Package 'GenomicSuperSignature'

September 21, 2021

Title Interpretation of RNA-seq experiments through robust, efficient comparison to public databases

Version 1.0.1

Date 2021-05-27

Description This package contains the index, which is the Replicable and interpretable Axes of Variation (RAV) extracted from public RNA sequencing datasets by clustering and averaging top PCs. This index, named as RAVindex, is further annotated with MeSH terms and GSEA. Functions to connect PCs from new datasets to RAVs, extract interpretable annotations, and provide intuitive visualization, are implemented in this package. Overall, this package enables researchers to analyze new data in the context of existing databases with minimal computing resources.

Depends R (>= 4.0), SummarizedExperiment

Imports ComplexHeatmap, ggplot2, methods, S4Vectors, Biobase, ggpubr, dplyr, plotly, BiocFileCache, grid, flextable

Suggests knitr, rmarkdown, devtools, roxygen2, pkgdown, usethis, BiocStyle, testthat, forcats, stats, wordcloud, circlize, EnrichmentBrowser, clusterProfiler, msigdbr, cluster, RColorBrewer, reshape2, tibble, BiocManager, bcellViper, readr, utils

License Artistic-2.0

biocViews Transcriptomics, SystemsBiology, PrincipalComponent, RNASeq, Sequencing, Pathways, Clustering

Encoding UTF-8

LazyData true

RoxygenNote 7.1.1.9000

VignetteBuilder knitr

URL https://github.com/shbrief/GenomicSuperSignature

BugReports https://github.com/shbrief/GenomicSuperSignature/issues

Collate 'GenomicSignatures-class.R' 'GenomicSignatures-methods.R'
'PCAGenomicSignatures-methods.R' 'annotatePC.R'
'annotatePCcluster.R' 'buildAvgLoading.R' 'calculateScore.R'
'data.R' 'drawWordcloud.R' 'extractPC.R' 'findSignature.R'
'findStudiesInCluster.R' 'funcForGSEA.R' 'getModel.R'
'heatmapTable.R' 'plotAnnotedPCA.R' 'plotValidate.R'
'rmNaInf.R' 'rowNorm.R' 'sampleScoreHeatmap.R'
'subsetEnrichedPathways.R' 'utils.R' 'validate.R'
'validatedSignatures.R'

git_url https://git.bioconductor.org/packages/GenomicSuperSignature

git_branch RELEASE_3_13

git_last_commit 56891d8

git_last_commit_date 2021-05-27

Date/Publication 2021-09-21

Author Sehyun Oh [aut, cre],
Levi Waldron [aut],
Sean Davis [aut]

Maintainer Sehyun Oh <shbrief@gmail.com>

R topics documented:

`.calculateSilhouetteWidth`

Calculate Silhouette Information of RAVs

Description

The silhouette value is a measure of how similar an object is to its own cluster (cohesion) compared to other clusters (separation). The silhouette width ranges from -1 to +1, where a high value indicates that the object is well matched to its own cluster and poorly matched to neighboring clusters.

Usage

```
.calculateSilhouetteWidth(dat, kmeansRes)
```

Arguments

`dat`	A matrix with all the top PCs from training data to be clustered.
`kmeansRes`	Output from `stats::kmeans`.

Value

Silhouette-class object, which is an n x 3 matrix with attributes.

See Also

`kmeans`

.loadingCor *Validating new dataset*

Description

Validating new dataset

Usage

```
.loadingCor(dataset, avgLoading, method = "pearson", scale = FALSE)
```

Arguments

dataset	A gene expression profile to be validated. Different classes of objects can be used including ExpressionSet, SummarizedExperiment, RangedSummarizedExperiment, or matrix. Rownames (genes) should be in symbol format. If it is a matrix, genes should be in rows and samples in columns.
avgLoading	A matrix with genes by RAVs.
method	A character string indicating which correlation coefficient is to be computed. One of "pearson" (default), "kendall", or "spearman": can be abbreviated.
scale	Default is FALSE. If it is set to TRUE, dataset will be row normalized by rowNorm function.

Value

A matrix of Pearson correlation coefficient (default, defined through method argument) between RAVs (row) and the top 8 PCs from the datasets (column)

.RAVName *Formatting RAV name*

Description

Keep the name with 'k + cluster number + number of PCs + number of unique studies' info during the model construction to make it easy to keep track of them, but at the PCAGenomicSignatures-class object building step, covert them into 'RAV + cluster number'.

Usage

```
.RAVName(x, ...)
```

Arguments

x	PCAGenomicSignatures object
...	Additional arguments for supporting functions.

Value

a character vector

`annotatePC` *Annotate top PCs from the dataset*

Description

This function finds the RAV scored highest with the top PCs of the dataset, including RAVs with the negative average silhouette.

