Package 'pcaGoPromoter'

September 24, 2012

Version 1.0.0

Date 2012-03-16

Title pcaGoPromoter is used to analyze DNA micro array data

Author Morten Hansen, Jorgen Olsen

Maintainer Morten Hansen <mhansen@sund.ku.dk>

Description This package contains functions to ease the analyses of DNA micro arrays. It utilizes principal component analysis as the initial multivariate analysis, followed by functional interpretation of the principal component dimensions with overrepresentation analysis for GO terms and regulatory interpretations using overrepresentation analysis of predicted transcription factor binding sites with the primo algorithm.

biocViews GeneExpression, Microarray, GO , Visualization

Imports Biobase (>= 2.10.0) , AnnotationDbi

Depends R (>= 2.14.0) , ellipse

Suggests
Rgraphviz, GO.db, hgu133plus2.db, mouse4302.db, rat2302.db,hugene10sttranscriptcluster.db, mogene10sttranscriptcluster.db,Biostrings, pcaGoPromoter.Hs.hg19, pcaGoPromoter.Mm.mm9,pcaGoPromoter.Rn.rn4, serumStimulation, parallel

LazyLoad yes

License GPL (>= 2)

R topics documented:

pcaGoPromoter-package *Tools for analyzing data from DNA micro arrays.*

Description

Welcome to the pcaGoPromoter package. The purpose of this package is to ease the analyses of data from genome wide gene expression analysis. Multiple platforms can be used (Affymetrix GeneChip, Illumina expression beadchips, qRT-PCR and more). Probe identifiers should be either Affymetrix Probe Set ID, Gene Symbol or Entrez gene ID.

The overall analysis strategy is to conduct principal component analysis followed by the interpretation of the principal component dimensions by functional annotation for biological function and prediction of regulatory transcription factor networks.

You can learn what objects this package supports with the following command:

'ls("package:pcaGoPromoter")'

Each of these objects has their own manual page detailing where relevant data was obtained along with some examples of how to use it.

Author(s)

Morten Hansen <mhansen@sund.ku.dk> and Jorgen Olsen <jolsen@sund.ku.dk>

Maintainer: Morten Hansen <mhansen@sund.ku.dk>

Examples

```
ls("package:pcaGoPromoter")
```

GOtree *GOtree, plot.GOtree and GOtreeWithLeaveOneOut*

Description

GOtree finds significantly overrepresented Gene Ontology terms in a list of probes (only Biological processes) and return an object of type 'GOtree'.

print.GOtree list significant GO terms from an object of type 'GOtree'.

plot.GOtree creates a visual representation of the GO connection from an object of type 'GOtree'.

GOtreeHits return the genes/probes for a specific GO term.

GOtreeWithLeaveOut returns the same as GOtree, but run through the samples multiple times with 'Leave one out' cross-validation.

Usage

```
GOtree(input, inputType = "hgu133plus2", org = "Hs",
       statisticalTest = "binom", binomAlpha = NA,
       p.adjust.method = "fdr")

## S3 method for class 'GOtree'
print(x, ...)

## S3 method for class 'GOtree'
plot(x, boxes = 25, legendPosition = "topright",
     main = "Gene Ontology tree, biological processes", ...)

GOtreeHits(input, inputType = "hgu133plus2", org = "Hs",
           GOid, returnGeneSymbols = TRUE)

GOtreeWithLeaveOut(exprsData, inputType = "hgu133plus2", org = "Hs",
                   pc = 1, decreasing = TRUE, noProbes = 1000,
                   leaveOut = 1, runs = NCOL(exprsData))
```

Arguments

input	a character vector of Affymetrix probe ids, gene symbols or Entres gene IDs.
inputType	a character vector description the input type. Must be Affymetrix chip type, "geneSymbol" or "entrezID". The following Affymetrix chip type are supported: hgu133plus2, mouse4302, rat2302, hugene10st and mogene10st. Default is Affymetrix chip type "hgu133plus2".
org	a character vector with the organism. Can be "Hs", "Mm" or "Rn". Only needed if inputType is "geneSymbol" or "entrezID". See details. Default is "Hs".
statisticalTest	a character vector with the statistical method to be used. Can be "binom" or "fisher". Default is "binom".
binomAlpha	a value with the pvalue for use in self contained test.
p.adjust.method	the method for adjust p-values due to multiple testing. This will come in a separate column.
x	an object of type 'GOtree'.
boxes	an integer indication the amount of boxes (terms) in the plot.
legendPosition	a vector description the position of the legend. See ?xy.coords for possibilities. Set to NULL for no legend. Default is "topright".
main	a title for the GO tree plot
...	other parameters to be passed through to plotting functions.
GOid	a vector with the GO term of interest.
returnGeneSymbols	a logical indication whether gene symbols or probe ids should be returned. Default is gene symbols.
exprsData	A table with expression data. Row names should be probe identifiers (Affymetrix Probe set ID, Gene Symbols or Entrez gene ID). Column names should be sample identifiers.

