Package 'MMGeneSigIndex'

October 19, 2016

Title Gene Singature Index Estimation for Multiple Myeloma

Version 0.13-2

Date 2016-10-19

Author Tae-Hoon Chung

Maintainer Tae-Hoon Chung <hoontaechung@gmail.com>

Description This is a package to estimate gene expression signature indices from MAS5 processed expression matrices of multiple myeloma.

Depends RSQLite, gdata, PerformanceAnalytics, RColorBrewer, grid

Suggests

License GPL-2

R topics documented:

CellDeath 2
Centrosome 3
CIN 4
EMC92 6
fBox 7
fcategorize 8
fColorBar 9
fColorScheme 10
fGetDB 11
fHmap 12
fMultiplot 13
fpack 14
fPairs 14
fTemplate 15
GeneSigPack 16
HMCL 17
hovontest 18
IFM15 19
Proliferation 20
Transform 21
UAMS 22
Utilities 24

Index **26**

CellDeath *Homozygously Deleted Cell Death Signature Index*

Description

Functions to estimate HZDCD signature index based on MAS5-processed Affymetrix GeneChip data.

Usage

```
fHZDCD(x, val.type="in")
fSingleHZDCD(x, med, val.type="in")
```

Arguments

x	a numeric matrix (fHZDCD: gene expression matrix) or a numeric vector (fSingleHZDCD: gene expression vector).
med	a numeric vector (median expression values for probe(set)s).
val.type	either "intensity" or "logintensity" or "logratio" (value type of x)

Details

fHZDCD estimates prognostic signature indices of samples in an Affymetrix GeneChip gene expression matrix x. Preferably, the matrix is supposed to be MAS5-processed but this is not absolutely necessary as far as the row names bear original probesets while columns correspond to samples as usual.

fSingleHZDCD estimates a prognostic signature index for a named vector x which corresponds to an individual sample (column vector) of a gene expression matrix.

For implementation details, please refer to fCentroIndex.

Value

A numeric vector (per each sample/column) for fHZDCD.

A number for fSingleHZDCD.

Author(s)

Tae-Hoon Chung

References

Dickens NJ, Walker BA, Leone PE, *et al.* (2010) Homozygous deletion mapping in myeloma samples identifies genes and an expression signature relevant to pathogenesis and outcome. *Clin Cancer Res* **16:**1856–1864.

See Also

See also fGeneSigPack.

Examples

```
library(MMGeneSigIndex)
data(hovontest)

## .. case of using intensity data
gsig <- fHZDCD(hovontest)

## .. case of using logratio data
med <- apply(hovontest, 1, median)
logr <- log2(apply(hovontest, 2, function(x) x / med))
gsig2 <- fHZDCD(logr, val.type="logratio")

## .. case of using fSingleHZDCD
gsig3 <- apply(hovontest, 2, fSingleHZDCD, med)

## .. sanity check
plot(gsig, gsig2)
plot(gsig, gsig3)
```

`Centrosome` *Centrosome Index*

Description

Functions to estimate centrosome index based on MAS5-processed Affymetrix GeneChip data.

Usage

```
fCentroIndex(x, val.type="in")
fCI(x, val.type="in")
fSingleCentroIndex(x, med, val.type="in")
fSingleCI(x, med, val.type="in")
```

Arguments

`x`	a numeric matrix (`fCentroIndex`, `fCI`: gene expression matrix) or a numeric vector (`fSingleCentroIndex`, `fSingleCI`: gene expression vector).
`med`	median values for probesets.
`val.type`	either "intensity" or "logintensity" or "logratio" (value type of `x`).

Details

`fCentroIndex` and `fCI` estimate centrosome indices of samples in an Affymetrix GeneChip gene expression matrix `x`. Preferably, the matrix is supposed to be MAS5-processed but this is not absolutely necessary as far as the row names bear original probesets while columns correspond to samples as usual. `fSingleCentroIndex` and `fSingleCI` estimate centrosome indices for a named vector `x` which corresponds to an individual sample (a column) of a gene expression matrix.

Internally, estimation of centrosome index uses log of median-normalized gene expression matrix. So, when gene expression matrix is provided in intensity type (`val.type="intensity"`), we first estimate probeset-wise median values across the whole samples/columns, divide expression intensity of each sample by the estimated median, and take the base-2 logarithm. The procedure goes

exactly the same for logarithm type (`val.type="logintensity"`) except that division by median is substituted with subtraction by median. For this reason, users should provide probeset-wise median across dataset when using `fSingleCentroIndex` or `fSingleCI`.

Value

A numeric vector (per each sample/column) for `fCentroIndex` and `fCI`.