Usage

```
annotatePC(
  PCnum,
  val_all,
  RAVmodel,
  n = 5,
  scoreCutoff = 0.5,
  nesCutoff = NULL,
  simplify = TRUE,
  abs = FALSE,
  trimed_pathway_len = 45
)
```

Arguments

`PCnum`	A numeric vector. PC number of your dataset that you want to get the annotation results. The vector can contain any integer number among `1:8`.
`val_all`	The output from `validate`
`RAVmodel`	The RAVmodel used to generate the input for the argument, `val_all`.
`n`	An integer. Default is 5. The number of the top enriched pathways to print out. If there are fewer than `n` pathways passed the cutoff, it will print out `NA`.
`scoreCutoff`	A numeric value for the minimum correlation. Default is 0.5.
`nesCutoff`	A numeric value for the minimum NES. Default is `NULL` and the suggested value is 3.
`simplify`	A logical. Under default (`TRUE`), the output will be a data frame with the number of column same as the length of `PCnum` argument, and the number of row same as the `n` argument. If it is set to `FALSE`, the output will be a list with the length of `PCnum` argument, where each element is a data frame containing detailed GSEA output of enriched pathways.
`abs`	Default is `FALSE`. If it's set to `TRUE`, the enriched pathways will be listed based on `abs(NES)`.
`trimed_pathway_len`	Positive inter values, which is the display width of pathway names. Default is 45.

Value

A data frame of a list based on the `simplify` argument. Check the output detail above.

Examples

```
data(miniRAVmodel)
library(bcellViper)
data(bcellViper)
val_all <- validate(dset, miniRAVmodel)
annotatePC(2, val_all, miniRAVmodel)
```

`annotateRAV` *Search the top enriched pathways for RAV*

Description

Search the top enriched pathways for RAV

Usage

```
annotateRAV(RAVmodel, ind, n = 5, abs = FALSE)
```

Arguments

`RAVmodel`	PCAGenomicSignatures object.
`ind`	An integer for RAV you want to check the enriched pathways.
`n`	A number of top enriched pathways to output. Default is 5.
`abs`	Default is `FALSE`. If it's set to `TRUE`, the enriched pathways will be listed based on `abs(NES)`.

Value

A data frame with `n` rows and 4 columns; Description, NES, pvalue, and qvalues

Examples

```
data(miniRAVmodel)
annotateRAV(miniRAVmodel, ind = 695)
```

`buildAvgLoading` *Calculate average loadings of each cluster*

Description

Calculate average loadings of each cluster

Usage

```
buildAvgLoading(dat, k, n = 20, cluster = NULL, study = TRUE)
```

Arguments

`dat`	A data frame. Each row represents principle components from different training datasets. Columns are genes used for PCA analysis.
`k`	The number of clusters used for hierarchical clustering
`n`	The number of top principle components from each datasets used for model building. Default is 20.
`cluster`	Provide pre-defined cluster membership of your data.
`study`	Under default (`TRUE`), studies involved in each cluster will be added in the output.

Value

A named list of 6 elements is returned. It contains:

`cluster` A numeric vector on cluster membership of PCs

`size` A integer vector on the size of clusters

`avgLoading` A matrix of average loadings. Columns for clusters and rows for genes

`k` The number of clusters

`n` The number of top PCs used for clustering

`studies` A list of character vector containing studies in each cluster

Examples

```
data(miniAllZ)
data(res_hcut)
res <- buildAvgLoading(miniAllZ, k = 40, cluster = res_hcut$cluster)
```

calculateScore *Calculate the validation score for a new dataset*

Description

Calculate the validation score for a new dataset

Usage

```
calculateScore(dataset, RAVmodel, rescale.after = TRUE)
```

Arguments

dataset	A gene expression profile to be validated. Different classes of objects can be used including ExpressionSet, SummarizedExperiment, RangedSummarizedExperiment, or matrix. Rownames (genes) should be in symbol format. If it is a matrix, genes should be in rows and samples in columns.
RAVmodel	PCAGenomicSignatures object. A matrix of average loadings, an output from buildAvgLoading, can be directly provided.
rescale.after	Under the default (TRUE), the continuous scores are rescaled post assignment, so average loadings have the same standard deviation in different studies. If it is FALSE, the rescaling of column (= dividing by sqrt(sum(x^2)) is done before score assignment.

Value

A list containing the score matrices for input datasets. Scores are assigned to each sample (row) for each cluster (column).

Examples

```
data(miniRAVmodel)
library(bcellViper)
data(bcellViper)
score <- calculateScore(dset, miniRAVmodel)
```

drawWordcloud *Draw wordcloud using the collection of RAVs' MeSH terms*

Description

Plot a word cloud using the remaining MeSH terms in the selected RAV after user-defined filtering.

Usage

```
drawWordcloud(
  RAVmodel,
  ind,
  rm.noise = NULL,
  scale = c(3, 0.5),
  weighted = TRUE
)
```

Arguments

RAVmodel	PCAGenomicSignatures object
ind	An index of the RAV you want to draw wordcloud.
rm.noise	An integer. Under the default (rm.noise=NULL), if cluster size (= s) is smaller than 8, rm.noise = floor(s*0.5). For clusters with >= 8 PCs, rm.noise = 4. If rm.noise = 0, all the MeSH terms in RAV will be used to draw wordcloud.
scale	A scale argument for wordcloud function
weighted	A logical. If TRUE (default), MeSH terms from each study are weighted based on the variance explained by the principle component of the study contributing to a given RAV.

Value

A word cloud with the MeSH terms associated with the given cluster.

