Package 'MFSelector'

May 15, 2014

Version 1.0

Date 2014-05-15

Title Monotonic Feature Selector (MFSelector) in R

Author Hsei-Wei Wang <hwwang@ym.edu.tw>, Hsing-Jen Sun <rodney7646@gmail.com>, Ting-Yu Chang <taiwanbird@gmail.com>, Hung-Hao Lo <jerry5790489@me.com>, Wei-Chung Cheng <cwc0702@gmail.com>, George C. Tseng <ctseng@pitt.edu>, Chin-Teng Lin <ctlin@mail.nctu.edu.tw>, Shing-Jyh Chang <justine3@ms8.hinet.net>, Nikhil Ranjan Pal <nikhil@isical.ac.in>, I-Fang Chung <ifchung@ym.edu.tw>

Description

Discovering monotonic stemness marker genes from time-series stem cell microarray data.

Depends R (window version, >= 3.0.2), parallel

URL http://microarray.ym.edu.tw/tools/MFSelector/

R topics documented:

`mfselector` *Monotonic Features Selector*

Description

Identification of genes with ascending or descending monotonic expression patterns over time or stages of stem cells is an important issue in time-series microarray data analysis. We propose a method named Monotonic Feature Selector (MFSelector) based on a concept of total discriminating error (DE_{total}) to identify monotonic genes. MFSelector considers various time stages in stage order (i.e., Stage One vs. all other stages, Stages One and Two vs. remaining stages and so on) and computes DE_{total} of each gene. MFSelector can successfully identify genes with monotonic characteristics. We have

demonstrated the effectiveness of MFSelector on two stem cell differentiation datasets: embryonic stem cell neurogenesis (ESCN) and embryonic stem cell vasculogenesis (ESCV) datasets. Some of the monotonic marker genes such as *OCT4*, *NANOG*, *BLBP*, discovered from the ESCN dataset exhibit consistent behavior with that reported in other studies. The role of monotonic genes found by MFSelector in either stemness or differentiation is validated using information obtained from Gene Ontology analysis and other literature. We justify and demonstrate that descending genes are involved in the proliferation or self-renewal activity of stem cells, while ascending genes are involved in differentiation of stem cells into variant cell lineages.

Usage

```
mfselector(data, nsc, stageord = F, stagename = F, type = 1, nline
= T, dline = T, pdf = 1:100, cmp = 0, permut = 0, svdenoise = 0.03,
svdetimes = 0, cores = detectCores())
```

Arguments

`data`	A tab-delimited text file of a gene expression matrix with the first column containing IDs, such as probesets, gene symbol, etc.
`nsc`	An integer vector containing the corresponding sample numbers in each stage.
`stageord`	A vector describing which samples belong to the corresponding stages respectively in the gene expression matrix. Each element is in the (*S:E*) format indicating the start(*S*) and the end(*E*) of sample positions in a stage. The sample position in the matrix starts from 1. By default, the element values are determined by "`nsc`" from 1 to the last one in order.
`stagename`	A character vector containing the names of each stage in the gene expression matrix.
`type`	An integer indicating the type of mfselector will be performed: "1" is for the selection of monotonically *descending* genes and "2" is for the selection of monotonically *ascending* genes.
`nline`	Logical or the character "both". If "TRUE" (default), mfselector selects genes only with (*N*-1) distinct discriminating lines where *N* is the number of stages. If "FALSE", genes without *N*-1 distinct discriminating lines are selected. If "both", genes with all the possible number of discriminating lines (from 1 to *N*-1 distinct discriminating lines) are selected.
`dline`	Logical. If "TRUE" (default), the discriminating lines are shown on the scatter plots for each gene. If "FALSE", the discriminating lines are not shown on the scatter plots for each gene.
`pdf`	An integer or an integer vector. If `pdf` is an integer, *L*, the output will select the gene which is the L^{th} gene of the best monotonic genes; if `pdf` is an integer vector, (*L:M*), the output will select the genes which are the top L^{th}~M^{th} genes of the best monotonic genes. The default is the top hundred (1:100) of all genes with monotonic features. If `pdf` is the character "all", the output will select all genes. Those expression profiles of selected genes will be exhibited by scatterplots in the PDF file.

`cmp` An integer, *M*, determining that samples in the last *M* of stages are excluded from the computation of DE_{total}. The default value is 0. Note that samples in those stages excluded from the computation of DE_{total} will still be shown in the PDF file.

`permut` An integer, *M*, indicating mfselector will randomly permute the stage labels *M* times (default value of 0 means that do not calculate *p*- and *q*-value) for assessing the statistical significance of the DE_{total} index associated with the identified genes in ascending and descending characteristics. Note that, mfselector would spend more time (at least many hours) computing *p*-/*q*-values when `permut` is set higher.