pc	a number indication which principal component to extract the probe list based on the loading values from the pca.
decreasing	a logical value indication whether the loadings should be sorted in decreasing of ascending order (`decreasing == FALSE`). Decreasing order yields information about the positive direction and ascending order about the negative direction of the particular principal component.
noProbes	a number indicating the number of probes included in the calculations
leaveOut	a number indication what percentage to leave out in the cross-validation. If set to 1, each observation would be left out once and runs is set equal to number of observations. Deafault is 1.
runs	a number indicating how many times to run with leave out. If leaveOut = 1, runs is overrided with number of observations.

Details

GOtree returns a GOtree object. In contains a list of significant GO terms. plot() generated a visual plot of the GO tree.

GOtreeHits returns a vector with the genes/probes in a specific GO term.

GOtreeWithLeaveOut repeats function GOtree, but with different input. GOtreeWithLeaveOut takes a table of expression data as input, performs PCA, extracts probes / genes for the specified principal component and subsequenly performs GOtree. This is repeated the specified number of times. It can run with leave one out or with leave out a percentage. Only the GO terms that is found overrepresented in all the runs "qualifies" and a new p-value is calculated as the median of the p-value from all the runs. An object of type GOtree is returned.

Value

GOtree returns a object of type GOtree.

GOtreeWithLeaveOut returns a object of type GOtree.

Note

GOtreeWithLeaveOut may take some time to run - depending on the number of samples.

Author(s)

Morten Hansen <mhansen@sund.ku.dk> and Jorgen Olsen <jolsen@sund.ku.dk>

Examples

```
library(serumStimulation)
data(serumStimulation)
pcaOutput <- pca( serumStimulation )
posLoadings <- getRankedProbeIds( x=pcaOutput )
GOs <- GOtree( input=posLoadings[1:1000] )
GOs
plot(GOs, legendPosition=NULL)

## Not run:
GOs <- GOtreeWithLeaveOut( exprsData=serumStimulation )
## End(Not run)
```

pca *pca - Principal Component Analysis (PCA)*

Description

pca - This is a wrapper function around prcomp from stats package. It will prepare the DNA micro array data, run the principal component analysis and return an object of type 'pca'

plot.pca - Makes a score plot with legend of object type 'pca'.

getProbeIds.pca - Get probe ids from PCA loadings.

pcaInfoPlot - Makes a informative score plot with optional overrepresented GO terms and predicted transcription factor binding sites annotated along the axis.

Usage

```
pca(eData, printDropped = TRUE, scale=TRUE, center=TRUE)

## S3 method for class 'pca'
plot(x, groups, PCs = c(1,2), printNames = TRUE,
     symbolColors = TRUE, plotCI = TRUE, GOtreeObjs, primoObjs,
     main, ...)

## S3 method for class 'pca'
getRankedProbeIds(x, pc = 1, decreasing = TRUE)

pcaInfoPlot(eData, inputType = "hgu133plus2", org = "Hs", groups,
            noProbes = 1365, GOtermsAnnotation = TRUE,
            primoAnnotation = TRUE)
```

Arguments

eData	an ExpressionSet object or a table with expression data with variables in rows and observations in columns. Row names should be probe identifiers (Affymetrix probe set ID, Gene Symbol or Entrez IDs), column names should be sample identifiers.
printDropped	a logical value determination whether or not dropped probes should be printed. Probes are dropped if all values are equal.
scale	a logical value determination whether or not input values should be scaled. Default is TRUE.
center	a logical value determination whether or not input values should be centered. Default is TRUE.
x	Object of type 'pca'. Created by function pca.
groups	a factor containing group information.
PCs	an integer vector indication which principal components to use in the score plot. Exactly two principal components should be given.
printNames	a logical value indicating whether observation names should be printed. Default is TRUE.
symbolColors	a logical value indicating whether the symbols should be plotted with colors. Default is TRUE.