Examples

```
data(miniRAVmodel)
drawWordcloud(miniRAVmodel, 1139)
```

---

extractPC *PCA on gene expression profile*

Description

Performs a principal components analysis on the given data matrix and returns the results as an object of class prcomp.

Usage

```
extractPC(x)
```

Arguments

x	a numeric or complex matrix (or data frame) which provides the gene expression data for the principal components analysis. Genes in the rows and samples in the columns.

Value

A prcomp object.

See Also

prcomp

Examples

```
m = matrix(rnorm(100),ncol=5)
extractPC(m)
```

---

findKeywordInRAV *Find the rank of your keyword in the RAV's GSEA annotation*

Description

Once you provide RAVmodel, keyword you're searching for, and the RAV number to this function, it will give you the abs(NES)-based rank of your keyword in the enriched pathways of the target RAV. If can be useful to find out how uniquely your keyword-containing pathways are represented.

Usage

```
findKeywordInRAV(RAVmodel, keyword, ind, n = NULL, includeTotal = TRUE)
```

Arguments

RAVmodel	PCAGenomicSignatures-object.
keyword	A character vector. If you are searching for multiple keywords at the same time, use paste with collapse="|" argument.
ind	An integer. The RAV number you want to check.
n	An integer. The number of top enriched pathways (based on abs(NES)) to search. Under default (NULL), all the enriched pathways are used.
includeTotal	Under the default condition (TRUE), the total number of enriched pathways will be also printed out as a part of the output.

Value

A character containing the rank of keyword-containing pathways (separated by |), followed by the total number of enriched pathways in parenthesis.

Examples

```
data(miniRAVmodel)
findKeywordInRAV(miniRAVmodel, "Bcell", ind = 695)
```

`findSignature` *Find the RAVs with the keyword-containing enriched pathways*

Description

This function finds RAVs containing the keyword you provide. If you provide "the number of keyword-containing pathways per RAV" in argument `k`, it will give you the RAV number.

Usage

```
findSignature(RAVmodel, keyword, n = 5, k = NULL)
```

Arguments

`RAVmodel`	PCAGenomicSignatures-object
`keyword`	A character vector. If you are searching for multiple keywords at the same time, use `paste` with `collapse="|"` argument.
`n`	The number of top ranked (based on abs(NES)) pathways you want to search your keyword
`k`	The number of keyword-containing pathways you want to get the RAV number. Under default (`NULL`), the output will be a data frame with two columns: '# of keyword-containing pathways' and 'Freq'. If you assign the value for this argument, the output will be an integer vector containing the RAV index.

Value

A data frame or integer vector depending on the parameter `k`.

Examples

```
data(miniRAVmodel)
library(bcellViper)
data(bcellViper)
findSignature(miniRAVmodel, "Bcell")
findSignature(miniRAVmodel, "Bcell", k = 5)
```

findStudiesInCluster *Find the studies contributing each RAV*

Description

Find the studies contributing each RAV

Usage

```
findStudiesInCluster(RAVmodel, ind = NULL)
```

Arguments

RAVmodel	PCAGenomicSignatures object.
ind	A numeric vector containing the RAV indexes. Under the default (NULL), studies associated with all the RAV indexes will be returned as a list.

Value

A list of character vectors. Under the default condition (ind = NULL), all the RAVs will be checked for their contributing studies and the length of the list will be same as the number of RAVs (= metadata(x)$k). If you provide the ind argument, studies associated with only the specified RAVs will be returned.

Note

Mainly used for model building, within buildAvgLoading.

Examples

```
data(miniRAVmodel)
findStudiesInCluster(miniRAVmodel, 1076)
```

GenomicSignatures-class *Virtual class inherited from SummarizedExperiment*

Description

GenomicSignatures is a virtual class inherited from SummarizedExperiment and hosts GenomicSignatures models built from different dimensional reduction methods. Currently, PCA-based model, called PCAGenomicSignatures, is available.

Arguments

x	A GenomicSignatures-class object
value	See details.

Details

GenomicSignatures

GenomicSignatures-methods

Methods and accesors for GenomicSignatures *object*

Description

The default contents of GenomicSignatures object, with a set of getter and setter generic functions, which extract either the assay, colData, or metadata slots of a GenomicSignatures-class object. When you create this object, colData$studies should be populated before adding any information in trainingData slot.

Usage

```
## S4 method for signature 'GenomicSignatures'
RAVindex(x)

## S4 method for signature 'GenomicSignatures'
geneSets(x)

## S4 method for signature 'GenomicSignatures'
updateNote(x)

## S4 replacement method for signature 'GenomicSignatures'
geneSets(x) <- value

## S4 replacement method for signature 'GenomicSignatures'
updateNote(x) <- value
```

Arguments

x	A GenomicSignatures object
value	See details.