A number for `fSingleCentroIndex` and `fSingleCI`.

Author(s)

Tae-Hoon Chung

References

Chng W-J, Ahmann GJ, Henderson K, *et al.* (2006) Clinical implication of centrosome amplification in plasma cell neoplasm. *Blood* **107:**3669–3675.

Chng W-J, Braggio E, Mulligan G, *et al.* (2008) The centrosome index is a powerful prognostic marker in myeloma and identifies a cohort of patients that might benefit from aurora kinase inhibition. *Blood* **111:**1603–1609.

See Also

See also `fGeneSigPack`.

Examples

```
library(MMGeneSigIndex)
data(hovontest)

## .. case of using intensity data
gsig <- fCI(hovontest)

## .. case of using logratio data
med <- apply(hovontest, 1, median)
logr <- log2(apply(hovontest, 2, function(x) x / med))
gsig2 <- fCI(logr, val.type="logratio")

## .. case of using fSingleCI
gsig3 <- apply(hovontest, 2, fSingleCI, med)

## .. sanity check
plot(gsig, gsig2)
plot(gsig, gsig3)
```

CIN *Chromosomal Instability (CIN) Signature Index*

Description

Functions to estimate CIN signature index based on MAS5-processed Affymetrix GeneChip data.

Usage

```
fCINGEC(x, val.type="in")
fSingleCINGEC(x, med, val.type="in")
```

Arguments

x	a numeric matrix (fCINGEC: gene expression matrix) or a numeric vector (fSingleCINGEC: gene expression vector).
med	a numeric vector (median values for probe(set)s).
val.type	either "intensity" or "logintensity" or "logratio" (value type of x)

Details

fCINGEC estimates prognostic signature indices of samples in an Affymetrix GeneChip gene expression matrix x. Preferably, the matrix is supposed to be MAS5-processed but this is not absolutely necessary as far as the row names bear original probesets while columns correspond to samples as usual.

fSingleCINGEC estimates a prognostic signature index for a named vector x which corresponds to an individual sample (column vector) of a gene expression matrix.

For implementation details, please refer to fCentroIndex.

Value

A numeric vector (per each sample/column) for fCINGEC.

A number for fSingleCINGEC.

Author(s)

Tae-Hoon Chung

References

Chung TH, Mulligan G, Fonseca R, Chng WJ (2013) A novel measure of chromosome instability can account for prognostic difference in multiple myeloma. *PLoS ONE* **8:**e66361.

See Also

See also fGeneSigPack.

Examples

```
library(MMGeneSigIndex)
data(hovontest)

## .. case of using intensity data
gsig <- fCINGEC(hovontest)

## .. case of using logratio data
med <- apply(hovontest, 1, median)
logr <- log2(apply(hovontest, 2, function(x) x / med))
gsig2 <- fCINGEC(logr, val.type="logratio")

## .. case of using fSingleCINGEC
```

```
gsig3 <- apply(hovontest, 2, fSingleCINGEC, med)

## .. sanity check
plot(gsig, gsig2)
plot(gsig, gsig3)
```

EMC92 *EMC92 Signature Index*

Description

Functions to estimate EMC92 signature indices based on MAS5-processed Affymetrix GeneChip data.

Usage

```
fEMC92(x, val.type="in")
fSingleEMC92(x, med, val.type="in")
```

Arguments

x	a numeric matrix (fEMC92: gene expression matrix) or a numeric vector (fSingleEMC92: gene expression vector).
med	median values for probesets.
val.type	either "intensity" or "logintensity" or "logratio" (value type of x).

Details

fEMC92 estimates a prognostic signature index of samples in an Affymetrix GeneChip gene expression matrix x. Preferably, the matrix is supposed to be MAS5-processed but this is not absolutely necessary as far as the row names bear original probesets while columns correspond to samples as usual.

fSingleEMC92 estimates a prognostic signature index for a named vector x which corresponds to an individual sample (column vector) of a gene expression matrix.

For implementation details, please refer to fCentroIndex.

Value

A numeric vector (per each sample/column) for fEMC92.

A number for fSingleEMC92.

Author(s)

Tae-Hoon Chung

References

Kuiper R, Broyl A, de Knegt Y, *et al.* (2012) A gene expression signature for high-risk multiple myeloma. *Leukemia* **26:**2406–2413.

See Also

See also fGeneSigPack.

