Package ‘oposSOM’

March 23, 2017

Title Comprehensive analysis of transcriptome data

Version 1.12.0

Description This package translates microarray expression data into metadata
of reduced dimension. It provides various sample-centered and group-centered
visualizations, sample similarity analyses and functional enrichment analyses.
The underlying SOM algorithm combines feature clustering, multidimensional
scaling and dimension reduction, along with strong visualization capabilities.
It enables extraction and description of functional expression modules inherent
in the data.

Depends R (>= 3.0), igraph (>= 1.0.0)

Imports som, fastICA, scatterplot3d, pixmap, fdrtool, ape, KernSmooth,
biomaRt, Biobase

License GPL (>=2)

Author Henry Loeffler-Wirth <wirth@izbi.uni-leipzig.de> and Martin Kalcher <mkalcher@porkbox.net>

Maintainer Henry Loeffler-Wirth <wirth@izbi.uni-leipzig.de>

URL http://som.izbi.uni-leipzig.de

biocViews GeneExpression, DifferentialExpression, GeneSetEnrichment,
DataRepresentation, Visualization

NeedsCompilation no

R topics documented:

oposSOM-package ............................................................ 2
opossm.genesets ............................................................ 3
opossm.new ................................................................. 4
opossm.run ................................................................. 6
opossm.tissues ............................................................. 6

Index 8
This package translates microarray expression data into metadata of reduced dimension. It provides various sample-centered and group-centered visualizations, sample similarity analyses and functional enrichment analyses. The underlying SOM algorithm combines feature clustering, multidimensional scaling and dimension reduction, along with strong visualization capabilities. It enables extraction and description of functional expression modules inherent in the data. The results are given within a separate folder and can be browsed using the summary HTML file.

Details

Package: oposSOM
Type: Package
Version: 1.9.2
Date: 2016-24-03
License: GPL (>= 2)

Author(s)

Author: Henry Loeffler-Wirth <wirth@izbi.uni-leipzig.de> and Martin Kalcher <mkalcher@porkbox.net>
Maintainer: Henry Loeffler-Wirth <wirth@izbi.uni-leipzig.de>

References

Loeffler-Wirth, Kalcher, Binder: oposSOM: R-package for high-dimensional portraying of genome-wide expression landscapes on Bioconductor. (Bioinformatics 2015)

Examples

# Example with artificial data
env <- opossom.new(list(dataset.name="Example", 
                      dim.1stLvlSom=20))

env$indata <- matrix(runif(10000), 1000, 10)

env$group.labels <- "auto"

opossom.run(env)

# Real Example - This will take several minutes
opossom.genesets

#env <- opossom.new(list(dataset.name="Tissues",
#  dim.1stLvlSom=30,
#  geneset.analysis=TRUE,
#  pairwise.comparison.list=list(
#    list("Homeostasis"=c(1, 2), "Imune System"=c(9, 10)),
#    list("Homeostasis"=c(1, 2), "Muscle"=c(8))
#  )))
#data(opossom.tissues)
#env$indata <- opossom.tissues
# #env$group.labels <- c(rep("Homeostasis", 2),
# #  "Endocrine",
# #  "Digestion",
# #  "Exocrine",
# #  "Epithelium",
# #  "Reproduction",
# #  "Muscle",
# #  rep("Imune System", 2),
# #  rep("Nervous System", 2))
# #opossom.run(env)

opossom.genesets | Additional literature genesets

Description
Genesets collected from publications and independent analyses.

Usage
data(opossom.genesets)

Format
The data set is stored in RData (binary) format. Each element of the list represents one distinct gene set and contains the Ensembl-IDs of the member genes.

Details
The oposSOM package allows for analysing the biological background of the samples using predefined sets of genes of known biological context. A large and diverse collection of such gene sets is automatically derived from the Gene Ontology (GO) annotation database using biomaRt interface. opossom.genesets contains more than 4,500 additional gene sets collected from Biocarta, KEGG and Reactome databases, from literature on chemical and genetic perturbations, from literature on cancer types and subtypes, and from previous analyses using the oposSOM pipeline.
Initialize the oposSOM pipeline.

Description
This function initializes the oposSOM environment and sets the preferences.

Usage
oppossom.new(preferences)

Arguments
preferences list with the following optional values:

- indata: input data matrix containing the expression values or an Biobase::ExpressionSet object (see 'Details' and 'Examples')
- group.labels: sample assignment to a distinct group, subtype or class (character; "auto" or one label for each sample; may be given with indata ExpressionSet)
- group.colors: colors of the samples for diverse visualizations (character; one color for each sample; may be given with indata ExpressionSet)
- dim.1stLvlSom: dimension of primary SOM; use "auto" to apply automatic size estimation (integer, >5)
- dim.2ndLvlSom: dimensions of second level SOM (integer, >5)
- training.extension: factor to extend the number of iterations in SOM training (numerical, >0)
- rotate.SOM.portraits: number of roations of the primary SOM in counter-clockwise fashion (integer \{0,1,2,3\})
- flip.SOM.portraits: mirroring the primary SOM along the bottom-left to top-right diagonal (boolean)
- database.dataset: type of ensemble dataset addressed with biomaRt interface; use "auto" to detect parameter automatically (character)
- database.id.type: type of rowname identifier in biomaRt database; obsolete if database.dataset="auto" (character)
- geneset.analysis: perform geneset analysis (boolean)
- geneset.analysis.exact: enables p-value and fdr calculation in geneset analysis; obsolete if geneset.analysis=F (boolean)
- standard.spot.modules: spot modules utilized in diverse downstream analyses (character, one of \{"overexpression", "group.overexpression", "underexpression", "kmeans", "correlation", "dmap"\})
- spot.coresize.modules: spot detection in summary maps, minimum size (numerical, >0)
- spot.threshold.modules: spot detection in summary maps, expression threshold (numerical, between 0 and 1)
- spot.coresize.groupmap: spot detection in group-specific summary maps, minimum size (numerical, >0)
- spot.threshold.groupmap: spot detection in group-specific summary maps, expression threshold (numerical, between 0 and 1)
- feature.centralization: enables centralization of the features (boolean)
• `sample.quantile.normalization`: enables quantile normalization of the samples (boolean)
• `pairwise.comparison.list`: group list for pairwise analyses (list of group lists, see 'Examples') or NULL otherwise

**Details**

The package then accepts the `indata` parameter in two formats:<br>Firstly a simple two-dimensional numerical matrix, where the columns and rows represent the samples and genes, respectively. The expression values are usually obtained by calibration and summarization algorithms (e.g. MAS5, VSN or RMA), and transformed into logarithmic scale prior to utilizing them in the pipeline. Secondly the input data can also be given as `Biobase::ExpressionSet` object. Please check the vignette for more details on the parameters.

**Value**

A new oposSOM environment which is passed to `opossom.run`.

**Examples**

```r
env <- opossom.new(list(dataset.name="Example",
  dim.1stLvlSom="auto",
  dim.2ndLvlSom=10,
  training.extension=1,
  rotate.SOM.portraits=0,
  flip.SOM.portraits=FALSE,
  database.dataset="auto",
  geneset.analysis=TRUE,
  geneset.analysis.exact=TRUE,
  standard.spot.modules="dmap",
  spot.coresize.modules=4,
  spot.threshold.modules=0.9,
  spot.coresize.groupmap=4,
  spot.threshold.groupmap=0.7,
  feature.centralization=TRUE,
  sample.quantile.normalization=TRUE,
  pairwise.comparison.list=list(
    list("groupA"=c("sample1", "sample2"),
      "groupB"=c("sample3", "sample4")))))
```

# definition of indata, group.labels and group.colors
env$indata = matrix( runif(1000), 100, 10 )
env$group.labels = c( rep("class 1", 5), rep("class 2", 4), "class 3" )
env$group.colors = c( rep("red", 5), rep("blue", 4), "green" )

# alternative definition of indata, group.labels and group.colors using `Biobase::ExpressionSet`
library(Biobase)
env$indata = ExpressionSet( assayData=matrix(runif(1000), 100, 10),
  phenoData=AnnotatedDataFrame(data.frame(
    group.labels = c( rep("class 1", 5), rep("class 2", 4), "class 3" ),
    group.colors = c( rep("red", 5), rep("blue", 4), "green" ) ),
  )
)
opossom.run

*Execute the oposSOM pipeline.*

**Description**

This function realizes the complete pipeline functionality: single gene expression values are clustered to metagenes using a self-organizing map. Based on these metagenes, visualizations (e.g. expression profiles), downstreaming sample similarity analyses (e.g. hierarchical clustering, ICA) and functional enrichment analyses are performed. The results are given within a separate folder and can be browsed using the summary HTML file.

**Usage**

```r
opossom.run(env)
```

**Arguments**

- `env` the opossom environment created with `opossom.new` according to the users’ preferences

**Examples**

```r
# Example with artificial data
env <- opossom.new(list(dataset.name="Example", 
                      dim.1stLvlSom=20))

env$indata <- matrix(runif(1000), 100, 10)

opossom.run(env)
```

---

**opossom.tissues**

*Example data set.*

**Description**

A data set comprising of 12 selected human tissues.

**Usage**

```r
data(opossom.tissues)
```

**Format**

The data set is stored in RData (binary) format.

**Details**

The data set was downloaded from Gene Expression Omnibus repository (http://www.ncbi.nlm.nih.gov/geo, GEO accession no. GSE7307). About 20,000 genes in more than 650 samples were measured using the Affymetrix HGU133-Plus2 microarray. A subset of 12 selected tissues from different categories is used as example data set for the oposSOM-package.
Source

Index

«Topic package
  oposSOM-package, 2

oposSOM (oposSOM-package), 2
oposSOM-package, 2
opossom.genesets, 3
opossom.new, 4
opossom.run, 6
opossom.tissues, 6