Details

- assay(x) : RAVindex (= avgLoadings) containing genes x RAVs
- metadata(x) : Metadata associated with RAVindex building process
- colData(x) : Information on RAVs

Value

A GenomicSignatures object for the constructor

Setters

Setter method values (i.e., `function(x) <-value`):

- metadata<- : Assign metadata
- coldata<- : Assign extra information associated with RAVs
- geneSets<- : A character vector containing the name of gene sets used to annotate average loadings
- updateNote<- : A character vector. Describes the main feature of a model construction

Getters

- RAVindex : Equivalent to `assays(x)$RAVindex`
- geneSets : Access the `metadata(x)$geneSets` slot
- updateNote : Access the `metadata(x)$updateNote` slot

Examples

```
data(miniRAVmodel)
miniRAVmodel
```

`getModel` *Download a PCAGenomicSignatures model*

Description

Download a PCAGenomicSignatures model

Usage

```
getModel(prior = c("C2", "PLIERpriors"), load = TRUE)
```

Arguments

`prior`	The name of gene sets used to annotate PCAGenomicSignatures. Currently there are two available options. • `C2` : MSigDB C2 (curated gene sets) • `PLIERpriors` : bloodCellMarkersIRISDMAP, svmMarkers, and canonicalPathways
`load`	Default is `TRUE`. If it's set to `FALSE`, the models are just downloaded to cache but not loaded into memory.

Value

File cache location or PCAGenomicSignatures object loaded from it.

Examples

```
z = getModel("C2")
```

`heatmapTable` *Validation result in heatmap format*

Description

This function subsets `validate` outputs with different criteria and visualize it in a heatmap-like table.

Usage

```
heatmapTable(
  val_all,
  ind = NULL,
  num.out = 5,
  scoreCutoff = NULL,
  swCutoff = NULL,
  clsizeCutoff = NULL,
  breaks = c(0, 0.5, 1),
  colors = c("white", "white smoke", "red"),
  column_title = NULL,
  row_title = NULL,
  whichPC = NULL,
  ...
)
```

Arguments

`val_all`	An output matrix from `validate` function with the parameter `level = "max"`. Subset of this matrix is plotted as a heatmap using `Heatmap`
`ind`	An integer vector. If this parameter is provided, the other parameters, `num.out`, `scoreCutoff`, `swCutoff`, will be ignored and the heatmap table containing only the provided index will be printed.
`num.out`	A number of highly validated RAVs to output. Default is 5. If any of the cutoff parameters are provided, `num.out` or the number of filtered RAVs, whichever smaller, will be chosen.
`scoreCutoff`	A numeric value for the minimum correlation. If `val_all` input is from multiple studies, the default is 0.7 and this is the only cutoff criteria considered: `swCutoff` and `clsizeCutoff` will be ignored.

swCutoff	A numeric value for the minimum average silhouette width.
clsizeCutoff	A integer value for the minimum cluster size.
breaks	A numeric vector of length 3. Number represents the values assigned to three colors. Default is c(0,0.5,1).
colors	A character vector of length 3. Each represents the color assigned to three breaks. Default is c("white","white smoke","red").
column_title	A character string. Provide the column title.
row_title	A character string. Provide the row title.
whichPC	An integer value between 1 and 8. PC number of your data to check the validated signatures with. Under the default (NULL), it outputs top scored signatures with any PC of your data.
...	any additional argument for Heatmap

Value

A heatmap displaying the subset of the validation result that met the given cutoff criteria. If val_all input is from a single dataset, the output heatmap will contain both score and average silhouette width for each cluster.

If val_all input is from multiple studies, the output heatmap's rows will represent each study and the columns will be RAVs, which meet scoreCutoff for any of the input studies.

Examples

```
data(miniRAVmodel)
library(bcellViper)
data(bcellViper)
val_all <- validate(dset, miniRAVmodel)
heatmapTable(val_all, swCutoff = 0)
```

---

makeGeneList *Order genes in loading vectors*

Description

This function takes Z matrix (= averaged loadings) and orders the genes in each loading vector (= RAV) in a descending manner.

Usage

```
makeGeneList(LoadingMatrix, LoadingVector = NULL, abs = TRUE)
```

Arguments

LoadingMatrix	An avgloading matrix. Rows represent genes and columns represent clusters of principle components
LoadingVector	A list of column names or indexes of LoadingMatrix you want to check. Default is NULL, under which the function takes the all column names of LoadingMatrix
abs	Under the defaul condition (TRUE), this function will create a gene list based on the absolute value.

Value

A list of loadings selected by LoadingVector, where all the genes in each loading are listed in descending order.

---

meshTable *Build a two-column word/frequency table*

Description

Build a two-column word/frequency table

Usage

```
meshTable(RAVmodel, ind, rm.noise = NULL, weighted = TRUE)
```

Arguments

RAVmodel	A PCAGenomicSignatures object
ind	An index of RAV
rm.noise	An integer. Under the default (rm.noise=NULL), if cluster size (= s) is smaller than 8, rm.noise = floor(s*0.5). For clusters with >= 8 PCs, rm.noise = 4. If rm.noise = 0, all the MeSH terms in RAV will be used to draw wordcloud.
weighted	A logical. If TRUE, MeSH terms from each study are weighted based on the variance explained by the principle component of the study contributing a give RAV. Default is TRUE.

Value

A table with two columns, word and freq. MeSH terms in the defined RAV (by ind argument) is ordered based on their frequency.