plotCI	a logical value indicating whether the confidence intervals for each group should be plotted. Default is TRUE.
GOtreeObjs	a list of GOtree objects which is used for annotation of the axis in the PCA plot - used by function pcaInfoPlot().
primoObjs	a list of primo objects which is used for annotation of the axis in the PCA plot - used by function pcaInfoPlot().
main	a character vector with the title of the plot. If no title is given, a default one will be provided. Set to NULL for no title.
...	other parameters to be passed through to plotting functions.
pc	a number indication which principal component to extract the probe identifiers from.
decreasing	a logical value indication whether the probe identifiers from PCA loadings should be sorted in decreasing of ascending order (`decreasing == FALSE`).
inputType	a character vector with the input type. See '?primo' for details. Default is "hgu133plus2".
org	a character vector with the organism. See '?primo' for details. Default is "Hs".
noProbes	a numeric value indication the number of probes to use for GOtree and primo.
GOtermsAnnotation	a logical value to select GO terms annotation along the axis. The default is TRUE.
primoAnnotation	a logical value to select predicted transcription factor binding site annotation along the axis. The default is TRUE.

Details

pca uses `prcomp` to do the principal component analysis. The input data (x) is scaled and centered, so constant variables (sd = 0) will be removed to avoid divison by zero.

plot.pca makes a PCA score plot. The input data is an object of type 'pca'. The score plot is formatted and the observations are colored according to the class. A legend describing the class is placed below the plot. The proportion of variance is plotted along the axis.

getRankedProbeIds.pca extract the probe identifiers from PCA loadings from an object of type `pca`. The purpose of this function is to ease the extraction of probe identifiers from an object of type `pca`. It generally just gets the rownames from the loadings and sort it according to `pc` and `decreasing`.

Value

`pca` returns the an object of type 'pca'.

`getRankedProbeIds` return a vector with the probe ids sorted from higher to lower if `decreasing = TRUE` or from lower to higher if `decreasing = FALSE`.

Author(s)

Morten Hansen <mhansen@sund.ku.dk> and Jorgen Olsen <jolsen@sund.ku.dk>

See Also

`prcomp`.

Examples

```
library(serumStimulation)
data(serumStimulation)
groups <- as.factor(c(rep("control", 5), rep("serumInhib", 5),
                    rep("serumOnly", 3)))
pcaOutput <- pca(serumStimulation)
pcaOutput
plot(pcaOutput, groups=groups)
posLoadings <- getRankedProbeIds(pcaOutput)
```

---

primo *primo, primoHits and primoWithLeaveOut*

Description

primo returns the predicted transcription factor binding sites for a given subset of probes.

primoHits returns a list of refseqs where a given pwm scores above a threshold.

primoWithLeaveOut returns the same as `primo`, but run through the samples multiple times with applied 'Leave out' cross-validation settings.

Usage

```
primo( input, inputType = "hgu133plus2", org = "Hs",
       PvalueCutOff = 0.05, cutOff = 0.9,
       p.adjust.method = "fdr", printIgnored = FALSE , primoData = NULL )

primoHits( input, inputType = "hgu133plus2", org = "Hs",
           id, cutOff = 0.9, printIgnored = FALSE , primoData = NULL )

primoWithLeaveOut( exprsData, inputType = "hgu133plus2", org = "Hs",
                   pc = 1, decreasing = TRUE, noProbes = 1000,
                   leaveOut=1, runs = NCOL(exprsData) )
```

Arguments

`input`	a character vector with the input. Can be Affymetrix probeset IDs, Entrez IDs or gene symbols.
`inputType`	a character vector description the input type. Must be Affymetrix chip type, "geneSymbol" or "entrezID". The following Affymetrix chip type are supported: hgu133plus2, mouse4302, rat2302, hugene10st and mogene10st. Default is Affymetrix chip type "hgu133plus2".
`org`	a character vector with the organism. Can be "Hs", "Mm" or "Rn". Only needed if `inputType` is "geneSymbol" or "entrezID". See details. Default is "Hs".
`PvalueCutOff`	a P-value cut off. All P-value that are below this cut off will be returned.
`cutOff`	the percentage of promoters that each TF should bind to in theory
`p.adjust.method`	the method for adjust p-values due to multiple testing. Default is "fdr" (False Discovery Rate). See manual pages p.adjust for possibilities.
`printIgnored`	Should the ignored RefSeqs be printed? Default is FALSE.

primoData	primoData object from function primoData. See manual pages for primoData for more information. Default is NULL.
id	a vector with the pwm id used to retrieve the probes with the corresponding predicted hits. The ids should be taken from the "id" column of the primo output.
exprsData	a table with expression data. Variables in rows and observations in columns. Row names should be probe identifiers (Affymetrix probe set ID, Gene Symbol or Entrez IDs), column names should be sample identifiers.
pc	a number indication which principal component to extract the loadings from.
decreasing	a logical value indication whether the loadings should be sorted in decreasing of ascending order (`decreasing == FALSE`).
noProbes	a number indicating how many probe loadings to use
leaveOut	a number indication what percentage to leave out. If set to 1, each observation would be left out once and runs is set equal to number of observations. Deafault is 1.
runs	a number indicating how many times to run with leave out. If leaveOut = 1, runs is overrided with number of observations.

