispersion generated by edgeR method

6. Heatmap and gene clustering

GeneCloudOmics apply hierarchical clustering on the output of DE analysis using edgeR in the previous section (326 genes). Alternatively, the user can carry out clustering independently without going through DE analysis by specifying the minimum fold change of gene expression

between two samples. GeneCloudOmics also lists the name of genes for each cluster in the Gene clusters tab

Hierarchical clustering is used to find the groups of co-expressed genes. The clustering is performed on normalized expressions of differentially expressed genes using Ward clustering method. Normalized expression of the jth gene at time ti is defined as $z_j(t_i) = (x_j(t_i) - \bar{x_j})/\sigma_j$ where x_j(t_i) is the expression of the jth gene at time ti, x_j(bar) is the mean expression across all time points, and σj is the standard deviation.

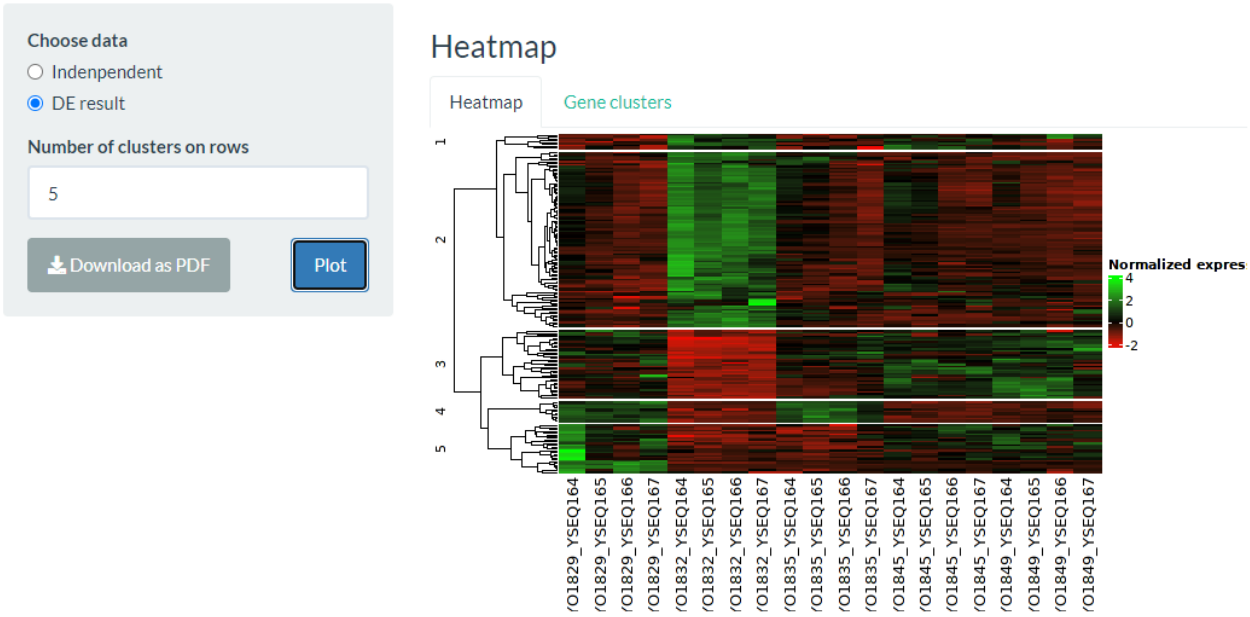


Figure 13. Heatmap and hierarchical clustering (5 clusters) on genes with at least 2-fold change in minimum 3 columns (carried out independently from DE analysis)

7. Transcriptome-wide average noise

To quantify between gene expressions scatter of all replicates in one experimental condition, we computed transcriptome-wide average noise for each cell type, defined as

$$\eta_{tot}^2 = \frac{1}{n}\sum_{i=1}^{n} \eta_i^2$$

where n is the number of genes and n_i^2 is the pairwise noise of the ith gene (variability between any two replicates), defined as $\eta_i^2 = \frac{2}{m(m-1)}\sum_{j=1}^{m-1}\sum_{k=j+1}^{m} \eta_{ijk}^2$, where m is the number of replicates in each condition and n_ijk^2 is the expression noise of the ith gene, defined by the variance divided by the squared mean expression in the pair of replicates (j,k).

Citation: Piras, V., Tomita, M. & Selvarajoo, K. Transcriptome-wide Variability in Single Embryonic Development Cells. Sci Rep 4, 7137 (2014). https://doi.org/10.1038/srep07137

8. Entropy

Shannon entropy (Shannon, 1948) measures the disorder of a high-dimensional system, where higher values indicate an increasing disorder. The entropy of each transcriptome, X, is defined as

$$H(\boldsymbol{X}) = -\sum_{i=1}^{n} p(x_i)\log_2 p(x_i)$$

where p(xi) is the probability of gene expression value x=xi.