Examples

```
data(miniRAVmodel)
meshTable(miniRAVmodel,1139)
```

miniAllZ *Subset of allZ matrix constructed from 8 CRC training datasets*

Description

Eight colorectal cancer microarray datasets were used to build RAVmodel and the intermediate file containing genes and top PCs from each dataset is named as `allZ`. Hierarchical clustering result of `allZ` is saved as `res_hcut`. For demonstration, we subset the `allZ` matrix with the first 100 genes, which is named as `miniAllZ`.

Usage

```
miniAllZ
```

Format

A matrix with 100 genes and 160 PCs from 8 training datasets.

Author(s)

Sehyun Oh <shbrief@gmail.com>

Source

https://github.com/shbrief/model_building/tree/main/RAVmodel_8CRC

miniRAVmodel *RAVmodel from 536 studies, annotated with MSigDB C2*

Description

A object providing a miniature version of RAVmodel_C2 (PCAGenomicSignatures object constructed from 536 studies and annotated with MSigDB C2).

Usage

```
miniRAVmodel
```

Format

PCAGenomicSignatures

Author(s)

Sehyun Oh <shbrief@gmail.com>

miniTCGA *Subset of TCGA-COAD and TCGA-BRCA RNA sequencing datasets*

Description

TCGA-COAD and TCGA-BRCA RNA sequencing data were acquired using `GSEABenchmarkeR::loadEData` and log-transformed. Conversion from EntrezID to gene symbol was done with `EnrichmentBrowser::idMap`. Only 8 samples from each dataset are kept.

Usage

```
miniTCGA
```

Format

A list containing two SummarizedExperiment objects.

Author(s)

Sehyun Oh <shbrief@gmail.com>

msigdb_gsea *MSigDB GSEA results*

Description

MSigDB GSEA results

Usage

```
msigdb_gsea(
  ind,
  RAVmodel,
  category = "C2",
  n = NULL,
  pvalueCutoff = 0.5,
  minGSSize = 10,
  maxGSSize = 500,
  pAdjustMethod = "BH",
  verbose = FALSE,
  seed = FALSE,
  by = "fgsea",
  geneSets = NULL
)
```

Arguments

ind	An interger. Index of RAV to apply GSEA.
RAVmodel	PCAGenomicSignature object.
category	A character vector representing MSigDB category. Options are "H", "C1", "C2"(default), "C3", "C4", "C5", "C6", and "C7"
n	An interger. The number of top and bottom enriched pathways to plot. Default is NULL, which prints out all the pathways enriched under pvalueCutoff.
pvalueCutoff	Cutoff for both pvalue and p.adjust. Default is 0.5.
minGSSize	A mininum size of gene set to be analyzed
maxGSSize	A maximum size of gene set to be analyzed
pAdjustMethod	p-value adjustment methods, which will be used as an input for method argument of p.adjust function. Available options are "holm", "hochberg", "hommel", "bonferroni", "BH"(default), "BY", "fdr", "none".
verbose	Logical. Default is FALSE.
seed	Logical. Default is FALSE.
by	Available options are c("fgsea","DOSE"). Default "fgsea".
geneSets	Custom genesets to use with MSigDB genesets. It should be in a named list format.

Value

Barplot of GSEA output. Top and bottom n genesets based on NES are plotted and qvalues are denoted by color.

PCAGenomicSignatures *Construct* PCAGenomicSignatures *object*

Description

The default contents of PCAGenomicSignatures object, with a set of accessors and setter generic functions, which extract either the assay, colData, metadata, or trainingData slots of a PCAGenomicSignatures-class object. When you create this object, colData$studies should be populated before adding any information in trainingData slot

Usage

```
PCAGenomicSignatures(...)
```

Arguments

...	Additional arguments for supporting functions.

Details

- RAVindex(x) : RAVindex (= avgLoadings) containing genes x RAVs
- metadata(x)$cluster : A vector of integers (from 1:k) indicating the cluster to which each point is allocated.
- metadata(x)$size : The number of PCs in each cluster.
- metadata(x)$k : The number of RAVs.
- metadata(x)$n : The number of top PCs from each dataset.
- metadata(x)$geneSets : Name of the prior gene sets used to annotate average loadings.
- colData(x)$studies : A list of character vectors containing studies contributing to each PC cluster.
- colData(x)$silhouetteWidth : A numeric array of average silhouette widths of each clusters
- colData(x)$gsea : A list of data frames. Each element is a subset of outputs from `clusterProfiler::GSEA` function.

Value

PCAGenomicSignatures object with multiple setters or accessors

Slots

`trainingData` A `DataFrame` class object for metadata associated with training data

Setters

Setter method values (i.e., `function(x) <-value`):

- geneSets<- : A character vector containing the name of gene sets used to annotate average loadings
- studies<- : A list of character vectors containing gene sets used to annotate average loadings
- gsea<- : A list of data frames. Each element is a subset of output from `gseaResult` objects.
- metadata<- : A `list` object of metadata
- '$<-' : A vector to replace the indicated column in `colData`

Accessors

All the accessors inherited from `SummarizedExperiment` are available and the additional accessors for `PCAGenomicSignatures` specific data are listed below.