Examples

```
library(MMGeneSigIndex)
data(hovontest)

## .. case of using intensity data
gsig <- fEMC92(hovontest)

## .. case of using logratio data
med <- apply(hovontest, 1, median)
logr <- log2(apply(hovontest, 2, function(x) x / med))
gsig2 <- fEMC92(logr, val.type="logratio")

## .. case of using fSingleEMC92
gsig3 <- apply(hovontest, 2, fSingleEMC92, med)

## .. sanity check
plot(gsig, gsig2)
plot(gsig, gsig3)
```

---

fBox *Function to generate modified boxplot*

Description

This function generates a boxplot with individual data points overlaid to the group boxes.

Usage

```
fBox(x, ...)
```

Arguments

x	a list
...	limited graphical parameters

Details

Only a limited set of graphic parameters can be adjusted for this function. In particular, data point symbols, colors, and line widths cannot be adjusted in this function.

Author(s)

Tae-Hoon Chung

Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==>  Define data, use random,
##--or do  help(data=index)  for the standard data sets.

## The function is currently defined as
function (x)
{
  }
```

`fcategorize` *Function to categorize input vector into discrete levels*

Description

These functions categorize input vector into discrete levels.

Usage

```
fcategorize(x, nlev, by=c("equal.lev", "equal.quantile", "fixed.lev", "fixed.quantile", "fixed.ra
fcategorize2(x, nlev, by=c("equal.lev", "equal.quantile", "fixed.lev", "fixed.quantile", "fixed.r
fcategorize.rev(x)
```

Arguments

`x`	a numeric vector
`nlev`	a numeric vector
`by`	one of "equal.lev", "equal.quantile", "fixed.lev", "fixed.quantile", "fixed.range"

Details

If `by` is "equal.lev", input data is divided into `nlev` number of euqal intervals. If `by` is "equal.quantile", input data is divided into `nlev` number of equal quatiles. If `by` is "fixed.lev", input data is divided into discrete levels using the values in `nlev` as cutoff values. If `by` is "fixed.quantile", input data is divided into discrete levels using the values in `nlev` as cutoff quantile values. If `by` is "fixed.range", input data is divided into discrete levels at fixed ranges in `nlev`.

Value

`fcategorize` produces a numeric vector in $1, \ldots, nlev$ range where 1 represents the lowest value level and `nlev` represents the highest value level. `fcategorize2` produces a list with 2 elements, a numeric vector of level cutoff values and another numeric vector of discrete levels. `fcategorize.rev` reverses the discrete level indicators so that 1 represents the highest value level and `nlev` represents the lowest one.

Note

For cases where `by` is "equal.lev" or "equal.quantile", `nlev` should be integers > 1. For the case where `by` is "fixed.lev", `nlev` should be within the input data range. For other cases, `nlev` should be between 0 and 1.

Author(s)

Tae-Hoon Chung

Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==>  Define data, use random,
##--or do  help(data=index)  for the standard data sets.

## The function is currently defined as
x <- rnorm(100)
y <- fcategorize(x, 3, by="equal.lev")
table(y)

y <- fcategorize(x, 3, by="equal.quantile")
table(y)

y <- fcategorize(x, 0.5, by="fixed.lev")
table(y)

y <- fcategorize(x, 0.5, by="fixed.quantile")
table(y)

y <- fcategorize(x, 0.5, by="fixed.range")
table(y)

z <- fcategorize.rev(y)
cbind(y, z)[1:25]
```

---

fColorBar *Function to generate a color bar*

Description

This function generates a color bar used generally for heatmaps.

Usage

```
fColorBar(x, horizontal=TRUE, col=heat.colors(50), scale=1:length(x), k= 7, CEX=1, ...)
```

Arguments

x	numeric vector or character vector (colors)
horizontal	logical (draw color bar horizontally?)
col	character vector (colors to use)
scale	numeric vector (used only when x is character vector)
k	integer (# of steps in color bar)
CEX	numeric (cex.axis of numeric scale of input vector)
...	limited graphical parameters

Details

Only a very limited set of graphic parameters can be adjusted for this function. In particular, data point symbols, colors, and line widths cannot be adjusted in this function.

Author(s)

Tae-Hoon Chung

Examples

```
##---- Should be DIRECTLY executable !! ----
```

`fColorScheme` *Function to generate a vector of color rgb values*

Description

This function generates a set of colors used for plots.

Usage

```
fColorScheme(mode)
```

Arguments

`mode`	integer (between 1 and 10)

Value

The following colors are generated depending on `mode`.

mode	Minimum color	Middle color	Maximum color	Steps
1	green	black	red	50
2	green	white	red	50
3	red	black	green	50
4	red	white	green	50
5	white		red	25
6	white		green	25
7	black		red	25
8	black		green	25
9	black		white	25
10	white		black	25

Author(s)

Tae-Hoon Chung

Examples

```
##---- Should be DIRECTLY executable !! ----
```

fGetDB *Functions for SQLite Database Connection*

Description

These functions connect, create, and release SQLite database file.

