Package 'transcriptogramer'

April 12, 2018

Type Package

Title Transcriptional analysis based on transcriptograms

Version 1.0.0

Date 2017-10-23

Description R package for transcriptional analysis based on transcriptograms, a method to analyze transcriptomes that projects expression values on a set of ordered proteins, arranged such that the probability that gene products participate in the same metabolic pathway exponentially decreases with the increase of the distance between two proteins of the ordering. Transcriptograms are, hence, genome wide gene expression profiles that provide a global view for the cellular metabolism, while indicating gene sets whose expression are altered.

Depends R (>= 3.4), methods

License GPL (>= 2)

Encoding UTF-8

LazyData true

biocViews Software, Network, Visualization, SystemsBiology, GeneExpression, GenomeWideAssociation, GeneSetEnrichment

Imports biomaRt, data.table, doSNOW, foreach, graphics, grDevices, igraph, limma, parallel, progress, RedeR, snow, stats, topGO

RoxygenNote 6.0.1

VignetteBuilder knitr

Suggests BiocStyle, knitr, rmarkdown, RUnit, BiocGenerics

SystemRequirements Java Runtime Environment (>= 6)

NeedsCompilation no

Author Diego Morais [aut, cre],
Rodrigo Dalmolin [aut]

Maintainer Diego Morais <arthur.vinx@gmail.com>

R topics documented:

1

association	*Association*

Description

A subset of the Homo sapiens protein network data from STRINGdb, release 10.5. This subset contains only associations of proteins of combined score greater than or equal to 900.

Usage

```
association
```

Format

Each row of the data frame contains two variables:

V1 The ENSEMBL Peptide ID of the first protein

V2 The ENSEMBL Peptide ID of the second protein

Author(s)

Diego Morais

See Also

[Hs900](#)

Examples

```
association
```

---

clusterEnrichment   *Term enrichment*

Description

If `species` is a character, this method uses the **biomaRt** package to build a gene2GO list, if `species` is a data frame, it will be used instead. The gene2GO list will be used with the **topGO** package to detect the most significant terms of each cluster present in the DE slot of the `object`.

Usage

```
clusterEnrichment(object, universe = NULL, species,
  ontology = "biological process", algorithm = "classic",
  statistic = "fisher", pValue = 0.05, adjustMethod = "BH", nCores = 1L)

## S4 method for signature 'Transcriptogram'
clusterEnrichment(object, universe = NULL,
  species, ontology = "biological process", algorithm = "classic",
  statistic = "fisher", pValue = 0.05, adjustMethod = "BH", nCores = 1L)
```

Arguments

object	An object of class Transcriptogram
universe	A character vector containing ENSEMBL Peptide IDs, or NULL, if the universe is composed by all the proteins present in the transcriptogramS2 slot of `object`
species	A character string specifying the species; or a data frame containing two columns, the first one with ENSEMBL Peptide IDs (character), which may, or not, to contain the taxonomy ID of the species as prefix, and the second containing its respective Gene Ontology term (character)
ontology	A character string specifying the Gene Ontology domain, ignoring case sensitivity, the possible values are 'biological process', 'cellular component' and 'molecular function', the default value for this argument is 'biological process'
algorithm	Character string specifying which algorithm to use, the possible values are 'classic', 'elim', 'weight', 'weight01', 'lea' and 'parentchild', the default value for this argument is 'classic'
statistic	Character string specifying which test to use, the possible values are 'fisher', 'ks', 't', 'sum' and 'globaltest', the default value for this argument is 'fisher'
pValue	A numeric value between 0 and 1 giving the required family-wise error rate or false discovery rate, the default value is 0.05
adjustMethod	Character string specifying p-value adjustment method, the possible values are 'none', 'BH', 'fdr' (equivalent to 'BH'), 'BY', 'hochberg', 'hommel', 'bonferroni', and 'holm', the default value for this argument is 'BH'

nCores   An integer number, referring to the number of processing cores to be used; or
                a logical value, TRUE indicating that all processing cores should be used, and
                FALSE indicating the use of just one processing core. The default value of this
                argument is 1.