- RAVindex : Equivalent to the `assay(x)`
- geneSets : Access the `metadata(x)$geneSets` slot
- studies : Access the `colData(x)$studies` slot
- gsea : Access the `colData(x)$gsea`
- '$' : Access a column in `colData`
- trainingData : Access the `trainingData` slot
- mesh : Access the `trainingData(x)$MeSH` slot
- PCAsummary : Access the `trainingData(x)$PCAsummary` slot

Examples

```
data(miniRAVmodel)
miniRAVmodel
```

`PCAGenomicSignatures-class` *PCAGenomicSignatures-class*

Description

PCA-based `GenomicSignatures-class`.

Arguments

`x`	A `GenomicSignatures-class` object
`value`	See details.

Details

PCAGenomicSignatures

Slots

`trainingData` A `DataFrame` class object for metadata associated with training data

Examples

```
data(miniRAVmodel)
miniRAVmodel
```

`PCAGenomicSignatures-methods` *Methods and accesors for* `PCAGenomicSignatures` *object*

Description

The default contents of `PCAGenomicSignatures` object, with a set of accessor and setter generic functions, which extract either the `assay`, `colData`, `metadata`, or `trainingData` slots of a `PCAGenomicSignatures-class` object. When you create this object, `colData$studies` should be populated before adding any information in `trainingData` slot

Usage

```
## S4 replacement method for signature 'PCAGenomicSignatures'
studies(x) <- value

## S4 replacement method for signature 'PCAGenomicSignatures'
silhouetteWidth(x) <- value

## S4 replacement method for signature 'PCAGenomicSignatures'
gsea(x) <- value

## S4 replacement method for signature 'PCAGenomicSignatures'
trainingData(x) <- value

## S4 replacement method for signature 'PCAGenomicSignatures'
mesh(x) <- value

## S4 replacement method for signature 'PCAGenomicSignatures'
PCAsummary(x) <- value

## S4 method for signature 'PCAGenomicSignatures'
studies(x)

## S4 method for signature 'PCAGenomicSignatures'
silhouetteWidth(x)

## S4 method for signature 'PCAGenomicSignatures'
gsea(x)

## S4 method for signature 'PCAGenomicSignatures'
trainingData(x)

## S4 method for signature 'PCAGenomicSignatures'
mesh(x)

## S4 method for signature 'PCAGenomicSignatures'
PCAsummary(x)

## S4 method for signature 'PCAGenomicSignatures'
show(object)
```

Arguments

value	See details.
object, x	A PCAGenomicSignatures object

Details

- RAVindex(x) : RAVindex (= avgLoadings) containing genes x RAVs

- metadata(x)$cluster : A vector of integers (from 1:k) indicating the cluster to which each PC is allocated.
- metadata(x)$size : The number of PCs in each cluster.
- metadata(x)$k : The number of RAVs.
- metadata(x)$n : The number of top PCs from each dataset.
- metadata(x)$geneSets : Name of the prior gene sets used to annotate average loadings.
- colData(x)$studies : A list of character vectors containing studies contributing to each PC cluster.
- colData(x)$gsea : A list of data frames. Each element is a subset of outputs from `clusterProfiler::GSEA` function.

Value

PCAGenomicSignatures object with multiple setters or accessors

Slots

`trainingData` A `DataFrame` class object for metadata associated with training data

Setters

Setter method values (i.e., `function(x) <-value`):

- geneSets<- : A character vector containing the name of gene sets used to annotate average loadings
- studies<- : A list of character vectors containing gene sets used to annotate average loadings
- gsea<- : A list of `gseaResult` objects.
- metadata<- : A `list` object of metadata
- '$<-' : A vector to replace the indicated column in `colData`

Accessors

All the accessors inherited from `SummarizedExperiment` are available and the additional accessors for `PCAGenomicSignatures` specific data are listed below.

- RAVindex : Equivalent to the `assay(x)`
- geneSets : Access the `metadata(x)$geneSets` slot
- studies : Access the `colData(x)$studies` slot
- gsea : Access the `colData(x)$gsea`
- '$' : Access a column in `colData`
- trainingData : Access the `trainingData` slot
- mesh : Access the `trainingData(x)$MeSH` slot
- PCAsummary : Access the `trainingData(x)$PCAsummary` slot

Examples

```
data(miniRAVmodel)
miniRAVmodel
```

PCinRAV *Extract the list of PCs in a cluster*

Description

A RAV model contain clusters of PCs from individual studies. This function extracts the names of the original PCs from the RAV model given the index in the RAV model.