Entropy values were obtained through histogram-based partitioning approach and the number of bins is determined using Doane's rule: $b(X)=1+\log 2n+\log 2(1+|gX|/\sigma g)$, where gX is the skewness of the expression distribution of each sample, and $\sigma g=\sqrt{6(n-2)/(n+1)(n+3)}$

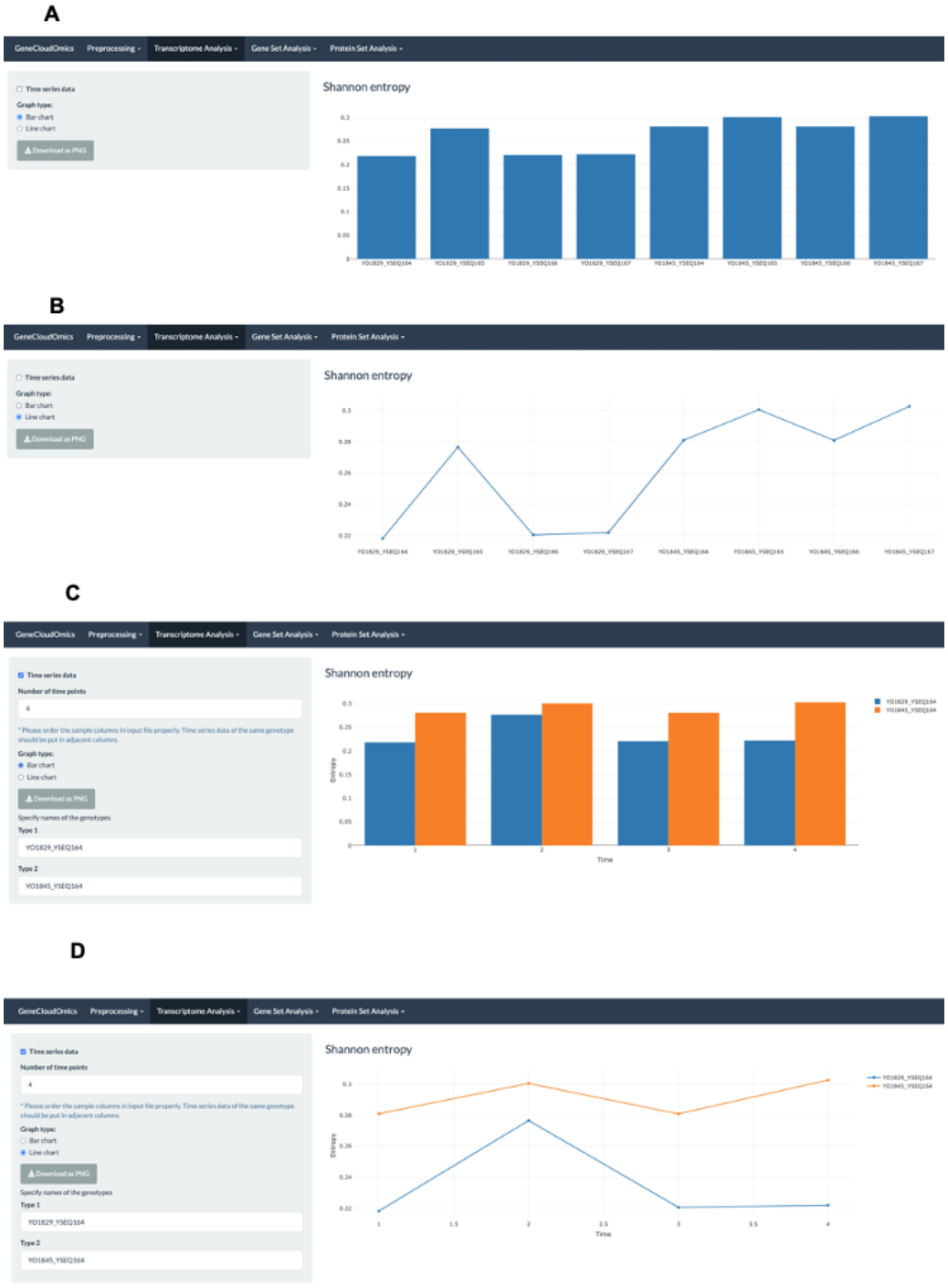


Figure 14. The Shannon entropy analysis on GeneCloudOmics. A) As bar plots for non-time points data, B) As a line plot for non-time points data, C) As bar plots for time-points data and D) As a line plot for time-points data

9. Random forest clustering

Clustering belongs to unsupervised learning, in which each sample is clustered into different classes, based on their similarity (usually based on Euclidean distance). The random forest algorithm is used to generate a proximity matrix - a rough estimate of the distance between samples based on the proportion of times the samples end up in the same leaf node of the decision tree. The proximity matrix is converted to a dist matrix which is then input to the hierarchical clustering algorithm. Implementation adapted from
https://nishanthu.github.io/articles/ClusteringUsingRandomForest.html