Details

primo returns the predicted transcription factor binding sites for a given set of probes (Affymetrix probe ids, Entrez ID or gene symbols). Note: All input values are first converted into refseqs.

primoHits returns all refseqs representing promoters where the pwm scores above the threshold. The threshold is based on the cutOff percentage. For each promoter the maximum value of the pwm is collected and sorted in increasing order. The threshold is the pwm score at the cutOff percentage in the list. Each promoter that has a scores above the threshold is considered to have potentionel binding site of the transcription factor.

primoWithLeaveOut repeats function primo, but with different input. primoWithLeaveOut runs primo the specified number of times. It can with leave one out or with leave out a percentage. Only the pwns that is represented in all the runs "qualifies" and a new p-value is calculated as the median of the p-value from all the runs. The list of over and under represneted pwms is returned.

Value

upRegulated	a data frame with the possible transcription factors for up regulated genes.
downRegulated	a data frame with the possible transcription factors for down regulated genes.

primoHits returns a character vector of probe ids (or refseqs).

primoWithLeaveOneOut returns the same data structure as primo.

Note

The P-values in column 'pValue' of the output is not corrected for multiple testing.

primoWithLeaveOneOut takes some time to run - especially if the is a lot of samples.

Author(s)

Morten Hansen <mhansen@sund.ku.dk> and Jorgen Olsen <jolsen@sund.ku.dk>

References

Stegmann, A., Hansen, M., et al. (2006). "Metabolome, transcriptome, and bioinformatic cis-element analyses point to HNF-4 as a central regulator of gene expression during enterocyte differentiation." Physiological Genomics 27(2): 141-155.

Examples

```
library(serumStimulation)
data(serumStimulation)
pcaOutput <- pca( serumStimulation )
negLoadings <- getRankedProbeIds( pcaOutput, pc=1, decreasing=FALSE )
TFs <- primo( negLoadings[1:1000] )
probeIds <- primoHits( negLoadings , id="M01031" )

## Not run:
TFs <- primoWithLeaveOut( serumStimulation )

## End(Not run)
```

`primoData` *primoData*

Description

primoData calculated the data objects used by primo().

Usage

```
primoData(promoters, matrices, cleanUpPromoters = TRUE)
```

Arguments

`promoters`	a list with promoter data. See details.
`matrices`	a list with matrices. See details.
`cleanUpPromoters`	Should some house-keeping procedures (remove dublets, map common name etc.) be preformed on promoters?

Details

primoData makes a custom data object for function primo().

primoData needs 2 arguments: promoters and matrices.

promoter must consist of 2 list: a list with the RefSeq names for all the promoters and a list with all the promoter sequences in lowercase letters. The length of the two lists must be the same.

The function pcaGoPromoters:::primoData.getPromoter(filename) can be used to load promoter from a file in FASTA format.

matrices must consist of one list of lists. Each list element must have a unique name and holds 3 data elements: baseId, name and pwm. baseId is a character vector with a base id. name is a character vector with the common name for the matrix. pwm is a position weighted matrix with the base A,C,G,T in rows and the weights in columns.

Example of promoter and matrices data types is shown in the wiki at: ???

primoData returns an object of type primoData. It contains to list - threshold and promotersRefseqs.

thresholds is a list of lists. Each list element have a unique name and holds 4 data elements: baseId, name, pwmLength and maxScores. baseId is a charactor vector with a base id. name is a character vector with the common name for the matrix. pwmLength is the length of the position weighted matrix. maxScores holds the max score from all the promoters.

promotersRefseq is a list with the Refseq names for all the promoters. Some promoters can have one that one refseq name. These names maps to the maxScores from thresholds.

Value

primoData returns an object to be used by primo() as argument primoData.

Note

primoData takes very long time (many hours) to run. Install package 'multicore' and use a computer with multiple cores.

Author(s)

Morten Hansen <mhansen@sund.ku.dk> and Jorgen Olsen <jolsen@sund.ku.dk>

References

Stegmann, A., Hansen, M., et al. (2006). "Metabolome, transcriptome, and bioinformatic cis-element analyses point to HNF-4 as a central regulator of gene expression during enterocyte differentiation." Physiological Genomics 27(2): 141-155.

Examples

```
## Not run:
myPrimoData <- primoData( myPromoters, myMatrices)

## End(Not run)
```

Index