Usage

```
PCinRAV(RAVmodel, ind)
```

Arguments

RAVmodel	A PCAGenomicSignatures object
ind	An index of RAV

Value

A character vector of PC/study names

Examples

```
data(miniRAVmodel)
PCinRAV(miniRAVmodel,695)
```

---

plotAnnotatedPCA *Two-dimentional PCA plot with the PC annotation*

Description

Two-dimentional PCA plot with the PC annotation

Usage

```
plotAnnotatedPCA(
  dataset,
  RAVmodel,
  PCs,
  val_all = NULL,
  scoreCutoff = 0.5,
  nesCutoff = NULL,
  color_by = NULL,
  color_lab = NULL,
  trimed_pathway_len = 45
)
```

Arguments

dataset	A gene expression profile to be validated. Different classes of objects can be used including ExpressionSet, SummarizedExperiment, RangedSummarizedExperiment, or matrix. Rownames (genes) should be in symbol format. If it is a matrix, genes should be in rows and samples in columns.
RAVmodel	PCAGenomicSignatures-class object
PCs	A numeric vector length of 2. It should be between 1 and 8.
val_all	The output from validate
scoreCutoff	A numeric value for the minimum correlation. Default 0.5.
nesCutoff	A numeric value for the minimum NES. Default is NULL and the suggested value is 3.
color_by	A named vector with the feature you want to color by. Name should be match with the sample names of the dataset.
color_lab	A name for color legend. If this argument is not provided, the color legend will be labeled as "Color By" by default.
trimed_pathway_len	Positive inter values, which is the display width of pathway names. Default is 45.

Value

Scatter plot and the table with annotation. If enriched pathway didn't pass the scoreCutoff the table will be labeled as "No significant pathways". If any enriched pathway didn't pass the nesCutoff, it will labeled as NA.

Examples

```
data(miniRAVmodel)
library(bcellViper)
data(bcellViper)
## Not run:
plotAnnotatedPCA(exprs(dset), miniRAVmodel, PCs = c(1,2))
```

```
## End(Not run)
```

---

plotValidate *Plot validation results in an interactive graph*

Description

There are three main information on the graph:

- x-axis : Pearson correlation coefficient. Higher value means that test dataset and RAV is more tightly associated with.
- y-axis : Silhouette width representing the quality of RAVs.
- size : The number of studies in each RAV. (= cluster size)
- color : Test dataset's PC number that validate each RAV. Because we used top 8 PCs of the test dataset, there are 8 categories.

Usage

```
plotValidate(
  val_all,
  minClusterSize = 2,
  swFilter = FALSE,
  minSilhouetteWidth = 0,
  interactive = FALSE,
  minClSize = NULL,
  maxClSize = NULL,
  colorPalette = "Dark2"
)
```

Arguments

`val_all`	Output from validate function.
`minClusterSize`	The minimum size of clusters to be included in the plotting. Default value is 2, so any single-element clusters are excluded.
`swFilter`	If `swFilter=TRUE`, only RAV above the cutoff, defined through `minSilhouetteWidth` argument will be plotted. Default is `swFilter=FALSE`
`minSilhouetteWidth`	A minimum average silhouette width to be plotted. Only effective under `swFilter=TRUE` condition. Default is 0.
`interactive`	If set to `TRUE`, the output will be interactive plot. Default is `FALSE`.
`minClSize`	The minimum number of PCs in the clusters you want.
`maxClSize`	The maximum number of PCs in the clusters you want.
`colorPalette`	Default is `Dark2`. For other color options, please check `scale_color_brewer`

Value

a ggplot object

Examples

```
data(miniRAVmodel)
library(bcellViper)
data(bcellViper)
val_all <- validate(dset, miniRAVmodel)
plotValidate(val_all)
```

---

res_hcut *Subset of allZ matrix constructed from 8 CRC training datasets*

Description

Eight colorectal cancer microarray datasets were used to build RAVmodel and the intermediate file containing genes and top PCs from each dataset is named as `allZ`. Hierarchical clustering result of `allZ` is saved as `res_hcut`.

Usage

```
res_hcut
```

Format

`hclust` object from `factoextra::hcut` function.

Author(s)

Sehyun Oh <shbrief@gmail.com>

---

rmNaInf *Remove rows with missing and Inf values from a matrix*

Description

Remove rows with missing and Inf values from a matrix

Usage

```
rmNaInf(x)
```

Arguments

x	A numeric matrix.

Value

The updated input matrix where rows with NA and Inf values are removed.

Examples

```
m = matrix(rnorm(100),ncol=10)
m[1,1] = NA

m1 = rmNaInf(m)
dim(m1)
```

`rowNorm` *z-score each row of the data*

Description

z-score each row of the data

Usage

```
rowNorm(x)
```

Arguments

`x`	A gene-expression matrix with genes in rows and samples in columns

Value

a matrix with each row mean centered and scaled by rowwise sd

References

https://github.com/shbrief/PLIER/blob/master/R/Allfuncs.R

Examples

```
x = matrix(rnorm(100),nc=10)
y = rowNorm(x)
apply(y,1,mean)
```

run_gsea *GSEA on pre-ordered gene lists*

Description

This function is a wrapper of GSEA function, making it applicable to a list of gene lists. Set seed for reproducible result.

Usage

```
run_gsea(
  geneList,
  TERM2GENE,
  TERM2NAME,
  minGSSize = 10,
  maxGSSize = 500,
  pvalueCutoff = 0.05,
  verbose = FALSE,
  ...
)
```

Arguments

geneList	A list of genes ordered by rank
TERM2GENE	User input annotation of TERM TO GENE mapping, a data frame of 2 column with term and gene
TERM2NAME	User input of TERM TO NAME mapping, a data.frame of 2 column with term and name. Optional.
minGSSize	A mininum size of gene set to be analyzed
maxGSSize	A maximum size of gene set to be analyzed
pvalueCutoff	p-value cutoff
verbose	Logical. Default is FALSE
...	Any additional argument inherited from GSEA.

