Package 'AutoPipe'

January 20, 2025

Type Package

Title Automated Transcriptome Classifier Pipeline: Comprehensive Transcriptome Analysis

Version 0.1.6

Author Karam Daka [cre, aut], Dieter Henrik Heiland [aut]

Maintainer Karam Daka <k.dacca@gmail.com>

Description An unsupervised fully-automated pipeline for transcriptome analysis or a supervised option to identify characteristic genes from predefined subclasses. We rely on the 'pamr' <http://www.bioconductor.org/packages//2.7/bioc/html/pamr.html> clustering algorithm to cluster the Data and then draw a heatmap of the clusters with the most significant genes and the least significant genes according to the 'pamr' algorithm. This way we get easy to grasp heatmaps that show us for each cluster which are the clusters most defining genes.

License GPL-3

Encoding UTF-8

LazyData true

Imports cluster ,pamr ,siggenes ,annotate ,fgsea ,org.Hs.eg.db ,RColorBrewer ,ConsensusClusterPlus ,Rtsne ,clusterProfiler ,msigdbr

Depends R (>= 3.5.0)

RoxygenNote 6.1.1

NeedsCompilation no

Repository CRAN

Date/Publication 2019-02-27 17:00:36 UTC

Contents

`AutoPipe_tSNE` *Implemented t-distributed stochastic neighbor embedding*

Description

This function is used to upload a table into R for further use in the AutoPipe

Usage

```
AutoPipe_tSNE(me,perplexity=30,max_iter=500,groups_men)
```

Arguments

`me`	The path of the expression table
`perplexity`	numeric; Perplexity parameter
`max_iter`	integer; Number of iterations (default: 1000)
`groups_men`	the data frame with the group clustering that the function Groups_Sup or top_supervised (2. place on the list) returns with the data about each sample and its coresspond-ing cluster.

`cons_clust` *A function to plot do a Consensus clustering to validate the results*

Description

this function calls the ConsensusClusterPlus function with thedaraset and plots a plot with the heatmaps of the clustering for each number of clusters from 2 to max_clust

Usage

```
cons_clust(data,max_clust,TOPgenes)
```

Arguments

data	this is the data for the ConsensusClusterPlus
max_clust	the max number of clusters that should be evaluated.
TOPgenes	the number of the top genes to choose for the clustering

Value

plots a plot with all the heatmaps from the ConsensusClusterPlus for the number ofd clusters 2 to max_clust the same return value as the COnsensusClusterPlus

Examples

```
data(rna)
cons_clust(rna,5,TOPgenes=50)
```

Groups_Sup *cluster the samples*

Description

This function clusters the samples into x clusters.

Usage

```
Groups_Sup(me_TOP, me, number_of_k,TRw)
```

Arguments

me_TOP	the matrix with the n top genes, usually the from output of the function TopPAM
me	the original expression matrix. (with genes in rows and samples in columns).
number_of_k	the number of clusters
TRw	threshold for the elemenation of the samples with a Silhouette width lower than TRw. Default value is -1.

Examples

```
## load data
library(org.Hs.eg.db)
data(rna)
me_x=rna
res<-AutoPipe::TopPAM(me_x,max_clusters = 8, TOP=100)
me_TOP=res[[1]]
number_of_k=res[[3]]
File_genes=Groups_Sup(me_TOP, me=me_x, number_of_k,TRw=-1)
groups_men=File_genes[[2]]
me_x=File_genes[[1]]
```

read_expression_file *Input Expression File*

Description

This function is used to upload a table into R for further use in the AutoPipe

Usage

```
read_expression_file(file, format = "csv", sep=";",gene_name="SYMBOL", Trans=FALSE)
```

Arguments

file	The path of the expression table
format	The format of the table "csv" or "txt"
sep	The seperator of the input table
gene_name	Genes are given in "SYMBOL" or "ENTREZID"
Trans	Need Matrix Transpose TRUE or FALSE

Value

A data.frame with a gene expression matrix

rna *rna egene expression of 48 meningiomas*

Description

A dataset containing the gene expression data od 48 meningioma tumors

Usage

```
rna
```

Format

A data frame with 200 rows and 48 variables:

BT_1008 sample BT_1008,

BT_1017 sample BT_1017,

BT_1025 sample BT_1025,

BT_1042 sample BT_1042,

BT_1050 sample BT_1050,

BT_1056 sample BT_1056,
BT_1065 sample BT_1065,
BT_1067 sample BT_1067,
BT_1072 sample BT_1072,
BT_1078 sample BT_1078,
BT_1082 sample BT_1082,
BT_1091 sample BT_1091,
BT_1094 sample BT_1094,
BT_1097 sample BT_1097,
BT_1115 sample BT_1115,
BT_605 sample BT_605,
BT_617 sample BT_617,
BT_619 sample BT_619,
BT_633 sample BT_633,
BT_634 sample BT_634,
BT_644 sample BT_644,
BT_654 sample BT_654,
BT_659 sample BT_659,
BT_690 sample BT_690,
BT_695 sample BT_695,
BT_700 sample BT_700,
BT_738 sample BT_738,
BT_751 sample BT_751,
BT_771 sample BT_771,
BT_797 sample BT_797,
BT_803 sample BT_803,
BT_808 sample BT_808,
BT_820 sample BT_820,
BT_837 sample BT_837,
BT_855 sample BT_855,
BT_862 sample BT_862,
BT_873 sample BT_873,
BT_882 sample BT_882,
BT_887 sample BT_887,
BT_900 sample BT_900,
BT_905 sample BT_905,
BT_907 sample BT_907,

BT_920 sample BT_920,

BT_944 sample BT_944,

BT_962 sample BT_962,

BT_963 sample BT_963,

BT_982 sample BT_982,

BT_990 sample BT_990, ...

`Supervised_Cluster_Heatmap`

Produce a Heatmap using a Supervised clustering Algorithm

Description

This function produces a plot with a Heatmap using a supervised clustering algorithm which the user choses. with a the mean Silhouette width plotted on the right top corner and the Silhouette width for each sample on top. On the right side of the plot the n highest and lowest scoring genes for each cluster will added. And next to them the coressponding pathways (see Details)

Usage

```
Supervised_Cluster_Heatmap(groups_men, gene_matrix,
method="PAMR",TOP=1000,TOP_Cluster=150,
show_sil=FALSE,show_clin=FALSE,genes_to_print=5,
print_genes=FALSE,samples_data=NULL,colors="RdBu",
GSE=FALSE,topPaths=5,db="c2",plot_mean_sil=FALSE,stats_clust =NULL,threshold=2)
```

Arguments

`groups_men`	the data frame with the group clustering that the function Groups_Sup or top_supervised (2. place on the list) returns with the data about each sample and its coressponding cluster.
`gene_matrix`	the matrix of n selected genes that the function Groups_Sup returns
`method`	the method to cluster of Clustering. The default is "PAMR" which uses the pamr library. other methods are SAM and our own "EXReg" (see details)
`TOP`	the number of the top genes to take. the default value is 1000.
`TOP_Cluster`	a numeric variable for the number of genes to include in the clusters. Default is 150.
`show_sil`	a logical value that indicates if the function should show the Silhouette width for each sample. Default is FALSE.
`show_clin`	a logical value if TRUE the function will plot the clinical data provided by the user. Default value is FALSE.
`genes_to_print`	the number of genes to print for each cluster. this function adds on the right side. of the heatmap the n highest expressed genes and the n lowest expressed genes for each cluster. Default value is 5.

print_genes	a logical value indicating if or not to plot the TOP genes for each cluster.Default value is FALSE.
samples_data	the clinical data provided by the user to plot under the heatmap. it will be plotted only if show_clin is TRUE. Default value is NULL. see details for format.
colors	the colors for the Heatmap. The function RColorBrewer palletes.
GSE	a logical variable that indicates wether to plot thr Gene Set Enrichment Analysis next to the heatmap. Default value is FALSE.
topPaths	a numerical value that says how many pathways the Gene Set Enrichment plots should contain fo each cluster. Default value is 5.
db	a value for the database for the GSE to be used. Default value is "c1". the paramater can one of the values: "c1","c2","c3",c4","c5","c6","c7","h". See the broad institue GSE GSE webpage for further information in each dataset.
plot_mean_sil	A logical value. if TRUE the function plots the mean of the Silhouette width for each cluster number or gap statistic.
stats_clust	A vector with the mean Silhouette widths or gap statistic for the number of clusters. The first value should be for 2 Clusters. 2nd is for 3 clusters and so on.
threshold	the threshhold for the pam analysis default is 2.

Details

sample data should be a data.frame with the sample names as rownames and the clinical triats as columns. each trait must be a numeric variable.

