Package ‘KANT’

February 19, 2015

Type Package

Title Package to identify and sort genes overexpressed

Version 2.0

Date 2014-07-30

Author Noemie Robil

Maintainer Noemie Robil `<noemie.robil@ligue-cancer.net>`

Description Identify and sort genes overexpressed and associated to transmembrane protein in Affymetrix expression set or any other results of microarray experiment.

License GPL

Depends affy,Biobase

NeedsCompilation no

Repository CRAN

Date/Publication 2014-08-26 13:42:59

R topics documented:

`KANT-package` *KANT : Kancer Antigene Tracker*

Description

This algorithm was designed to identify and sort genes, associated to transmembrane protein, found to be more strongly expressed in a tumor dataset than in a normal dataset. It can be used in other conditions.

Details

Package:	KANT
Type:	Package
Version:	2.0
Date:	2014-07-30
License:	GPL

Author(s)

Noemie Robil

Maintainer: Who to complain to <noemie.robil@ligue-cancer.net>

References

KANT : a gene expression-based tool for detecting putative membrane cancer-specific antigens, not published

---

`expression_up` *Identify and sort genes overexpressed in one condition agains another in an affymetrix dataset or a matrix of data (genes per row, samples per column)*

Description

For each probeset of the dataset, the algorithm calculates the maximum level of expression in control samples (M) and select all test samples with and expression exceedings M + a threshold. A score is then attributed to the gene on the basis of the subpopulation selected size and overexpression.

Usage

```
expression_up(data,type="eset", CASE, CTRL,seuil=0.5)
```

Arguments

`data`	An affymetrix expressionset
`type`	Type of the data : "eset" if it is an affymetrix expressionSet with featureData; "tab" it it is a matrix of data (genes per row,samples per column)
`CASE`	index of the case samples in the dataset
`CTRL`	index of the control samples in the dataset
`seuil`	threshold to use Depending of the type of control, we usually choose 0.5 for an homogene control population and 0.2 for an heterogene control population

Value

An affymetrix expressionSet (or a matrix) containing results of the algorithm for every probesets : score of overexpression, number of samples in which the gene is overexpressed, maximum expression in control samples, delta (difference between the median of the subpopulation selected and the maximum of control samples), statistics about the population overexpressed (mean, median, standard deviation, interquartile range), names of samples in this subpopulation. This expressionSet (or matrix) in ranked by score in probesets.

Author(s)

Noemie Robil

Examples

```
library(Biobase)
library(affy)
data(sample.ExpressionSet)
TEST=which(pData(sample.ExpressionSet)[,"type"]=="Case")
CTRL=which(pData(sample.ExpressionSet)[,"type"]=="Control")
data_up=expression_up(sample.ExpressionSet,type="eset",TEST,CTRL,0.5)
```

`plot_up` *Function to plot the results of expression_up*

Description

Generates a graphic of the 50 best probesets of the algorithm. Every probeset is plotted in function of the number of samples in the subpopulation that overexpressed it and delta (difference of expression between max of normal samples and median of the subpopulation). They are represented by a circle, with a size proportional to 1/(standard deviation of expression in the subpopulation)

Usage

```
plot_up(N, eset_up, cells_type, fontsize)
```

Arguments

`N`	Number of probesets to plot
`eset_up`	Result of expression_up
`cells_type`	Name of the type of cells tested (just for title of the graph)
`fontsize`	For legends

Author(s)

Noemie Robil

See Also

See Also `link{expression_up}`

Examples

```
## The function is currently defined as
function (N, eset_up, cells_type, fontsize)
{
    probes <- 1:N
    plot(as.numeric(as.character(pData(featureData(eset_up))[probes,
        "Numbre_up"])), as.numeric(as.character(pData(featureData(eset_up))[probes,
        "Delta_median_up"])),
        cex = (1/(0.1 + as.numeric(as.character(pData(featureData(eset_up))[probes,
        "IQR_up"])))), pch = 21, main = c("Over-expression in  :",
      cells_type), xlab = "Number of samples in the sub-population over-expressing the gene",
        ylab = "Delta")
    text(as.numeric(as.character(pData(featureData(eset_up))[probes,
        "Numbre_up"])), as.numeric(as.character(pData(featureData(eset_up))[probes,
        "Delta_median_up"])), as.character(pData(featureData(eset_up))[probes,
        "Gene.Symbol"]), pos = 2, cex = fontsize)
  }
```

`TM` *List of geneid linked with transmembrane proteins*

Description

GeneID of genes coding for proteins identified as transmembrane by MEMSAT-SVM (modified version), among the Uniprot swissprot database from february 2014.

Usage

`data(TM)`

Format

A data frame with 4964 observations on the following variable.

Index