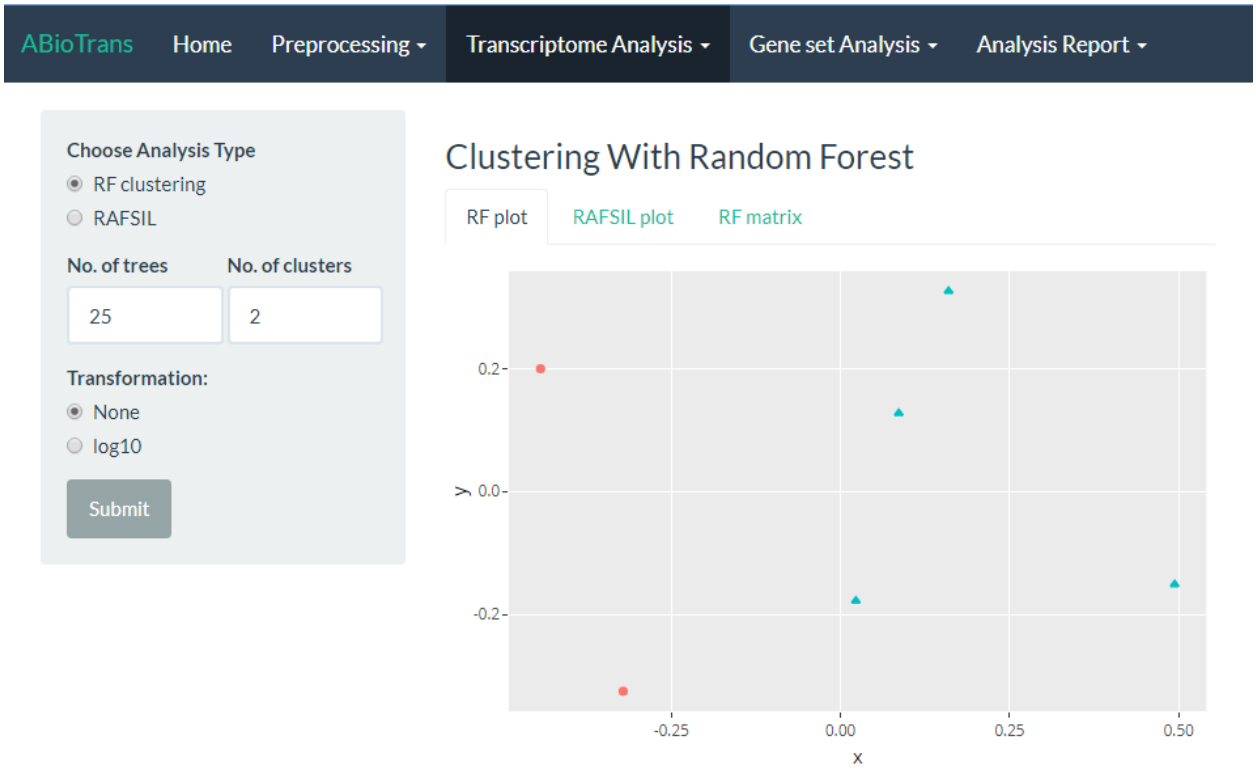


Figure 15. The Random Forest clustering analysis on GeneCloudOmics. The user determines the no. of trees and the no. of clusters and chooses whether to transform the data or not.

10. Self-organizing map (SOM)

A self-organizing map (SOM) produces a two-dimensional, discretized representation of the high-dimensional gene expression matrix, and is, therefore, a dimensionality reduction

technique. Self-organizing maps apply uses a neighbourhood function to preserve the topological properties of the input gene expression matrix.

Each data point (1 sample) in the input gene expression matrix recognizes itself by competing for representation. SOM mapping steps start from initializing the weight vectors. From there a sample vector is selected randomly and the map of weight vectors is searched to find which weight best represents that sample. Each weight vector has neighboring weights that are close to it. The weight that is chosen is rewarded by being able to become more like that randomly selected sample vector. The neighbours of that weight are also rewarded by being able to become more like the chosen sample vector. This allows the map to grow and form different shapes. Most generally, they form square/rectangular/hexagonal/L shapes in 2D feature space.

Citation: Villmann T., Bauber H. Applications of the growing self-organizing map. Neurocomputing 21, 1-3 (1998). https://doi.org/10.1016/S0925-2312(98)00037-X

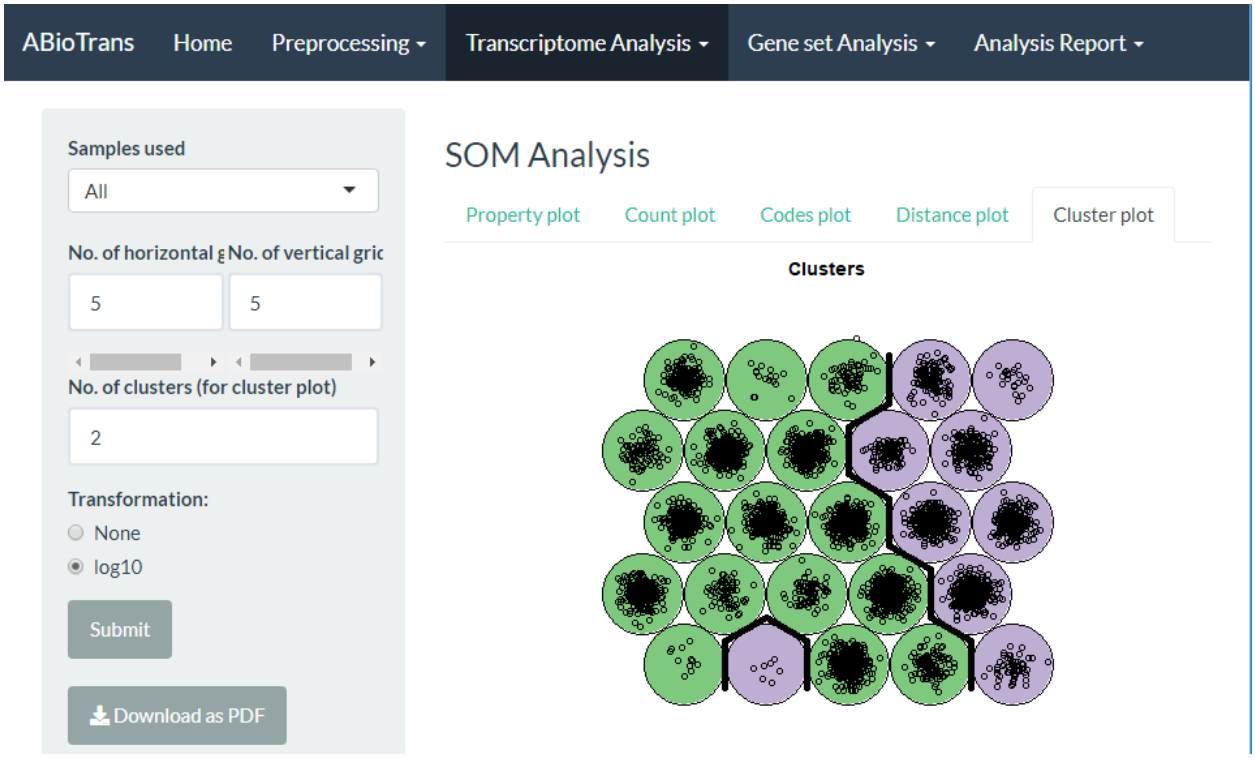


Figure 16. The self-organization map (SOM) analysis on GeneCloudOmics. The user determines the size of the grid and the no. of clusters and chooses whether to transform the data or not.