Value

A data frame containing the most significant terms of each cluster

Author(s)

Diego Morais

See Also

differentiallyExpressed-method, transcriptogramPreprocess, GSE9988, GPL570, Hs900, HsBPTerms, association, transcriptogramStep1-method, transcriptogramStep2-method

Examples

```
transcriptogram <- transcriptogramPreprocess(association, Hs900, 50)
## Not run:
transcriptogram <- transcriptogramStep1(transcriptogram, GSE9988, GPL570)
transcriptogram <- transcriptogramStep2(transcriptogram)
levels <- c(rep(FALSE, 3), rep(TRUE, 3))
transcriptogram <- differentiallyExpressed(transcriptogram, levels, 0.005)
terms <- clusterEnrichment(transcriptogram, species = "Homo sapiens",
pValue = 0.005)

## this call also works
terms <- clusterEnrichment(transcriptogram, species = HsBPTerms,
pValue = 0.005)

## End(Not run)
```

---

clusterVisualization   *Displays graphs of the differentially expressed clusters*

Description

This method uses the **RedeR** package to display graphs of the differentially expressed clusters. If
the DE slot has a column named Symbol, its contents will be used as node alias.

Usage

```
clusterVisualization(object, maincomp = FALSE, connected = FALSE,
  host = "127.0.0.1", port = 9091, clusters = NULL)

## S4 method for signature 'Transcriptogram'
clusterVisualization(object, maincomp = FALSE,
  connected = FALSE, host = "127.0.0.1", port = 9091, clusters = NULL)
```

Arguments

`object`	An object of class Transcriptogram
`maincomp`	Logical value, whether to display only the main component of each cluster
`connected`	Logical value, whether to display only connected nodes
`host`	The domain name of the machine that is running the RedeR XML-RPC server
`port`	An integer specifying the port on which the XML-RPC server should listen
`clusters`	An integer vector specifying the clusters to be displayed, if NULL, all clusters will be displayed

Details

RedeR package requirements: Java Runtime Environment (>= 6)

Value

This function returns an object of the RedPort Class

Author(s)

Diego Morais

See Also

differentiallyExpressed-method, transcriptogramPreprocess, GSE9988, GPL570, Hs900, association, transcriptogramStep1-method, transcriptogramStep2-method, RedPort

Examples

```
transcriptogram <- transcriptogramPreprocess(association, Hs900, 50)
## Not run:
transcriptogram <- transcriptogramStep1(transcriptogram, GSE9988, GPL570)
transcriptogram <- transcriptogramStep2(transcriptogram)
levels <- c(rep(FALSE, 3), rep(TRUE, 3))
transcriptogram <- differentiallyExpressed(transcriptogram, levels, 0.005,
DEsymbols)
rdp <- clusterVisualization(transcriptogram)

## End(Not run)
```

---

connectivityProperties   *Calculates average graph properties as a function of the node connectivity*

Description

Calculates protein (node) connectivity/degree (k) and network properties as a function of this connectivity, such as: probability of a protein of the graph has degree k, average assortativity of the nodes of degree k, and the average clustering coefficient of the nodes of degree k.

Usage

```
connectivityProperties(object)

## S4 method for signature 'Transcriptogram'
connectivityProperties(object)
```

Arguments

object   An object of class Transcriptogram

Details

The assortativity of a node can be measured by the average degree of its neighbors

Value

This method returns a data frame containing: unique degrees (k) of the nodes of the graph, probability (pk) of a node of the graph has degree k, average assortativity (ak) of the nodes of degree k, and the average clustering coefficient (ck) of the nodes of degree k

Author(s)

Diego Morais

See Also

transcriptogramPreprocess, Hs900, association

Examples

```
transcriptogram <- transcriptogramPreprocess(association, Hs900)
## Not run:
cProperties <- connectivityProperties(transcriptogram)

## End(Not run)
```

DE   *Get DE*

Description

Gets the content of the DE slot of an object of class Transcriptogram.