Examples

```
##load the org.Hs.eg Library
library(org.Hs.eg.db)
## load data
data(rna)
me_x=rna
## calculate best number of clusters and
res<-AutoPipe::TopPAM(me_x,max_clusters = 6, TOP=100)
me_TOP=res[[1]]
number_of_k=res[[3]]
File_genes=Groups_Sup(me_TOP, me=me_x, number_of_k,TRw=-1)
groups_men=File_genes[[2]]
me_x=File_genes[[1]]
o_g<-Supervised_Cluster_Heatmap(groups_men = groups_men, gene_matrix=me_x,
    method="PAMR",show_sil=TRUE,print_genes=TRUE,threshold=0,
    TOP = 100,GSE=FALSE,plot_mean_sil=TRUE,stats_clust=res[[2]])
```

TopPAM *Compute Top genes*

Description

This function computes the n=TOP genes and the the best number of clusters

Usage

```
TopPAM(me, max_clusters=15,TOP=1000,B=100,clusterboot=FALSE)
```

Arguments

me	a matrix with genes in rows and samples in columns
max_clusters	max. number of clusters to check
TOP	the number of genes to take.
B	integer, number of Monte Carlo ("bootstrap") samples.
clusterboot	A logical value indicating wether or not to calculate the Gap statistic and to bootstrap.

Details

we use the clusGap algorithm from the package cluster to calculate the Gap statistic.

Value

a list of 1. A matrix with the top genes 2. A list of means of the Silhouette width for each number of clusters. 3. The optimal number of clusters. 4. gap_st the gap statistic of the clustering 5. best number of clusters according to the gap statistic.

Examples

```
##load the org.Hs.eg Library
library(org.Hs.eg.db)
#' ## load data
data(rna)
me_x=rna
res<-AutoPipe::TopPAM(me_x,max_clusters = 8, TOP=100,clusterboot=FALSE)
me_TOP=res[[1]]
number_of_k=res[[3]]
```

top_supervised *A Function for Assisting Supervised Clustering*

Description

when perfoming a supervised clustering the user should run this function in order to get the best results.

Usage

```
top_supervised(me,TOP=1000,cluster_which,TRw=-1)
```

Arguments

me	the matrix of the gene exporessions, the olums should be the samples and the colnames the sample names the rownames should be the genes . at best the ENTEREZID
TOP	the top genes to choose, default is 100.
cluster_which	a dataframe with the supervised clustering arrangment of the samples. the dataframe should have the sample names in the first column and the clustering in the secound column.
TRw	the threshhold for excluding samples with silhouette width < TRw

Value

a list. the first place is the expression matrix, the secound is the silhouette for each sample.

Examples

```
library(org.Hs.eg.db)
data(rna)
cluster_which<-cbind(colnames(rna),c(rep(1,times=24),rep(2,times=24)))
me_x=rna
## calculate best number of clusters and
res<-top_supervised(me_x,TOP = 100,cluster_which)
me_TOP=res[[1]]
number_of_k=2
groups_men=res[[2]]
me_x=me_TOP
colnames(me_x)
o_g<-Supervised_Cluster_Heatmap(groups_men = groups_men, gene_matrix=me_x,
                            method="PAMR",show_sil=TRUE,print_genes=TRUE,threshold = 0,
                          TOP = 100,GSE=FALSE,plot_mean_sil=FALSE,stats_clust=res[[2]],
                              samples_data = as.data.frame(groups_men[,1,drop=FALSE]))
```

UnSuperClassifier *Unsupervised Clustering*

Description

A function for unsupervised Clustering of the data

Usage

```
UnSuperClassifier(data,clinical_data=NULL,thr=2,TOP_Cluster=150,TOP=100)
```

Arguments

data	the data for the clustering. Data should be in the following format: samples in columns and the genes in the rows (colnames and rownames accordingly). The rownames should be Entrez ID in order to plot a gene set enrichment analysis.
clinical_data	the clinical data provided by the user to plot under the heatmap. it will be plotted only if show_clin is TRUE. Default value is NULL. see details for format.
thr	The threshold for the PAMR algorithm default is 2.
TOP_Cluster	numeric; Number of genes in each cluster.
TOP	numeric; the number of the TOP genes to take from the gene exoression matrix see TopPAM TOP.

Details

sample data should be a data.frame with the sample names as rownames and the clinical triats as columns. each trait must be a numeric variable. @return the function is an autated Pipeline for clustering it plot cluster analysis for the geneset

Index