11. **t-distributed stochastic neighbour embedding (t-SNE)**

t-SNE is a dimensionality-reduction approach that reduces the complexity of highly complex data such as transcriptomic data. It visualizes the sample interrelations in a 2- or 3-dimensional visualization. This allows the identification of the close similarities between samples through the relative location of mapped points. Since t-SNE is nonlinear and able to control the trade-off between local and global relationships among points, its visualization of the clusters is usually more compelling when compared with the other methods (Cieslak et al 2020). GeneCloudOmics introduces an intuitive interface that allows performing t-SNE analysis on the processed untransformed transcriptomic data through entering three inputs perplexity value, the number of principal components (PC) and the number of clusters. The user can also choose to log transform the data before submission.

Figure 17. The t-SNE analysis on GeneCloudOmics.

Section II - Gene/Protein set analysis

DGE analysis usually outputs a list of genes that are statistically determined as differentially expressed. Then, the list of DEGs is analyzed, interpreting and annotated to learn more about the functions, pathways and cellular processes that these genes are involved in. Most of the gene differential expression analysis tools do not include bioinformatics features for gene set analysis or include few basic analyses such as GO and pathways enrichment. GeneCloudOmics provided one bioinformatics tool for interpreting the DEGs, that is gene ontology (GO) (Zou et al 2019). The GO enrichment at GeneCloudOmics was performed using three different enrichment tests: enrichR, clusterProfiler and GOstats (Falcon and Gentleman 2007, Yu et al 2012, Kuleshov et al 2016) using a local database that is part of the installation package of GeneCloudOmics. The local database content required continuous updating to keep it up-to-date. In GeneCloudOmics, we redesigned the GO feature to be dynamic by reading the GO terms associated with the genes/proteins directly from UniProt Knowledgebase (UniProt Consortium 2019) then visualize each of the three GO domains (cellular component, molecular function and biological process) in an independent tab in a bar chart and on a downloadable tabular format. Furthermore, we introduced 11 new bioinformatics analyses that can be performed on a given gene/protein dataset.

1) Pathways Enrichment Analysis: For a given gene or protein set, GeneCloudOmics uses g:Profiler (Raudvere et al 2019) to perform a pathway enrichment analysis and displays the results as a network where the nodes are the pathways and the edges are the overlap between the pathways. We use Cytoscape.JS for the network visualization (Franz et al 2016) and, therefore, the network properties such as the colour and layout can be changed from the left panel. The users can choose from nine different network layouts and five different network colour schemes. The overlap between the pathways can be also changed from the provided controllers so that the user can choose an overlap cutoff or overlap range. The enrichment results can also be downloaded as a CSV file.

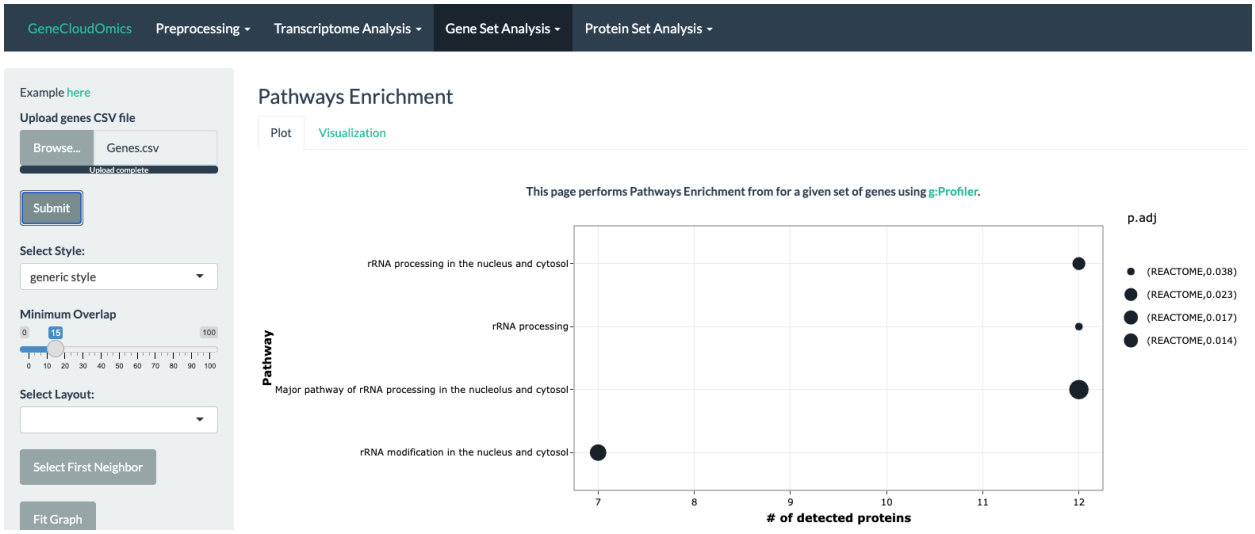


Figure 18. The pathway enrichment analysis of GeneCloudOmics.