Usage

```
fGetDB(name, dir, new=TRUE)
fReleaseDB(con)
```

Arguments

name	a string (sqlite database name with suffix .db)
dir	a string (location for database file)
new	a logical (is this database new one?)
con	database connection object

Details

fGetDB checks if the database is a new one. If so (new=TRUE), it will check the folder dir and proceed if the file doesn't exist there. If the database is not a new one and the file exists in the folder dir, it returns the connection. fReleaseDB closes the database connection and releases it.

Value

A database connection for fGetDB and fGetNewDB.

Note

If dir is empty, then fGetDB uses current folder as location.

Author(s)

Tae-Hoon Chung

Examples

```
##---- Should be DIRECTLY executable !! ----

## The function is currently defined as
```

fHmap *Function to generate a heatmap*

Description

This function generates a heatmap of a matrix by aligning rows to y-axis and columns to x-axis.

Usage

```
fHmap(z, Range=NULL, Grid=TRUE, Color=1, GridLine=list(Col="orange", Weight=1, GridType=1), RowLi
```

Arguments

z	a numeric matrix
Range	a 2-element numeric vector or NULL
Grid	a logical value
Color	a numeric value (1-10) or 2-/3-element character vector
GridLine	a list with "Col"", "Weight"", "GridType" elements
RowLine	a list with "Pos", "Col", "Weight", "Type" elements
ColLine	a list with "Pos", "Col", "Weight", "Type" elements
Mark	a list with lists of "R" and "C" elements

Details

Range sets the range of values in z to plot. If NULL, full range values will be used. Grid determines whether to put grid lines in the heatmap or not. If Grid is turned on, then GridLine determines the color (Col), line width (Weight), and line type (GridType). If horizontal or vertical lines are required, RowLine or ColLine can be used with Pos to specify the position to put those lines at. For row lines, the position is counted from the top not from the bottom as done in image function. If specific elements in the matrix z is to be highlighted, then Mark can be used where each element is represented as list("R"=10, "C"=4) for instance.

Value

A character vector of color specification will be returned to help color bar generation (using colorbar.plot function in fields package for instance).

Author(s)

Tae-Hoon Chung

Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==>  Define data, use random,
##--or do  help(data=index)  for the standard data sets.

## The function is currently defined as
function (z)
{
  }
```

`fMultiplot` *Function to generate a multiplot using* **ggplot2** *library*

Description

This is a function adapted from the **Cookbook for R** website

Usage

```
fMultiplot(..., plotlist=NULL, file, cols=1, layout=NULL)
```

Arguments

`...`	ggplot objects
`plotlist`	a list of ggplot objects
`file`	a character (one of "pearson", "kendall", and "spearman")
`cols`	a numeric (number of columns in final plot)
`layout`	a matrix (layout specification)

Details

ggplot objects can be passed in ..., or to plotlist (as a list of ggplot objects). If `layout` is present, `cols` is ignored. If the layout is something like matrix(c(1,2,3,3), nrow=2, byrow=TRUE), then plot 1 will go in the upper left, 2 will go in the upper right, and 3 will go all the way across the bottom as usual.

Author(s)

Tae-Hoon Chung

Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==>  Define data, use random,
##--or do  help(data=index)  for the standard data sets.

## The function is currently defined as
function (x)
{
  }
```

fpack *Function to concatenate strings without gaps*

Description

This function concatenates strings without gaps.

Usage

```
fpack(...)
```

Arguments

...	a character list or character vectors

Details

With a character list, this function generates a concatenated string. With character vectors, this function generates element-by-element concatenated string vector if lengths of them are the same. If one of the vector is of length 1, then the element will be concatenated to all elements of other vectors. Otherwise, it will stop working.

Author(s)

Tae-Hoon Chung

Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==>  Define data, use random,
##--or do  help(data=index)  for the standard data sets.

