Package ‘TSCAN’

April 10, 2015

Type Package

Title TSCAN: Tools for Single-Cell ANalysis

Version 1.2.0

Date 2014-09-24

Author Zhicheng Ji, Hongkai Ji

Maintainer Zhicheng Ji <zji4@jhu.edu>

Description TSCAN enables users to easily construct and tune pseudotemporal cell ordering as well as analyzing differentially expressed genes. TSCAN comes with a user-friendly GUI written in shiny. More features will come in the future.

License GPL(>=2)

Imports ggplot2, shiny, plyr, grid, fastICA, igraph, TSP, combinat, mgcv, gplots

VignetteBuilder knitr

Suggests knitr

Depends R(>= 2.10.0)

biocViews GeneExpression, Visualization, GUI

R topics documented:

difftest *difftest*

Description

testing differentially expressed genes

Usage

```
difftest(data, pseudotime, df = 3)
```

Arguments

`data` The raw single_cell data, which is a numeric matrix or data.frame. Rows represent genes/features and columns represent single cells.

`pseudotime` The pseudotime information. It is typically the first element of the return value of function `TSPpseudotime`.

`df` Numeric value specifying the degree of freedom used in the GAM model.

Details

This function tests whether a gene is significantly expressed given pseudotime ordering. Generalized additive model (GAM) with user-specified degrees of freedoms is compared with a constant fit to get the p-values. The p-values are adjusted for multiple testing using fdr to gain qvalues.

Value

Data frame containing pvalues and qvalues of testing differentially expression.

Author(s)

Zhicheng Ji, Hongkai Ji <zji4@zji4.edu>

See Also

`TSPpseudotime` for examples

Examples

```
data(lpsdata)
procdata <- preprocess(lpsdata)
#Choose STAT2 gene expression as marker gene
STAT2expr <- log2(lpsdata["STAT2",]+1)
lpspseudotime <- TSPpseudotime(procdata, geneexpr = STAT2expr, dim = 2)
diffval <- difftest(procdata,lpspseudotime[[1]])
#Selected differentially expressed genes under qvlue cutoff of 0.05
row.names(diffval)[diffval$qval < 0.05]
```

`lpsdata` *Sinlge-cell RNA-seq data for BMDC cells before and after LPS stimulation*

Description

The dataset contains 16776 rows and 131 columns. Each row represent a gene and each column represent a single cell. This dataset is a subset of single-cell RNA-seq data provided by GEO GSE48968. Only unstimulated cells and cells after 6h of LPS stimulation are retained for the purpose of demonstration. Genes which have raw expression values of greater than zero in at least one cell are retained. For the original dataset please refer to GSE48968 on GEO (http://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?

Format

A matrix with 16776 rows and 131 variables

Source

`http://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE48968`

References

Shalek, A. K., Satija, R., Shuga, J., Trombetta, J. J., Gennert, D., Lu, D., ... & Regev, A. (2014). Single-cell RNA-seq reveals dynamic paracrine control of cellular variation. Nature.

`orderscore` *orderscore*

Description

Calculate pseudotemporal ordering scores for orders

Usage

```
orderscore(subpopulation, orders)
```

Arguments

`subpopulation`	Data frame with two columns. First column: cell names. Second column: subpopulation codes.
`orders`	A list with various length containing pseudotime orderings. Each pseudotime ordering is typically the first element of the return value of function `TSPpseudotime`.

Details

This function calculates pseudotemporal ordering scores (POS) based on the sub-population information and order information given by users. Exactly two sub-population indicating early and late time points should be given. The early sub-population will be coded as 0 while the late sub-population will be coded as 1.

Value

a numeric vector of calculated POS.

Author(s)

Zhicheng Ji, Hongkai Ji <zji4@zji4.edu>

Examples

```
data(lpsdata)
procdata <- preprocess(lpsdata)
subpopulation <- data.frame(cell = colnames(procdata), sub = ifelse(grepl("Unstimulated",colnames(procdata)),0,1
#Choose STAT2 gene expression as marker gene
STAT2expr <- log2(lpsdata["STAT2",]+1)
#Comparing ordering with or without marker gene information
order1 <- TSPpseudotime(procdata, geneexpr = STAT2expr, dim = 2)
order2 <- TSPpseudotime(procdata, dim = 2)
orders <- list(order1[[1]],order2[[1]])
orderscore(subpopulation, orders)
```

`plotpseudotime` *plotpseudotime*

Description

Plot the TSP constructed pseudotime time

Usage

```
plotpseudotime(pseudotimedata, x = 1, y = 2, show_tree = T,
  show_cell_names = T, cell_name_size = 3, markerexpr = NULL)
```

Arguments

`pseudotimedata` The exact output from `TSPpseudotime`.