2) Protein-Protein Interaction: investigating PPIs is one of the essential steps in systems biology studies. GeneCloudOmics provides the users with an interface where they can upload a set of proteins (UniProt accessions) and get all the interactions associated with them. The interactions are visualized as a network where the nodes are the proteins and the edges are the interactions and the node size corresponds to the number of interactors of the protein. We use Cytoscape.JS for PPI visualization (Franz et al 2016). We provide users with five network styles and nine network layouts to customize their results. The results are also displayed as an interactions table and can be downloaded as a network or an interaction table.

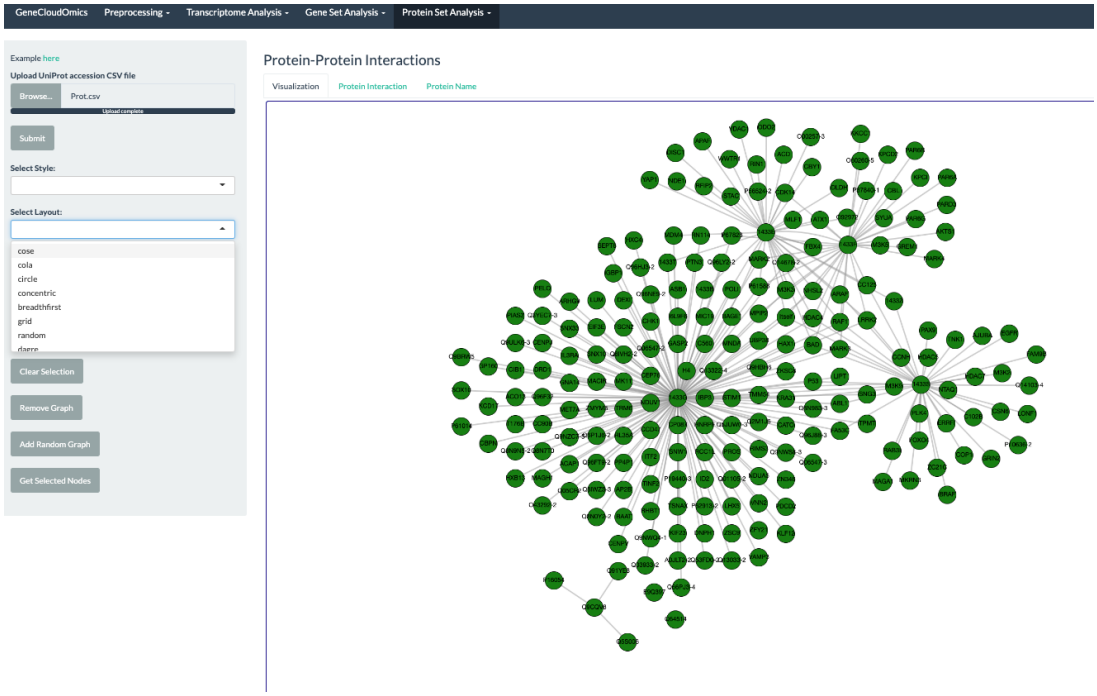


Figure 19. The protein-protein interaction (PPI) analysis of GeneCloudOmics.

3) Complex Enrichment: GeneCloudOmics provide the user with a complex enrichment feature that allows the identification of proteins in the provided dataset that are part of a known protein complex. This feature uses CORUM databases, which contain curated complex information for mammalian proteins (Giurgiu et al 2019). This feature provides the user with complex-forming proteins and the complex information in the submitted dataset.

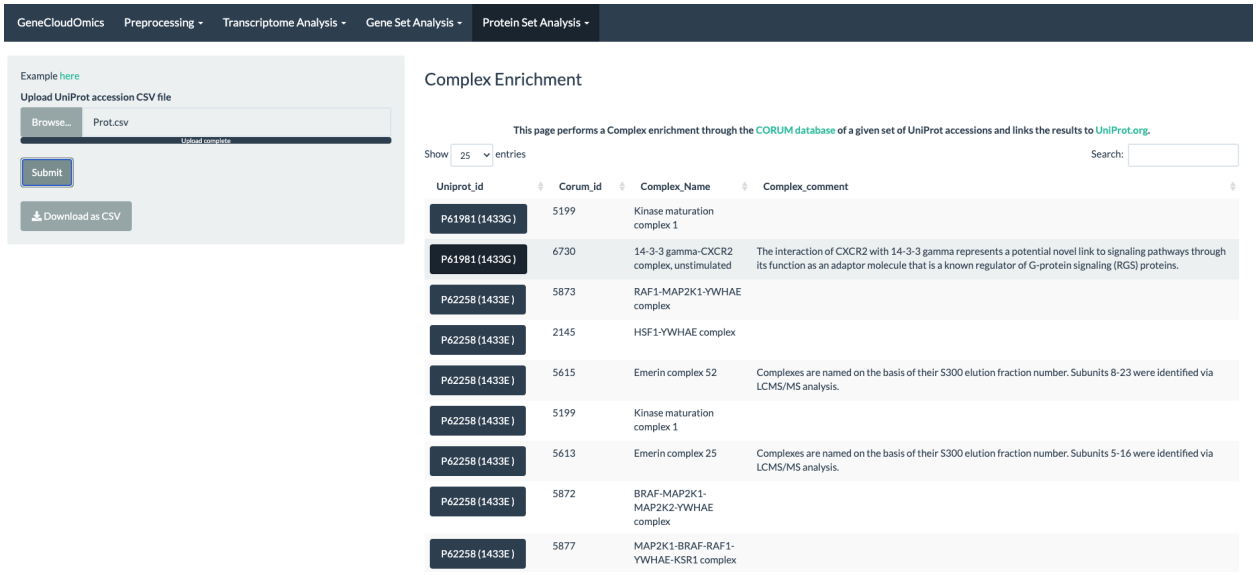


Figure 20. The complex enrichment analysis of GeneCloudOmics.