## The function is currently defined as
function (x)
{
  }
```

fPairs *Function to generate a modified pairs plot*

Description

This is a function adapted from the **PerformanceAnalytics** library

Usage

```
fPairs(R, histogram=TRUE, method=c("pearson", "kendall", "spearman"),
CEXCOR=1, fit="lin", ...)
```

Arguments

R	a matrix
histogram	a logical
method	a character (one of "pearson", "kendall", and "spearman")
CEXCOR	a numeric (correlation value size in pairs plot)
fit	a string (fitting method over scatter plot)
...	graphic parameters for pairs

Details

With a matrix R, this function generates a pairs plot using column vectors. In the upper triangular part, correlation values between two columns are displayed while, in the lower triangular part, corresponding scatter plots are displayed. Along the diagonal part, histograms of individual data vector will appear if histogram is turned on (TRUE). The correlation between two column vectors will be estimated by the method provided in method. If fit is "lin", linear regression lines will be overlaid on the scatter plots. Otherwise, lowess regression lines will be drawn.

Note

In the lower triangular part, straight lines are plotted on top of scatter plots while the version in **PerformanceAnalytics** library used smooth regression lines.

Author(s)

Tae-Hoon Chung

Examples

```
## Not run: fPairs(cbind("X"=rnorm(20), "Y"=rbinom(20), "Z"=rpois(20)))
```

---

fTemplate *Simple Templating Function*

Description

This function generates a text string from a template with keys replaced with user inputs.

Usage

```
fTemplate(txt, ...)
```

Arguments

txt	a template string
...	key=value matched string

Author(s)

Tae-Hoon Chung

Examples

```
txt <- "Select '%field%' from '%table%'"
sql <- fTemplate(txt, field="name", table="employee"); sql
```

GeneSigPack *Multiple Myeloma Prognostic Signature Indices by Batch*

Description

Functions to estimate a group of multiple myeloma prognostic signature indices en masse based on MAS5-processed Affymetrix GeneChip data.

Usage

```
fGeneSigPack(x, val.type="in")
fGeneSigPack2(x, pack, val.type="in")
```

Arguments

x	a numeric matrix (gene expression matrix)
pack	named list of gene signatures (either 2-column data frame or a simple string vector)
val.type	either "intensity" or "logintensity" or "logratio" (value type of x).

Details

fGeneSigPack can be used for 9 fixed prognostic signatures "CINGEC", "CNTI", "EMC92", "HMCL7", "HZDCD", "IFM15", "PI", "UAMS70", "UAMS80" while fGeneSigPack2 can be used for arbitrary number of custom-made signature indices. The signatures for fGeneSigPack2 are provided through pack which expects a named list where names would be used as the signature identity and values as probe(set)s (and possibly weights) for signature estimation. If a signature is composed of probe(set)s with positive and negative weights that correspond to over- or under-expressed genes, then the signature list item would be a two-column dataframe with probe(set)s in the first column while weights in the second column. If a signature is composed of probe(set)s with all positive weights, then the signature list item would be simple vector of probe(set)s. For a concrete example, please look at the PROBE object.

Value

A matrix with rows corresponing to signature indices while columns corresponding to samples.

Author(s)

Tae-Hoon Chung

References

Chung TH, Mulligan G, Fonseca R, Chng WJ (2013) A novel measure of chromosome instability can account for prognostic difference in multiple myeloma. *PLoS ONE* **8:**e66361.

Examples

```
library(MMGeneSigIndex)

data(hovontest)

## .. case of using intensity data
gsig <- fGeneSigPack(hovontest)

## .. case of using logratio data
med <- apply(hovontest, 1, median)
logr <- log2(apply(hovontest, 2, function(x) x / med))
gsig2 <- fGeneSigPack(logr, val.type="logratio")

## .. case of using fGeneSigPack2
mysig <- data.frame('PROBE'=rownames(hovontest)[1:10], 'WEIGHT'=sample(c(-1,1), 10, replace=TRUE))
gsig3 <- fGeneSigPack2(hovontest, list("mysig"=mysig))

## .. sanity check
plot(gsig[,1], gsig2[,1])
plot(gsig[1,], gsig3[1,])
```

HMCL *Signature Index from IL6 Dependent MM Cell Line*

Description

Functions to estimate HMCL7 signature index based on MAS5-processed Affymetrix GeneChip data.

Usage

```
fHMCL7(x, val.type="in")
fHMCL248(x, val.type="in")
fSingleHMCL7(x, med, val.type="in")
fSingleHMCL248(x, med, val.type="in")
```

Arguments

x	a numeric matrix (fHMCL7, fHMCL248: gene expression matrix) or a numeric vector (fSingleIFM15, fSingleHCML248: gene expression vector).
med	a numeric vector (median values for probe(set)s).
val.type	either "intensity" or "logintensity" or "logratio" (value type of x).

Details

fHMCL7 and fHMCL248 estimate prognostic signature indices of samples in an Affymetrix GeneChip gene expression matrix x. Preferably, the matrix is supposed to be MAS5-processed but this is not absolutely necessary as far as the row names bear original probesets while columns correspond to samples as usual.

fSingleHMCL7 and fHMCL248 estimate prognostic signature indices for a named vector x which corresponds to an individual sample (column vector) of a gene expression matrix.

For implementation details, please refer to fCentroIndex.

Value

A numeric vector (per each sample/column) for `fHMCL7` and `fHMCL248`.

A number for `fSingleHMCL7` and `fSingleHCML248`.

Author(s)

Tae-Hoon Chung

References

Moreaux J, Klein B, Bataille R, *et al.* (2011) A high-risk signature for patients with multiple myeloma established from the molecular classification of human myeloma cell lines. *Haematologica* **96:**574–582.

See Also

See also `fGeneSigPack`.

Examples