Usage

```
DE(object)

## S4 method for signature 'Transcriptogram'
DE(object)
```

Arguments

object   An object of class Transcriptogram

Value

This method returns the content of the DE slot of an object of class Transcriptogram.

Author(s)

Diego Morais

See Also

[Hs900](#), [association](#), [transcriptogramPreprocess](#)

Examples

```
transcriptogram <- transcriptogramPreprocess(association, Hs900, 50)
DE(transcriptogram)
```

DEsymbols   *Dictionary Gene2Symbol*

Description

A mapping between ENSEMBL Peptide ID and Symbol (Gene Name) of a reduced set of proteins.

Usage

```
DEsymbols
```

Format

An object of class data.frame with 129 rows and 2 columns.

Author(s)

Diego Morais

Examples

```
DEsymbols
```

---

differentiallyExpressed

Identify which genes are differentially expressed

Description

This method uses the **limma** package to identify which genes are differentially expressed, meeting the `pValue` requirement, for the contrast "case-control". The `levels` lenght must be equal to the number of samples present in the transcriptogramS2 slot of the `object`, and its contents is related to the order that the samples appear. FALSE must be used to indicate case samples, and TRUE to indicate control samples. If `species` is NULL, no translation will be done, if `species` is a character, the **biomaRt** package will be used to translate the ENSEMBL Peptide ID to Symbol (Gene Name), and if `species` is a data frame, it will be used instead. If the translation fail for some protein, its ENSEMBL Peptide ID will be present into the Symbol column. This method also groups the proteins detected as differentially expressed in clusters, and plots a graphical representation of the groupings.

Usage

```
differentiallyExpressed(object, levels, pValue = 0.05, species = NULL,
  adjustMethod = "BH")

## S4 method for signature 'Transcriptogram'
differentiallyExpressed(object, levels,
  pValue = 0.05, species = NULL, adjustMethod = "BH")
```

Arguments

`object`	An object of class Transcriptogram
`levels`	A logical vector that classify the columns, referring to samples, of the transcriptogramS2 slot of the `object`
`pValue`	A numeric value between 0 and 1 giving the required family-wise error rate or false discovery rate, the default value is 0.05
`species`	If not NULL, a character string that will be used, ignoring case sensitivity, to translate the ENSEMBL Peptide ID to Symbol (Gene Name); or a data frame containing two columns, the first one with ENSEMBL Peptide IDs (character), which may, or not, to contain the taxonomy ID of the species as prefix, and the second containing its respective Symbol (character)
`adjustMethod`	Character string specifying p-value adjustment method, the possible values are 'none', 'BH', 'fdr' (equivalent to 'BH'), 'BY' and 'holm', the default value for this argument is 'BH'

Value

This method creates a data frame to feed the DE slot of an object of class Transcriptogram. This data frame of differentially expressed proteins contains the log Fold Change, the p-values and an integer number that indicates if the protein is downregulated or upregulated.

Author(s)

Diego Morais

See Also

transcriptogramPreprocess, GSE9988, GPL570, Hs900, association, DEsymbols, transcriptogramStep1-method, transcriptogramStep2-method

Examples

```
transcriptogram <- transcriptogramPreprocess(association, Hs900, 50)
## Not run:
transcriptogram <- transcriptogramStep1(transcriptogram, GSE9988, GPL570)
transcriptogram <- transcriptogramStep2(transcriptogram)
levels <- c(rep(FALSE, 3), rep(TRUE, 3))
transcriptogram <- differentiallyExpressed(transcriptogram, levels, 0.005)

## translating ENSEMBL Peptide IDs to Symbols
transcriptogram <- differentiallyExpressed(transcriptogram, levels, 0.005,
"Homo sapiens")

## these calls also works
transcriptogram <- differentiallyExpressed(transcriptogram, levels, 0.005,
"H sapiens")

transcriptogram <- differentiallyExpressed(transcriptogram, levels, 0.005,
DEsymbols)

## End(Not run)
```

GPL570   *Dictionary Gene2Probe*

Description

A mapping between ENSEMBL Peptide ID and probe identifier, for the Homo sapiens and the platform GPL570, [HG-U133_Plus_2] Affymetrix Human Genome U133 Plus 2.0 Array.