4) Protein Function: The protein function feature retrieves the protein function information from UniProt of a given protein set (UniProt accessions). The retrieved protein functions are displayed in a downloadable tabular format.

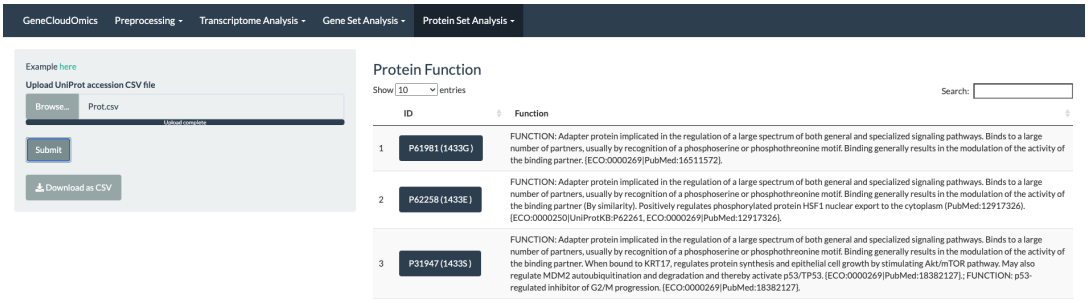


Figure 21. The protein functin analysis of GeneCloudOmics.

5) Protein Subcellular Localization: The protein subcellular localization feature of GeneCloudOmics provides the user with an interface to get the subcellular localization information for a given list of proteins (UniProt accessions) and display the results in a downloadable tabular format.

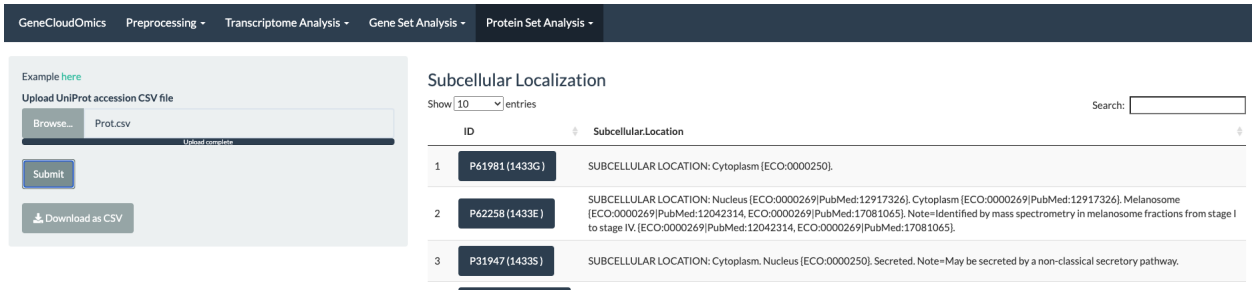


Figure 22. The protein subcellular localization analysis of GeneCloudOmics.

6) Protein Domains: GeneCloudOmics provides the users with a protein domain feature that connects to UniProt Knowledgebase and retrieves the domain information associated with each protein in a given list of UniProt accession.

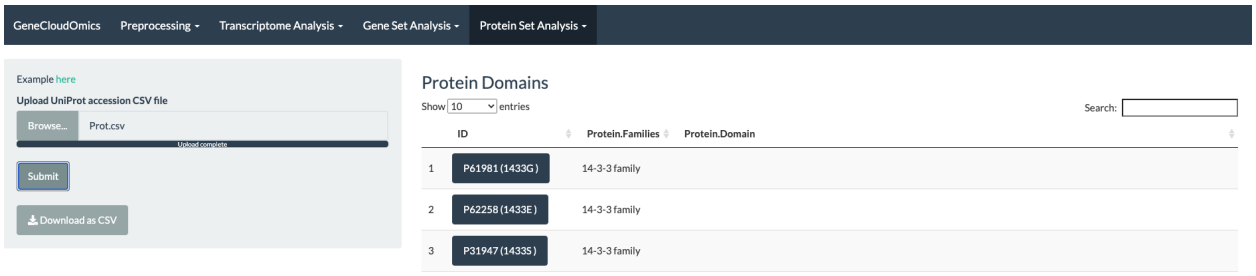


Figure 23. The protein domains analysis of GeneCloudOmics.

7) Tissue Expression: The tissue expression feature in GeneCloudOmics provides the user with the tissue expression for each protein in a given protein list (UniProt accessions) through retrieving this information from UniProt Knowledgebase. The result is displayed in a downloadable tabular format.

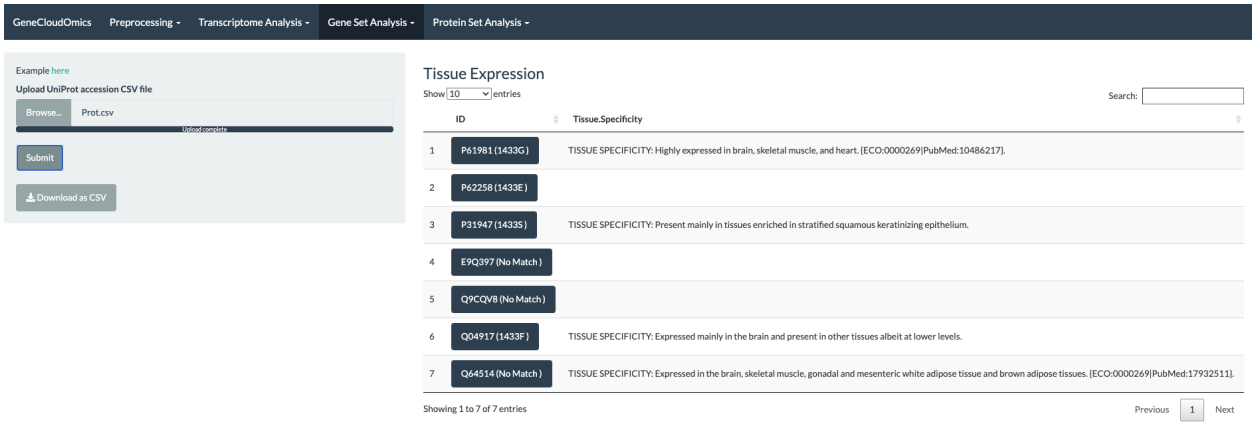


Figure 24. The tissue expression analysis of GeneCloudOmics.