```
library(MMGeneSigIndex)
data(hovontest)

## .. case of using intensity data
gsig <- fHMCL7(hovontest)

## .. case of using logratio data
med <- apply(hovontest, 1, median)
logr <- log2(apply(hovontest, 2, function(x) x / med))
gsig2 <- fHMCL7(logr, val.type="logratio")

## .. case of using fSingleHMCL7
gsig3 <- apply(hovontest, 2, fSingleHMCL7, med)

## .. sanity check
plot(gsig, gsig2)
plot(gsig, gsig3)
```

---

hovontest *Truncated HOVON Data for Simple Testing*

Description

This is a truncated HOVON data for testing purposes of functions.

Usage

```
data(hovontest)
```

Format

A numeric matrix

Examples

```
data(hovontest)
```

IFM15 *Signature Index from Intergroupe Francophone du Myelome study*

Description

Functions to estimate IFM15 signature index based on MAS5-processed Affymetrix GeneChip data.

Usage

```
fIFM15(x, val.type="in")
fSingleIFM15(x, med, val.type="in")
```

Arguments

x	a numeric matrix (fIFM15: gene expression matrix) or a numeric vector (fSingleIFM15: gene expression vector).
med	a numeric vector (median values for probe(set)s).
val.type	either "intensity" or "logintensity" or "logratio" (value type of x).

Details

fIFM15 estimates a prognostic signature index of samples in an Affymetrix GeneChip gene expression matrix x. Preferably, the matrix is supposed to be MAS5-processed but this is not absolutely necessary as far as the row names bear original probesets while columns correspond to samples as usual.

fSingleIFM15 estimates a prognostic signature index for a named vector x which corresponds to an individual sample (column vector) of a gene expression matrix.

For implementation details, please refer to fCentroIndex.

Value

A numeric vector (per each sample/column) for fIFM15.

A number for fSingleIFM15.

Author(s)

Tae-Hoon Chung

References

Decaux O, Lode L, Magrangeas F, *et al.* (2008) Prediction of survival in multiple myeloma based on gene expression profiles reveals cell cycle and chromosomal instability signatures in high-risk patients and hyperdiploid signatures in low-risk patients: a study of the Intergroupe Francophone du Myelome. *J Clin Oncol* **26:**4798–4805.

See Also

See also fGeneSigPack.

Examples

```
library(MMGeneSigIndex)
data(hovontest)

## .. case of using intensity data
gsig <- fIFM15(hovontest)

## .. case of using logratio data
med <- apply(hovontest, 1, median)
logr <- log2(apply(hovontest, 2, function(x) x / med))
gsig2 <- fIFM15(logr, val.type="logratio")

## .. case of using fSingleIFM15
gsig3 <- apply(hovontest, 2, fSingleIFM15, med)

## .. sanity check
plot(gsig, gsig2)
plot(gsig, gsig3)
```

Proliferation *Proliferation Index*

Description

Functions to estimate proliferation index based on MAS5-processed Affymetrix GeneChip data.

Usage

```
fProlifIndex(x, val.type="in")
fPI(x, val.type="in")
fSingleProlifIndex(x, med, val.type="in")
fSinglePI(x, med, val.type="in")
```

Arguments

x	a numeric matrix (fProlifIndex, fPI: gene expression matrix) or a numeric vector (fSingleProlifIndex, fSinglePI: gene expression vector).
med	a numeric vector (median values for probe(set)s).
val.type	either "intensity" or "logintensity" or "logratio" (value type of x).

Details

fProlifIndex and fPI estimate proliferation indices of samples in an Affymetrix GeneChip gene expression matrix x. Preferably, the matrix is supposed to be MAS5-processed but this is not absolutely necessary as far as the row names bear original probesets while columns correspond to samples as usual.

fSingleProlifIndex and fSinglePI estimate proliferation indices for a named vector x which corresponds to an individual sample (a column) of a gene expression matrix.

For implementation details, please refer to fCentroIndex.

Value

A numeric vector (per each sample/column) for `fProlifIndex` and `fPI`.

A number for `fSingleProlifIndex` and `fSinglePI`.

Author(s)

Tae-Hoon Chung

References

Hose D, R\'eme T, Hielscher T, *et al.* (2011) Proliferation is a central independent prognostic factor and target for personalized and risk-adapted treatment in multiple myeloma. *Haematologica* **96:**87-95.

See Also

See also `fGeneSigPack`.

Examples