`x` The column of data after dimension reduction to be plotted on the horizontal axis.

`y` The column of data after dimension reduction to be plotted on the vertical axis.

`show_tree` Whether to show the links between cells connected in the minimum spanning tree.

`show_cell_names`

Whether to draw the name of each cell in the plot.

`cell_name_size` The size of cell name labels if show_cell_names is TRUE.

`markerexpr` The gene expression used to define the size of nodes.

Details

This function will plot the gene expression data after dimension reduction and link the data points with the constructed pseudotime path. It is written by plot_spanning_tree functioin in package monocle.

Value

A ggplot2 object.

Author(s)

Zhicheng Ji, Hongkai Ji <zji4@zji4.edu>

References

Cole Trapnell and Davide Cacchiarelli et al (2014): The dynamics and regulators of cell fate decisions are revealed by pseudo-temporal ordering of single cells. Nature Biotechnology

See Also

`TSPpseudotime` for examples

Examples

```
data(lpsdata)
procdata <- preprocess(lpsdata)
#Choose STAT2 gene expression as marker gene
STAT2expr <- log2(lpsdata["STAT2",]+1)
lpspseudotime <- TSPpseudotime(procdata, geneexpr = STAT2expr, dim = 2)
plotpseudotime(lpspseudotime, markerexpr = STAT2expr)
```

`preprocess` *preprocess*

Description

preprocess the raw single-cell data

Usage

```
preprocess(data, takelog = TRUE, logbase = 2, pseudocount = 1,
  minexpr_value = 1, minexpr_percent = 0.5, cvcutoff = 1)
```

Arguments

`data` The raw single_cell data, which is a numeric matrix or data.frame. Rows represent genes/features and columns represent single cells.

`takelog` Logical value indicating whether to take logarithm

`logbase` Numeric value specifiying base of logarithm

`pseudocount` Numeric value to be added to the raw data when taking logarithm

`minexpr_value` Numeric value specifying the minimum cutoff of log transformed (if takelog is TRUE) value

`minexpr_percent`
Numeric value specifying the lowest percentage of highly expressed cells (expression value bigger than minexpr_value) for the genes/features to be retained.

`cvcutoff` Numeric value specifying the minimum value of coefficient of variance for the genes/features to be retained.

Details

This function first takes logarithm of the raw data and then filters out genes/features in which too many cells are low expressed. It also filters out genes/features with low coefficient of variance which indicates the genes/features does not contain much information. The default setting will first take log2 of the raw data after adding a pseudocount of 1. Then genes/features in which at least half of cells have expression values are greater than 1 and the coefficeints of variance across all cells are at least 1 are retained.

Value

Matrix or data frame with the same format as the input dataset.

Author(s)

Zhicheng Ji, Hongkai Ji <zji4@zji4.edu>

See Also

`TSPpseudotime` for examples

Examples

```
data(lpsdata)
procdata <- preprocess(lpsdata)
```

singlegeneplot *singlegeneplot*

Description

plot expression values of individual genes against pseudotime axis

Usage

```
singlegeneplot(geneexpr, pseudotime, cell_size = 2)
```

Arguments

geneexpr	The gene expression values. Names should agree with the pseudotime information.
pseudotime	The pseudotime information. It is typically the first element of the return value of function TSPpseudotime.
cell_size	Size of cells in the plot.

Details

This function plots the expression values of individual genes against given pseudotime

Value

ggplot2 object.