8) Gene Co-expression: The co-expression analysis is a common analysis that assesses the expression level of different genes to identify simultaneously expressed genes. The resultant

co-expression networks are used to identify functionally related genes or genes being controlled by the same transcriptional mechanism (Vella et al 2017). GeneCloudOmics provides the users with an interface where they can submit a co-expression query to GeneMANIA (Franz et al 2018) then shows the results at GeneMANIA's website in a new tab. Currently, we support queries for nine model organisms including human, yeast, E. coli, C. elegans, Arabidopsis, Drosophila, zebrafish, mouse and rat.

9) Protein Physicochemical Properties: For a given set of proteins (UniProt accessions), this feature provides the user with the complete sequences of them in a single FASTA file and allows the user to investigate their physicochemical properties. The physicochemical analysis includes sequence charge, GRAVY index (Kyte 1982) and hydrophobicity.

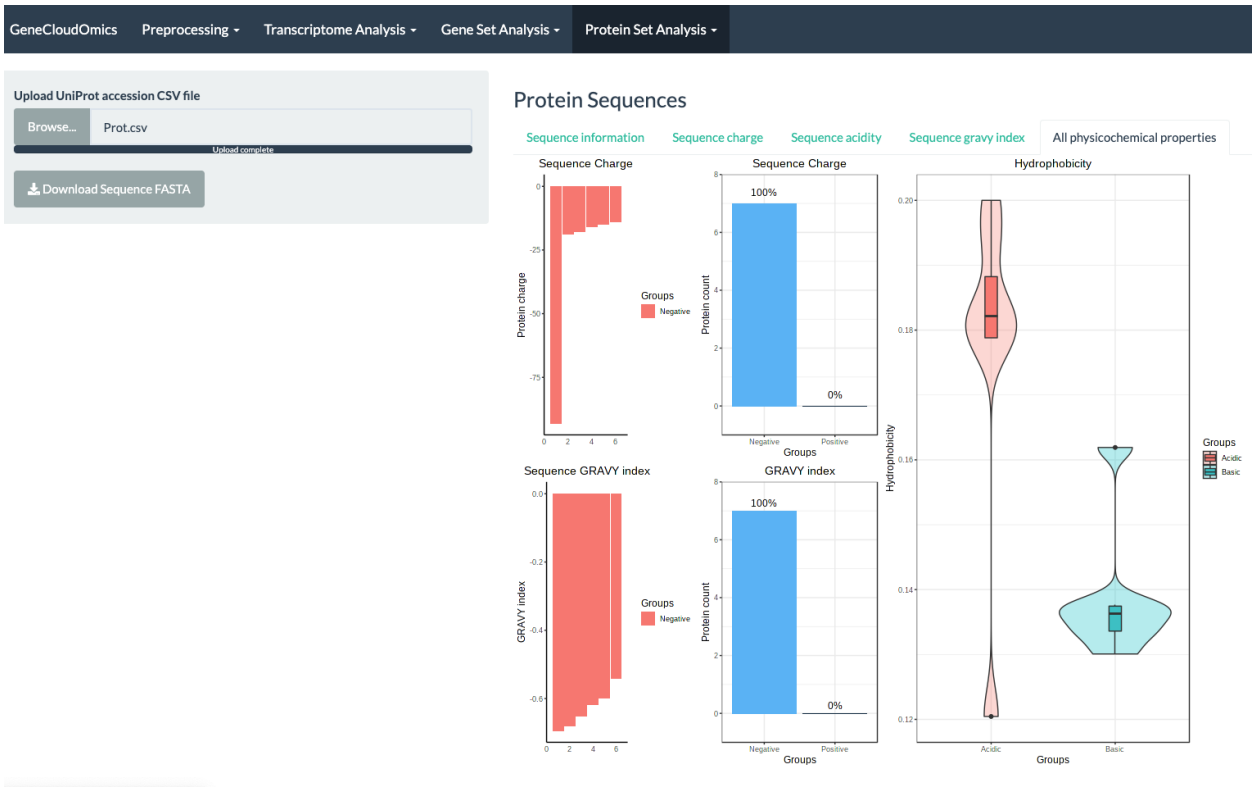


Figure 25. The protein physiochemical analysis of GeneCloudOmics.

10) Protein Evolutionary Analysis: For a given set of proteins (UniProt accessions), this feature provides the user with a phylogenetic and evolutionary analysis that include multiple sequence alignment (MSA) of the protein sequences, clustering based on the amino acid sequences, chromosomal location or gene tree.