```
library(MMGeneSigIndex)

data(hovontest)

## .. case of using intensity data
gsig <- fPI(hovontest)

## .. case of using logratio data
med <- apply(hovontest, 1, median)
logr <- log2(apply(hovontest, 2, function(x) x / med))
gsig2 <- fPI(logr, val.type="logratio")

## .. case of using fSinglePI
gsig3 <- apply(hovontest, 2, fSinglePI, med)

## .. sanity check
plot(gsig, gsig2)
plot(gsig, gsig3)
```

`Transform` *Function to transform values into specific distribution*

Description

These functions transform the distribution of input values into specific forms.

Usage

```
fNormalTransform(x, avg, std)
fStandardize(x, avg, std)
```

Arguments

x	a numeric vector
avg	a numeric (targeted average)
std	a numeric (targeted standard deviation)

Details

fNormalTransform transforms input data into a new data whose distribution follows the normal distribution with mean avg and standard deviation std. fStandardize is a z-score transformation with resulting mean avg and standard deviation std.

Author(s)

Tae-Hoon Chung

Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==>  Define data, use random,
##--or do  help(data=index)  for the standard data sets.

## The function is currently defined as
function (x)
{
  }
```

UAMS *University of Arkansas Medical School (UAMS) Signature Indices*

Description

Functions to estimate UAMS-devised indices based on MAS5-processed Affymetrix GeneChip data.

Usage

```
fUAMS70(x, val.type="in")
fUAMS80(x, val.type="in")
fSingleUAMS70(x, med, val.type="in")
fSingleUAMS80(x, med, val.type="in")
```

Arguments

x	a numeric matrix (fUAMS70, fUAMS80: gene expression matrix) or a numeric vector (fSingleUAMS70, fSingleUAMS80: gene expression vector).
med	a numeric vector (median values for probe(set)s).
val.type	either "intensity" or "logintensity" or "logratio" (value type of x).

Details

fUAMS70 and fUAMS80 estimate prognostic signature indices of samples in an Affymetrix GeneChip gene expression matrix x designed by researchers in University of Arkansas Medical School (UAMS). Preferably, the matrix is supposed to be MAS5-processed but this is not absolutely necessary as far as the row names bear original probesets while columns correspond to samples as usual.

fSingleUAMS70 and fSingleUAMS80 estimate UAMS prognostic signature indices for a named vector x which corresponds to an individual sample (a column) of a gene expression matrix.

For implementation details, please refer to fCentroIndex.

Value

A numeric vector (per each sample/column) for fUAMS70 and fUAMS80.

A number for fSingleUAMS70 and fSingleUAMS80.

Author(s)

Tae-Hoon Chung

References

Shaughnessy JD, Zhan F, Burington BE, *et al.* (2006) A validated gene expression model of high-risk multiple myeloma is defined by deregulated expression of genes mapping to chromosome 1. *Blood* **109:**2276–2284.

Shaughnessy JD, Qu P, Usmani S, *et al.* (2011) Pharmacogenomics of bortezomib test-dosing identifies hyperexpression of proteasome genes, especially PSMD4, as novel high-risk feature in myeloma treated with Total Therapy 3. *Blood* **118:**3512-3524.

See Also

See also fGeneSigPack.

Examples

```
library(MMGeneSigIndex)
data(hovontest)

## .. case of using intensity data
gsig <- fUAMS70(hovontest)

## .. case of using logratio data
med <- apply(hovontest, 1, median)
logr <- log2(apply(hovontest, 2, function(x) x / med))
gsig2 <- fUAMS70(logr, val.type="logratio")

## .. case of using fSingleEMC92
gsig3 <- apply(hovontest, 2, fSingleUAMS70, med)

## .. sanity check
plot(gsig, gsig2)
plot(gsig, gsig3)
```

Utilities *Utility Functions for Signature Index Estimation*

Description

Basic functions for estimating gene expression signature indices with a gene expression matrix.