Usage

```
GPL570
```

Format

Each row of the data frame contains two variables:

ENSP The ENSEMBL Peptide ID

Probe The probe identifier

Author(s)

Diego Morais

See Also

GSE9988

Examples

```
GPL570
```

GSE9988   *Data frame containing expression values*

Description

Expression values, obtained by microarray, of 3 cases and 3 controls referring to the innate immune responses to TREM-1 activation. The data frame has 6 columns, each one contains expression values of a sample, the first 3 columns are case samples, and the last 3 are control samples. Each row contain expression values obtained by the probe mentioned in its respective rowname. The expression values were normalized using the affy package and, to reduce the storage space required for the data, this data frame is a subset from the original samples (GSM252443, GSM252444, GSM252445, GSM252465, GSM252466, GSM252467), containing only the rows on which the probes are mapped by the platform GPL570 dictionary.

Usage

```
GSE9988
```

Format

An object of class `data.frame` with 32754 rows and 6 columns.

Author(s)

Diego Morais

Source

[GSE9988](#)

See Also

[GPL570](#)

Examples

```
GSE9988
```

Hs700 *Ordered Homo sapiens proteins of combined score greater than or equal to 700*

Description

A character vector containing the Homo sapiens proteins, from STRINGdb release 10.5, of combined score greater than or equal to 700.

Usage

```
Hs700
```

Format

An object of class `character` of length 15154.

Author(s)

Diego Morais

Examples

```
Hs700
```

Hs800 *Ordered Homo sapiens proteins of combined score greater than or equal to 800*

Description

A character vector containing the Homo sapiens proteins, from STRINGdb release 10.5, of combined score greater than or equal to 800.

Usage

```
Hs800
```

Format

An object of class `character` of length 13273.

Author(s)

Diego Morais

Examples

```
Hs800
```

Hs900	*Ordered Homo sapiens proteins of combined score greater than or equal to 900*

Description

A character vector containing the Homo sapiens proteins, from STRINGdb release 10.5, of combined score greater than or equal to 900.

Usage

```
Hs900
```

Format

An object of class `character` of length 11030.

Author(s)

Diego Morais

Examples

```
Hs900
```

HsBPTerms	*Dictionary Gene2GO*

Description

A mapping between ENSEMBL Peptide ID and Gene Ontology, biological process, terms of a set of proteins.

Usage

```
HsBPTerms
```

Format

An object of class `data.frame` with 107897 rows and 2 columns.

Author(s)

Diego Morais

Examples

```
HsBPTerms
```

Mm700	*Ordered Mus musculus proteins of combined score greater than or equal to 700*

Description

A character vector containing the Mus musculus proteins, from STRINGdb release 10.5, of combined score greater than or equal to 700.

Usage

```
Mm700
```

Format

An object of class character of length 13921.

Author(s)

Diego Morais

Examples

```
Mm700
```

Mm800	*Ordered Mus musculus proteins of combined score greater than or equal to 800*

Description

A character vector containing the Mus musculus proteins, from STRINGdb release 10.5, of combined score greater than or equal to 800.

Usage

```
Mm800
```

Format

An object of class character of length 12166.

Author(s)

Diego Morais

Examples

```
Mm800
```

Mm900	*Ordered Mus musculus proteins of combined score greater than or equal to 900*

Description

A character vector containing the Mus musculus proteins, from STRINGdb release 10.5, of combined score greater than or equal to 900.

Usage

```
Mm900
```

Format

An object of class `character` of length 9648.

