Package 'SmCCNet'

October 12, 2022

Type Package

Title Sparse Multiple Canonical Correlation Network Analysis Tool

Version 0.99.0

Date 2019-2-24

Author W. Jenny Shi [aut, cre],
Laura Saba [aut],
Katerina Kechris [aut]

Maintainer W. Jenny Shi <wjennyshi@gmail.com>

Description A canonical correlation based framework for constructing phenotype-specific multi-omics networks by integrating multiple omics data types and a quantitative phenotype of interest.

URL https://github.com/KechrisLab/SmCCNet

Depends R (>= 3.5)

Imports PMA, Matrix, pbapply, igraph

License GPL-3

Encoding UTF-8

LazyData true

biocViews Network

RoxygenNote 6.0.1

NeedsCompilation no

Repository CRAN

Date/Publication 2019-03-04 16:50:03 UTC

R topics documented:

`geneExpr` *A synthetic mRNA expression dataset.*

Description

A matrix containing simulated mRNA expression levels for 358 subjects (rows) and 500 features (columns).

Usage

```
geneExpr
```

Format

An object of class `matrix` with 358 rows and 500 columns.

`getAbar` *Compute the similarity matrix based on one or more canonical correlation weight vectors.*

Description

Compute the similarity matrix based on the outer products of absolute canonical correlation weights.

Usage

```
getAbar(Ws, P1 = NULL, FeatureLabel = NULL)
```

Arguments

`Ws`	A canonical correlation weight vector or matrix. If `Ws` is a matrix, then each column corresponds to one weight vector.
`P1`	Total number of features for the first omics data type.
`FeatureLabel`	If `FeatureLabel` = `NULL` (default), the feature names will be $\{TypeI_1, \cdots, TypeI_{p_1}, TypeII_1, \cdots, Type$ where p_1 =`P1`, and p is the total number of omics features.

Value

A $p \times p$ symmetric non-negative matrix.

Examples

```
w <- matrix(rnorm(6), nrow = 3)
Ws <- apply(w, 2, function(x)return(x/sqrt(sum(x^2))))
abar <- getAbar(Ws, P1 = 2, FeatureLabel = NULL)
```

getMultiOmicsModules *Extract multi-omics modules based on the similarity matrix.*

Description

Apply hierarchical tree cutting to the similarity matrix and extract modules that contain both omics data types.

Usage

```
getMultiOmicsModules(Abar, P1, CutHeight = 1 - 0.1^10, PlotTree = TRUE)
```

Arguments

Abar	A similary matrix for all features (both omics data types).
P1	Total number of features for the first omics data type.
CutHeight	Height threshold for the hierarchical tree cutting. Default is $1 - 0.1^{10}$.
PlotTree	Logical. Whether to create a hierarchical tree plot.

Value

A list of multi-omics modules.

Examples

```
set.seed(123)
w <- rnorm(5)
w <- w/sqrt(sum(w^2))
abar <- getAbar(w, P1 = 2, FeatureLabel = NULL)
modules <- getMultiOmicsModules(abar, P1 = 2, CutHeight = 0.5)
```

getRobustPseudoWeights

Calculate the canonical correlation weights based on sparse multiple canonical correlation analysis (SmCCA), sparse supervised canonical correlation analysis (SsCCA), or sparse canonical correlation analysis (SCCA).

Description

Integrate two omics data type (and a quantitative phenotype), and calculate the absolute canonical correlation weights for the omics features using SmCCA SsCCA, or SCCA. SmCCA and SsCCA take into account a phenotype/trait. SmCCA maximizes the total (weighted or unweighted) pairwise canonical correlation weights between two omics data types and the trait. It requires the trait to be quantitative. SsCCA prioritizes omics features based on the trait, and assigns non-zero canonical weights to features that are more correlated to the trait. SCCA does not use any trait information for computing the canonical correlation weights. All of these three methods are included in this function, along with an omics feature subsampling scheme.

Usage

```
getRobustPseudoWeights(X1, X2, Trait, Lambda1, Lambda2, s1 = 0.7, s2 = 0.7,
  NoTrait = FALSE, FilterByTrait = FALSE, SubsamplingNum = 1000,
  CCcoef = NULL, trace = FALSE)
```

Arguments

`X1`	An $n \times p_1$ data matrix (e.g. mRNA) with p_1 features and n subjects.
`X2`	An $n \times p_2$ data matrix (e.g. miRNA) with p_2 features and n subjects.
`Trait`	An $n \times 1$ trait data matrix for the same n subjects.
`Lambda1`	LASSO penalty parameter for `X1`. `Lambda1` needs to be between 0 and 1.
`Lambda2`	LASSO penalty parameter for `X2`. `Lambda2` needs to be between 0 and 1.
`s1`	Proportion of mRNA features to be included, default at `s1 = 0.7`. `s1` needs to be between 0 and 1.
`s2`	Proportion of miRNA features to be included, default at `s1 = 0.7`. `s2` needs to be between 0 and 1.
`NoTrait`	Logical, default is `FALSE`. Whether trait information is provided.
`FilterByTrait`	Logical, default is `FALSE`. Whether only the top (80%) features with highest correlation to the trait will be assigned nonzero weights. The choice of 80% is based on the PMA package.
`SubsamplingNum`	Number of feature subsamples. Default is 1000. Larger number leads to more accurate results, but at a higher cost.
`CCcoef`	Optional coefficients for the SmCCA pairwise canonical correlations. If `CCcoef = NULL` (default), then the objective function is the total sum of all pairwise canonical correlations. It can also be a coefficient vector that follows the column order of `combn(K, 2)`.
`trace`	Logical. Whether to display the CCA algorithm trace.