Usage

```
findex.mat(x, prb, val.type="in", alg="med", normalize=T, weight=T)
findex.mat.bare.bare(x, prb, val.type="in", alg="med")
findex.mat.bare.weight(x, prb, val.type="in", alg="med")
findex.mat.norm.bare(x, prb, val.type="in", alg="med")
findex.mat.norm.weight(x, prb, val.type="in", alg="med")
```

Arguments

x	a numeric matrix (gene expression matrix)
prb	a two-column data frame (column 1: probe(set) ids, column 2: weights) for findex.mat and findex.*.weight; a string vector (of probe(set) ids) for findex.mat with weight=F and findex.*.bare
val.type	either "intensity" (default) or "logintensity" or "logratio"; value type of x
alg	either "median" (default) or "mean"; signature index estimation, median or mean?
normalize	a logical (perform median-normalization of x? default: TRUE)
weight	a logical (signature contains positive- and negative-weights (TRUE) or just positive-weight ones (FALSE)? default: TRUE)

Details

When a gene expression matrix and a gene signature are prepared, findex.mat is the basic function for estimating the signature index. The default option setting is recommended for a MAS5 processed Affymetrix GeneChip expression matrix with a signature of over- as well as under-expressed probeset members. If the signature contains only over-expressed probesets instead, then the option would be changed as weight=F or, if the gene expression matrix is prepared with RMA algorithm instead, then the option would be changed as val.type="logintensity" with all other options untouched.

Value

a numeric vector

Author(s)

Tae-Hoon Chung

References

Chung TH, Mulligan G, Fonseca R, Chng WJ (2013) A novel measure of chromosome instability can account for prognostic difference in multiple myeloma. *PLoS ONE* **8:**e66361.

Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==>  Define data, use random,
##--or do  help(data=index)  for the standard data sets.

## The function is currently defined as
```

Index

∗Topic **datasets**
hovontest, 18
∗Topic **math**
CellDeath, 2
Centrosome, 3
CIN, 4
EMC92, 6
GeneSigPack, 16
HMCL, 17
IFM15, 19
Proliferation, 20
UAMS, 22
Utilities, 24
∗Topic **plot**
fBox, 7
fColorBar, 9
fColorScheme, 10
fHmap, 12
fMultiplot, 13
fPairs, 14
∗Topic **utility**
fBox, 7
fcategorize, 8
fColorBar, 9
fColorScheme, 10
fGetDB, 11
fHmap, 12
fMultiplot, 13
fpack, 14
fPairs, 14
fTemplate, 15
Transform, 21
Utilities, 24

CellDeath, 2
Centrosome, 3
CIN, 4

EMC92, 6

fBox, 7
fcategorize, 8
fcategorize2 *(*fcategorize*)*, 8
fCentroIndex, *2*, *5*, *6*, *17*, *19*, *20*, *23*
fCentroIndex *(*Centrosome*)*, 3
fCI *(*Centrosome*)*, 3
fCINGEC *(*CIN*)*, 4
fColorBar, 9
fColorScheme, 10
fEMC92 *(*EMC92*)*, 6
fGeneSigPack, *2*, *4*, *5*, *7*, *18*, *19*, *21*, *23*
fGeneSigPack *(*GeneSigPack*)*, 16
fGeneSigPack2 *(*GeneSigPack*)*, 16
fGetDB, 11
fHmap, 12
fHMCL248 *(*HMCL*)*, 17
fHMCL7 *(*HMCL*)*, 17
fHZDCD *(*CellDeath*)*, 2
fIFM15 *(*IFM15*)*, 19
findex.mat *(*Utilities*)*, 24
fMultiplot, 13
fNormalTransform *(*Transform*)*, 21
fpack, 14
fPairs, 14
fPI *(*Proliferation*)*, 20
fProlifIndex *(*Proliferation*)*, 20
fReleaseDB *(*fGetDB*)*, 11
fSingleCentroIndex *(*Centrosome*)*, 3
fSingleCI *(*Centrosome*)*, 3
fSingleCINGEC *(*CIN*)*, 4
fSingleEMC92 *(*EMC92*)*, 6
fSingleHMCL248 *(*HMCL*)*, 17
fSingleHMCL7 *(*HMCL*)*, 17
fSingleHZDCD *(*CellDeath*)*, 2
fSingleIFM15 *(*IFM15*)*, 19
fSinglePI *(*Proliferation*)*, 20
fSingleProlifIndex *(*Proliferation*)*, 20
fSingleUAMS70 *(*UAMS*)*, 22
fSingleUAMS80 *(*UAMS*)*, 22
fStandardize *(*Transform*)*, 21
fTemplate, 15
fUAMS70 *(*UAMS*)*, 22
fUAMS80 *(*UAMS*)*, 22

GeneSigPack, 16

HMCL, 17
hovontest, 18

IFM15, 19

Proliferation, 20

Transform, 21

UAMS, 22
Utilities, 24