Author(s)

Diego Morais

Examples

```
Mm700
```

orderingProperties	*Calculates graph properties projected on the ordered proteins*

Description

Calculates protein (node) properties, such as: degree/connectivity, number of triangles and clustering coefficient; and properties of the window, region of n (radius * 2 + 1) proteins centered at a protein, such as: connectivity, clustering coefficient and modularity.

Usage

```
orderingProperties(object, nCores = 1L)

## S4 method for signature 'Transcriptogram'
orderingProperties(object, nCores = 1L)
```

Arguments

object	An object of class Transcriptogram
nCores	An integer number, referring to the number of processing cores to be used; or a logical value, TRUE indicating that all processing cores should be used, and FALSE indicating the use of just one processing core. The default value of this argument is 1.

Details

Connectivity/degree of a node is the number of edges it presents. A triangle of a node represents a pair of connected neighbors, the number of triangles on the adjacency list of a node is required to calculate its clustering coefficient. The clustering coefficient of a node measures, in the interval [0, 1], the likelihood that any two of its neighbors are themselves connected, this is calculated by the ratio between the number of triangles that the node has, and the maximum possible number of edges on its cluster (nodeTriangles / (nodeDegree * (nodeDegree - 1) / 2)). The window connectivity is the average connectivity calculated over the window. The window clustering coefficient, a value in the interval [0, 1], is the average clustering coefficient calculated over the window. The window modularity, a value in the interval [0, 1], is defined as the ratio between the total number of edges between any two nodes of the window, and the sum of the degrees of the nodes presents in the window. The window considers periodic boundary conditions to deal with proteins near the ends of the ordering.

Value

This method returns a data frame containing: ENSEMBL Peptide ID, its position on the ordering, node degree, number of triangles and clustering coefficient, and window connectivity, clustering coefficient and modularity.

Author(s)

Diego Morais

References

da Silva, S.R.M., Perrone, G.C., Dinis, J.M. and de Almeida, R.M.C., Transcriptograms: Reproducibility enhancement and differential expression of non predefined functional gene sets in human genome. BMC Genomics, 15, 1181 (2014).

Rybarczyk-Filho, J.L., Castro, M.A.A., Dalmolin, R.J, Moreira, J.C.F., Brunnet, L.G. and de Almeida, R.M.C., Towards a genome-wide transcriptogram: the Saccharomyces cerevisiae case. Nucleic Acids Res., 39, 3005-3016 (2011). PMID:21169199

See Also

transcriptogramPreprocess, Hs900, association

Examples

```
transcriptogram <- transcriptogramPreprocess(association, Hs900, 2)
## Not run:
oProperties <- orderingProperties(transcriptogram)

## End(Not run)
```

radius<-   *Radius*

Description

Retrieve or set the content of the radius slot of an object of class Transcriptogram.

Usage

```
radius(object) <- value

radius(object)

## S4 replacement method for signature 'Transcriptogram'
radius(object) <- value

## S4 method for signature 'Transcriptogram'
radius(object)
```

Arguments

object	An object of class Transcriptogram
value	An integer, non negative, number referring to the window radius required for some methods

Value

This method returns the content of the radius slot of an object of class Transcriptogram.

Author(s)

Diego Morais

See Also

Hs900, association, transcriptogramPreprocess, transcriptogramStep2-method, orderingProperties-method

Examples

```
transcriptogram <- transcriptogramPreprocess(association, Hs900, 50)
radius(transcriptogram) <- 80
radius(transcriptogram)
```

Rn700   *Ordered Rattus norvegicus proteins of combined score greater than or*
                              equal to 700

Description

A character vector containing the Rattus norvegicus proteins, from STRINGdb release 10.5, of combined score greater than or equal to 700.

Usage

Rn700

Format

An object of class character of length 14285.