Author(s)

Zhicheng Ji, Hongkai Ji <zji4@zji4.edu>

See Also

TSPpseudotime for examples

Examples

```
data(lpsdata)
procdata <- preprocess(lpsdata)
#Choose STAT2 gene expression as marker gene
STAT2expr <- log2(lpsdata["STAT2",]+1)
lpspseudotime <- TSPpseudotime(procdata, geneexpr = STAT2expr)
#Choose STAT1 gene expression to plot
STAT1expr <- log2(lpsdata["STAT1",]+1)
singlegeneplot(STAT1expr, lpspseudotime[[1]])
```

TSCAN *TSCAN: Tools for Single-Cell ANalysis*

Description

This package provides essential tools used in analyzing data from single-cell experiments

Details

TSCAN enables users to easily construct and tune pseudotemporal cell ordering as well as analyzing differentially expressed genes. TSCAN comes with a user-friendly GUI written in shiny. More functions will come in the future.

TSCANui *TSCANui*

Description

Launch the TSCAN user interface in local machine

Usage

```
TSCANui()
```

Details

This function will automatically launch the TSCAN user interface in a web browser. The user interface provides many powerful functions which is not available by command line programming. It also provides a much easier and more convenient way to quickly explore single cell data and construct pseudotime analysis. The user interface can also be accessed by http://zhiji.shinyapps.io/TSCAN. Neither R nor any packages are required in this online version. However, it is highly recommended that the user interface be launched locally for faster running speed.

Author(s)

Zhicheng Ji, Hongkai Ji <zji4@zji4.edu>

Examples

```
## Not run:
   TSCANui()

## End(Not run)
```

TSPpseudotime *TSPpseudotime*

Description

Construct pseudotime using Travelling Salesman Problem (TSP) algorithm

Usage

```
TSPpseudotime(data, dim = "auto", statenum = 3, scale = TRUE,
  startpoint = NULL, flip = FALSE, geneexpr = NULL,
  exprtrend = "increasing", maxtime = 100, kmeansiter = 10)
```

Arguments

`data`	The raw single_cell data, which is a numeric matrix or data.frame. Rows represent genes/features and columns represent single cells.
`dim`	Either "auto" or a numeric value specifying the reduced dimenionality of PCA. If "auto" the optimal dimension will be chosen automatically.
`statenum`	Numeric value specifiying number of cell states in K-means clustering.
`scale`	Whether to scale gene expressions across all cells to have zero mean and unit variance. Normally this argument should be TRUE.
`startpoint`	Manually specify the starting point of TSP ordering. Should be one of the column names of data. Omitted when geneexpr is not null.
`flip`	Logical value specifying whether to flip the ordering.
`geneexpr`	Gene expression values used to determine optimal starting point. Gene expression values should change monotonically in the true biological process. Names should agree exactly with column names of data (can be of different order).
`exprtrend`	Trend of gene expression values. Either increasing or decreasing.
`maxtime`	Numeric value to specify the pseudotime of the ending point.
`kmeansiter`	Number of iterations of K-means clustering. The function will automatically pick an optimal clustering.

Details

This function first uses principal component analysis (PCA) to reduce dimensionality of original data. If not specified by the user, the optimal dimension will be automatically selected by fitting a set of continuous piecewise regression to the standard deviations of first 20 principal components and choose the one with the smallest residual sum of squares. The distance matrix between cells are calculated based on the reduced data. Then the function uses nearest insertion algorithm to construct a TSP path as a suboptimal solution. Because TSP path does not have a definite starting/ending point, users can specify a starting point, otherwise a random cell will be chosen as the starting point. Users can also use the expression value of a gene to determine the optimal starting point. The gene expression value must change monotonically over the true biological process. K-means clustering will be used to determine the different stages of cell during the biological process.

Value

a list containing

- pseudotime Data frame containing the constructed pseudotime information. First column: cell name. Second column: cell states. Third column: Pseudotime.
- reduceres Matrix storing the gene expression data after dimension reduction using PCA.

Author(s)

Zhicheng Ji, Hongkai Ji <zji4@zji4.edu>

References

Rosenkrantz, D. J., Stearns, R. E., & Lewis, II, P. M. (1977). An analysis of several heuristics for the traveling salesman problem. SIAM journal on computing, 6(3), 563-581.

Examples

```
data(lpsdata)
procdata <- preprocess(lpsdata)
#Choose STAT2 gene expression as marker gene
STAT2expr <- log2(lpsdata["STAT2",]+1)
TSPpseudotime(procdata, geneexpr = STAT2expr, dim = 2)
```

Index