`svdenoise` A decimal describing the strength of the noise in the experiment. For example, when `svdenoise` is "0.03" (3%), a random noise is generated in [-0.03, 0.03] and is added to the original expression value of a gene. This is done for all samples (See Details). Note that, this parameter is useless when `svdetimes` is set to "0".

`svdetimes` An integer indicating mfselector will repeat the SVDE procedure with random noise `svdenoise` in the same manner for `svdetimes` times (default: 0). (See Details) Note that, mfselector would spend more time (at least many hours) computing SVDE values when `svdetimes` is set higher.

`cores` The number of cores to use, i.e. how many processes will be spawned (default: at most).

Details

The case studies on ESCN and ESCV data sets have helped to get a better understanding of differentiation and stemness. This mfselector function creates a novel scheme to robustly identify gene sets with monotonic patterns in multiclass, time-series genomics matrices. No pre-existing hypothesis or knowledge is needed for gene filtration.

Computation of sample variance for discriminating error (SVDE):
Statistical tests are unable to distinguish which genes with the same level of statistical significance are better. Two genes with the same DE_{total} may exhibit different levels of monotonicity. Therefore, we develop a method to address the degree of monotonicity (a measure of confidence about the level of monotonicity) of a gene. This can also help to differentiate between genes with the same DE_{total} value.
In order to assess how strong the monotonicity of a gene is (particularly when more than one gene have the same DE_{total}), all samples for each of those genes are slightly altered in expression values to examine whether the DE_{total} of the altered expression values has changed significantly or not. To evaluate this, we propose an index, called Sample Variance for Discriminating Error (SVDE). We evaluate the extent of confidence on the monotonicity by adding random noise in [0, 1] to all samples (See full description in our publication: *MFSelector: Discovering monotonic stemness marker genes from time-series stem cell microarray data*)

Value

The message "`null device 1`" will be shown on the R screen after the mfselector command is successfully executed.

Two output files:

`Outputfile.txt` - A table of the selected genes which display gene ID, DE_{total} value, *p*-/*q*-value (if necessary), SVDE (if necessary), and the information indicating whether genes with *N*-1 distinct discriminating lines. An example of a TXT file name:

mfselector_2014-04-19_23.59.59.txt

`Outputfile.pdf` - A PDF file which contains the scatterplots of the top *N* monotonic genes which are determined by the parameter "`pdf`". An example of a PDF file name: mfselector_2014-04-19_23.59.59.pdf

See Also

multicore: Parallel processing of R code on machines with multiple cores or CPUs

Examples

```
### Two data sets (ESCN and ESCV) have been included in the package.
Users can directly load the data sets (note that, in this package,
the control probe sets 'AFFX-' have been already removed from the
two data sets):
>  library(MFSelector)
>  data(ESCN)
>  data(ESCV)
>  data(s50_Asc)


### User can also load the data set by the command as follows:
### > dataset <- read.table("dataset_name.txt", header = TRUE, sep
= "\t")


### Example 1: the ESC Neurogenesis data set (ESCN) matrix contains
5 stages "I ESC", "II EB", "III Day10+bFGF", "IV Day17+bFGF", "NSC".

>  mfselector(data = ESCN, nsc = c(3,3,6,6,9), stageord  = c(4:6,
1:3, 7:12, 13:18, 19:27), stagename = c("I ESC", "II EB", "III
Day10+bFGF", "IV Day17+bFGF", "NSC"), pdf = 1:200)


### Example 2: the Embryonic Stem Cell Vasculogenesis data set (ESCV)
matrix contains 4 stages "ESC", "MPC", "d14", "VEC".

>  mfselector(data = ESCV, nsc = c(3,3,4,3), stagename =
c("ESC","MPC","d14","B"))


### Example 3: Computation of p-/q-values for 50 permutations and
computation of SVDE value by adding three percent white noise to
each sample for 50 simulations on the ESCV data set. Note that, this
may take many hours when computing p-/q-values and SVDE value.

>  mfselector(data = ESCV, nsc = c(3,3,4,3), stagename =
c("ESC","MPC","d14","B"), permut = 50, svdenoise = 0.03, svdetimes
= 50)


### Example 4: The synthetic data set (s50_Asc) matrix contains 5
stages "stage1", "stage2", "stage3", "stage4", "stage5"

> mfselector(data = s50_Asc, nsc = c(10,10,10,10,10), stagename =
c("stage1","stage2","stage3","stage4","stage5"), type = 2)
```

```
### All the output files (PDF file and TXT file) are created in the
current working directory.
```