Author(s)

Diego Morais

Examples

Rn700

Rn800   *Ordered Rattus norvegicus proteins of combined score greater than or*
                              equal to 800

Description

A character vector containing the Rattus norvegicus proteins, from STRINGdb release 10.5, of combined score greater than or equal to 800.

Usage

Rn800

Format

An object of class character of length 12437.

Author(s)

Diego Morais

Examples

Rn800

Rn900	*Ordered Rattus norvegicus proteins of combined score greater than or equal to 900*

Description

A character vector containing the Rattus norvegicus proteins, from STRINGdb release 10.5, of combined score greater than or equal to 900.

Usage

Rn900

Format

An object of class character of length 9747.

Author(s)

Diego Morais

Examples

Rn900

Sc700	*Ordered Saccharomyces cerevisiae proteins of combined score greater than or equal to 700*

Description

A character vector containing the Saccharomyces cerevisiae proteins, from STRINGdb release 10.5, of combined score greater than or equal to 700.

Usage

Sc700

Format

An object of class character of length 5586.

Author(s)

Diego Morais

Examples

Sc700

Sc800	*Ordered Saccharomyces cerevisiae proteins of combined score greater than or equal to 800*

Description

A character vector containing the Saccharomyces cerevisiae proteins, from STRINGdb release 10.5, of combined score greater than or equal to 800.

Usage

```
Sc800
```

Format

An object of class `character` of length 5090.

Author(s)

Diego Morais

Examples

```
Sc800
```

Sc900	*Ordered Saccharomyces cerevisiae proteins of combined score greater than or equal to 900*

Description

A character vector containing the Saccharomyces cerevisiae proteins, from STRINGdb release 10.5, of combined score greater than or equal to 900.

Usage

```
Sc900
```

Format

An object of class `character` of length 4386.

Author(s)

Diego Morais

Examples

```
Sc900
```

Transcriptogram-class   *Class Transcriptogram*

Description

This S4 class includes methods to use expression values with ordered proteins.

Slots

association A data frame containing two columns, and rows containing proteins names that are connected

ordering A data frame containing two columns, the first one with proteins names, and the second containing its respective position

transcriptogramS1 A data frame produced as the result of averaging over all identifiers related to the same protein

transcriptogramS2 A data frame produced as the result of averaging over the window

radius An integer, non negative, number referring to the window radius

status An integer used internally to check the status of the object

DE A data frame of differentially expressed proteins

Author(s)

Diego Morais

See Also

transcriptogramPreprocess, DE-method, radius-method, orderingProperties-method, connectivityProperties-method, transcriptogramStep1-method, transcriptogramStep2-method, differentiallyExpressed-method, clusterVisualization-method, clusterEnrichment-method

---

transcriptogramPreprocess
                          Creates an object of class Transcriptogram

---

Description

Constructor for the Transcriptogram object.

Usage

```
transcriptogramPreprocess(association, ordering, radius = 0L)
```

Arguments

association	A matrix, or data frame, containing two columns of ENSEMBL Peptide IDs (character); or the path for a file containing two columns, no header, with rows composed by the ENSEMBL Peptide IDs of two proteins that are connected
ordering	A character vector containing ordered ENSEMBL Peptide IDs; a data frame containing two columns, the first one with ENSEMBL Peptide IDs (character), and the second containing its respective position (integer non negative); or the path for a file containing two columns, a row for the headers, with rows composed respectively, by a ENSEMBL Peptide ID and its respective position
radius	An integer, non negative, number referring to the window radius required for some methods

Value

A preprocessed object of class Transcriptogram.

Author(s)

Diego Morais

See Also

Transcriptogram-class, association, Hs900

Examples

```
transcriptogram <- transcriptogramPreprocess(association, Hs900)
```

---

transcriptogramStep1	*Calculates the average of the expression values related to the same protein*

Description

For each transcriptome sample, this method assigns to each protein the average of the expression values of all the identifiers related to it. It is necessary a `dictionary` to map the identifiers to proteins.