Details

To choose SmCCA, set `NoTrait = FALSE, FilterByTrait = FALSE`. To choose SsCCA, set `NoTrait = FALSE, FilterByTrait = TRUE`. To choose SCCA, set `Trait = NULL, NoTrait = TRUE`.

Value

A canonical correlation weight matrix with $p_1 + p_2$ rows. Each column is the canonical correlation weights based on subsampled `X1` and `X2` features. The number of columns is `SubsamplingNum`.

Examples

```
## For illustration, we only subsample 5 times.
set.seed(123)

# Unweighted SmCCA
W1 <- getRobustPseudoWeights(geneExpr, mirnaExpr, Trait = pheno, Lambda1 = 0.05,
  Lambda2 = 0.05, s1 = 0.7, s2 = 0.9, NoTrait = FALSE, FilterByTrait = FALSE,
  SubsamplingNum = 5, CCcoef = NULL, trace = FALSE)

# Weighted SmCCA
W2 <- getRobustPseudoWeights(geneExpr, mirnaExpr, Trait = pheno, Lambda1 = 0.05,
  Lambda2 = 0.05, s1 = 0.7, s2 = 0.9, NoTrait = FALSE, FilterByTrait = FALSE,
  SubsamplingNum = 5, CCcoef = c(1, 5, 5), trace = FALSE)

# SsCCA
W3 <- getRobustPseudoWeights(geneExpr, mirnaExpr, Trait = pheno, Lambda1 = .05, Lambda2 = 0.5,
  s1 = 0.7, s2 = 0.9, NoTrait = FALSE, FilterByTrait = TRUE,
  SubsamplingNum = 5, CCcoef = NULL, trace = FALSE)

# SCCA
W4 <- getRobustPseudoWeights(geneExpr, mirnaExpr, Trait = NULL, Lambda1 = 0.05,
  Lambda2 = 0.05, s1 = 0.7, s2 = 0.9, NoTrait = TRUE,
  SubsamplingNum = 5, CCcoef = NULL, trace = FALSE)
```

---

mirnaExpr *A synthetic miRNA expression dataset.*

Description

A matrix containing simulated miRNA expression levels for 358 subjects (rows) and 100 features (columns).

Usage

```
mirnaExpr
```

Format

An object of class `matrix` with 358 rows and 100 columns.

pheno *A synthetic phenotype dataset.*

Description

A matrix containing simulated quantitative phenotype measures for 358 subjects (rows).

Usage

```
pheno
```

Format

An object of class `matrix` with 358 rows and 1 columns.

plotMultiOmicsNetwork *Plot multi-omics module networks.*

Description

Plot multi-omics modules based on similarity matrix derived from pseudo canonical weights and pairwise feature correlations.

Usage

```
plotMultiOmicsNetwork(Abar, CorrMatrix, multiOmicsModule, ModuleIdx, P1,
  EdgeCut = 0, FeatureLabel = NULL, AddCorrSign = TRUE, SaveFile = NULL,
  ShowType1Label = TRUE, ShowType2Label = TRUE, PlotTitle = "",
  NetLayout = "lgl", ShowNodes = TRUE, VertexLabelCex = 1,
  VertexSize = 1)
```

Arguments

`Abar`	A $p \times p$ similary matrix for both omics data types based on pseudo canonical correlation weights. p is the number of total features for the two omics data types. All entries are non-negative.
`CorrMatrix`	A $p \times p$ correlation matrix that provides sign information for the network.
`multiOmicsModule`	A list of multi-omics modules.
`ModuleIdx`	Index for the module to be plotted. It can not exceed the length of `multiOmicsModule`.
`P1`	Total number of features for the first omics data type.
`EdgeCut`	A numerical value between 0 and 1, indicating an edge threshold for the network. Any features (network nodes) without any edge strength that passes the threshold are excluded from the figure. If `EdgeCut = 0` (default), then the full module network will be created.

FeatureLabel	A $1 \times p$ vector indicating feature names. If FeatureLabel = NULL (default), the feature names will be $\{TypeI_1, \cdots, TypeI_{p_1}, TypeII_1, \cdots, TypeII_{p-p_1}\}$, where $p_1 =$P1.
AddCorrSign	Logical. Whether to add a positive or negative sign to each network edge based on pairwise feature correlations.
SaveFile	A pdf file name for the figure output. If SaveFile = NULL (default), the figure will not be saved.
ShowType1Label	Logical. Whether to label the network nodes for the first omics data type.
ShowType2Label	Logical. Whether to label the network nodes for the second omics data type.
PlotTitle	A title for the figure. Default is without any title.
NetLayout	Graphical layout for the network. Possible options are circle for circle layout, sphere for 3D sphere, fr for Fruchterman-Reinhold, and lgl for the LGL algorithm. Refer to igraph manual for more details on the layout options.
ShowNodes	Logical. Whether to show network nodes.
VertexLabelCex	Scaling factor for the vertex labels.
VertexSize	Size of the vertices.

Value

A multi-omics network figure.

Examples

```
set.seed(123)
w <- rnorm(5)
w <- w/sqrt(sum(w^2))
abar <- getAbar(w, P1 = 2, FeatureLabel = NULL)
modules <- getMultiOmicsModules(abar, P1 = 2, CutHeight = 0.5)
x <- cbind(geneExpr[ ,seq_len(2)], mirnaExpr[ , seq_len(3)])
corr <- cor(x)

plotMultiOmicsNetwork(abar, corr, modules, ModuleIdx = 1, P1 = 2)
```

Index