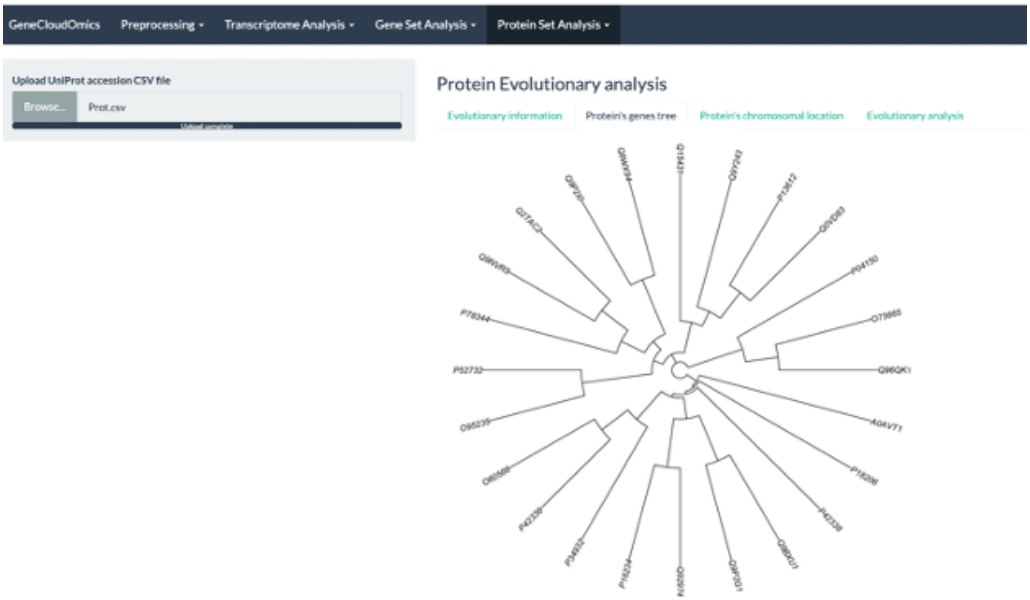


Figure 26. The protein evolutionarly analysis of GeneCloudOmics.

11) Protein Pathological Analysis: several diseases are associated with the malfunction of certain genes or proteins. The disease-protein association is collected in different online resources such as OMIM databases (Amberger et al 2019), DisProt (Hatos et al 2020) and DisGeNET (Pinero et al 2019). GeneCloudOmics provides the users with an interface that retrieves the disease-protein association from online databases for a given list of proteins (UniProt accessions). The disease-protein association is visualized as bubble chats that shows the distribution of the proteins among the disease (each bubble is protein and the bubble size is the number of associated diseases) or the distribution of proteins among the diseases (each bubble is a disease and the bubble size is the number of associated proteins).

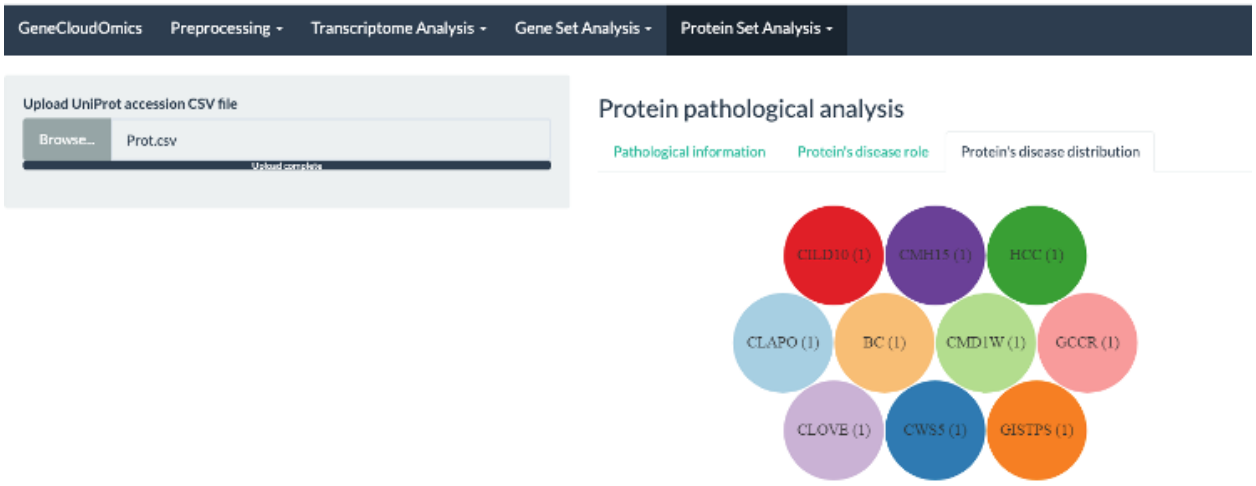


Figure 27. The protein pathological analysis of GeneCloudOmics.

Packages using in GeneCloudOmics

Pame	**Repository**	**Version**
cyjShiny	GitHub	0.99.8
RColorBrewer	CRAN	1.1-2
tidyverse	C**ackage** **N**RAN	1.3.1
networkD3	CRAN	0.2.10
data.tree	CRAN	1.0.0
bubbles	GitHub	0.2
UniprotR	CRAN	2.0.8
scales	CRAN	1.1.1
gprofiler2	CRAN	0.2.1
alakazam	CRAN	1.0.0
httr	CRAN	1.4.2
curl	CRAN	4.3.2
msa	Bioconductor	1.24.0
ape	CRAN	5.5
seqinr	CRAN	4.2-8
qdapRegex	CRAN	0.7.2
RUVSeq	Bioconductor	3.1
ArrayExpress	Bioconductor	1.49
fitdistrplus	CRAN	1.1

DESeq2	Bioconductor	1.2
EdgeR	Bioconductor	3.24
NOISeq	Bioconductor	2.31
entropy	CRAN	1.2
randomForest	CRAN	4.3
Rtsne	CRAN	0.13
RAFSIL	Kostka Lab	1
kohonen	CRAN	3