Usage

```
transcriptogramStep1(object, expression, dictionary, nCores = 1L)

## S4 method for signature 'Transcriptogram'
transcriptogramStep1(object, expression, dictionary,
  nCores = 1L)
```

Arguments

`object`	An object of class Transcriptogram
`expression`	A matrix, or data frame, containing normalized expression values from samples of microarrays or RNA-Seq
`dictionary`	A matrix, or data frame, containing two columns, the first column must contains the ENSEMBL Peptide ID, and the second column must contains values that appear as rownames in `expression`, in order to recognize the ENSEMBL Peptide ID of the other column
`nCores`	An integer number, referring to the number of processing cores to be used; or a logical value, TRUE indicating that all processing cores should be used, and FALSE indicating the use of just one processing core. The default value of this argument is 1.

Value

This method creates a data frame to feed the transcriptogramS1 slot of an object of class Transcriptogram. Each row of the data frame contains: an ENSEMBL Peptide ID, its respective position in the ordering and the mean of the expression values of the identifiers related to the same protein.

Author(s)

Diego Morais

References

da Silva, S.R.M., Perrone, G.C., Dinis, J.M. and de Almeida, R.M.C., Transcriptograms: Reproducibility enhancement and differential expression of non predefined functional gene sets in human genome. BMC Genomics, 15, 1181 (2014).

Rybarczyk-Filho, J.L., Castro, M.A.A., Dalmolin, R.J, Moreira, J.C.F., Brunnet, L.G. and de Almeida, R.M.C., Towards a genome-wide transcriptogram: the Saccharomyces cerevisiae case. Nucleic Acids Res., 39, 3005-3016 (2011). PMID:21169199

See Also

transcriptogramPreprocess, GSE9988, GPL570, Hs900, association

Examples

```
transcriptogram <- transcriptogramPreprocess(association, Hs900)
## Not run:
transcriptogram <- transcriptogramStep1(transcriptogram, GSE9988, GPL570)

## End(Not run)
```

---

transcriptogramStep2   *Calculates the average of the expression values using a sliding window*

Description

To each position of the ordering, this method assigns a value equal to the average of the expression values inside a window, region of n (radius * 2 + 1) proteins centered at a protein. The window considers periodic boundary conditions to deal with proteins near the ends of the ordering.

Usage

```
transcriptogramStep2(object, nCores = 1L)

## S4 method for signature 'Transcriptogram'
transcriptogramStep2(object, nCores = 1L)
```

Arguments

object   An object of class Transcriptogram

nCores   An integer number, referring to the number of processing cores to be used; or a logical value, TRUE indicating that all processing cores should be used, and FALSE indicating the use of just one processing core. The default value of this argument is 1.

Value

This method creates a data frame to feed the transcriptogramS2 slot of an object of class Transcriptogram. Each row of the data frame contains: the ENSEMBL Peptide ID used as center of the window, its position on the ordering, and the mean of the expression values of the window.

Author(s)

Diego Morais

References

da Silva, S.R.M., Perrone, G.C., Dinis, J.M. and de Almeida, R.M.C., Transcriptograms: Reproducibility enhancement and differential expression of non predefined functional gene sets in human genome. BMC Genomics, 15, 1181 (2014).

Rybarczyk-Filho, J.L., Castro, M.A.A., Dalmolin, R.J, Moreira, J.C.F., Brunnet, L.G. and de Almeida, R.M.C., Towards a genome-wide transcriptogram: the Saccharomyces cerevisiae case. Nucleic Acids Res., 39, 3005-3016 (2011). PMID:21169199

See Also

transcriptogramPreprocess, GSE9988, GPL570, Hs900, association, transcriptogramStep1-method

Examples

```
transcriptogram <- transcriptogramPreprocess(association, Hs900, 50)
## Not run:
transcriptogram <- transcriptogramStep1(transcriptogram, GSE9988, GPL570)
transcriptogram <- transcriptogramStep2(transcriptogram)

## End(Not run)
```

Index