Value

A list of gseaResult objects. If there is no enrichment result, NA will be returned.

`sampleScoreHeatmap` *Plot heatmap of the sample scores*

Description

Plot heatmap of the sample scores

Usage

```
sampleScoreHeatmap(
  score,
  dataName,
  modelName,
  cluster_rows = TRUE,
  cluster_columns = TRUE,
  show_row_names = TRUE,
  show_column_names = TRUE,
  row_names_gp = 0.7,
  column_names_gp = 5,
  ...
)
```

Arguments

`score`	An output from `calculateScore` function, which is a matrix with samples (row) and PrcompClusters (column) If it is a simple vector, it will be converted to a one-column matrix.
`dataName`	Title on the row. The name of the dataset to be scored.
`modelName`	Title on the column. The RAVmodel used for scoring.
`cluster_rows`	A logical. Under the default (`TRUE`), rows will be clustered.
`cluster_columns`	A logical. Under the default (`TRUE`), columns will be clustered.
`show_row_names`	Whether show row names. Default is `TRUE`, showing the row name.
`show_column_names`	Whether show column names. Default is `TRUE`, showing the column name.
`row_names_gp`	Graphic parameters for row names. The default is 0.7.
`column_names_gp`	Graphic parameters for column names. The default is 5.
`...`	Any additional argument for `Heatmap`

Value

A heatmap of the sample score. Rows represent samples and columns represent RAVs.

Examples

```
data(miniRAVmodel)
library(bcellViper)
data(bcellViper)
score <- calculateScore(dset, miniRAVmodel)
sampleScoreHeatmap(score, dataName="bcellViper", modelName="miniRAVmodel")
```

`subsetEnrichedPathways` *Subset enriched pathways of RAV*

Description

Subset enriched pathways of RAV

Usage

```
subsetEnrichedPathways(RAVmodel, ind = NULL, n = 10, both = FALSE)
```

Arguments

`RAVmodel`	PCAGenomicSignatures object. Also an output from `GSEA` can be used.
`ind`	A numeric vector containing the RAV number you want to check enriched pathways. If not specified, this function returns results from all the RAVs.
`n`	The number of top and bottom pathways to be selected based on normalized enrichment score (NES).
`both`	Default is `FALSE`, where only the top `n` pathways will be printed. If it is set to `TRUE`, the output will contain both top and bottom `n` pathways.

Value

A DataFrame with top and bottom `n` pathways from the enrichment results.

Examples

```
data(miniRAVmodel)

# all RAVS in model
subsetEnrichedPathways(miniRAVmodel,n=5)

# only a specific RAV (note the colnames above)
subsetEnrichedPathways(miniRAVmodel,ind=695,n=5)
```

subsetGSEA *Subset GSEA output*

Description

Subset GSEA output

Usage

```
subsetGSEA(gseaRes, n = 20)
```

Arguments

gseaRes	An output from GSEA
n	A number of output to keep based on the abs(NES)

Value

a subset the original gseaRes object

validate *Validate new datasets*

Description

Validate new datasets

Usage

```
validate(
  dataset,
  RAVmodel,
  method = "pearson",
  maxFrom = "PC",
  level = "max",
  scale = FALSE
)
```

Arguments

dataset	Single or a named list of SummarizedExperiment (RangedSummarizedExperiment, ExpressionSet or matrix) object(s). Gene names should be in 'symbol' format. Currently, each dataset should have at least 8 samples.
RAVmodel	PCAGenomicSignatures object.

method	A character string indicating which correlation coefficient is to be computed. One of "pearson" (default), "kendall", or "spearman": can be abbreviated.
maxFrom	Select whether to display the maximum value from dataset's PCs or avgLoadings. Under the default (maxFrom="PC"), the maximum correlation coefficient from top 8 PCs for each avgLoading will be selected as an output. If you choose (maxFrom="avgLoading"), the avgLoading with the maximum correlation coefficient with each PC will be in the output.
level	Output format of validated result. Two options are available: c("max","all"). Default is "max", which outputs the matrix containing only the maximum coefficient. To get the coefficient of all 8 PCs, set this argument as "all". level = "all" can be used only for one dataset.
scale	Default is FALSE. If it is set to TRUE, dataset will be row normalized by rowNorm function.

Value

A data frame containing the maximum pearson correlation coefficient between the top 8 PCs of the dataset and pre-calculated average loadings (in row) of training datasets (score column). It also contains other metadata associated with each RAV: PC for one of the top 8 PCs of the dataset that results in the given score, sw for the average silhouette width of the RAV, cl_size for the size of each RAV.

If the input for dataset argument is a list of different datasets, each row of the output represents a new dataset for test, and each column represents clusters from training datasets. If level = "all", a list containing the matrices of the pearson correlation coefficient between all top 8 PCs of the datasets and avgLoading.

Examples

```
data(miniRAVmodel)
library(bcellViper)
data(bcellViper)
validate(dset, miniRAVmodel)
validate(dset, miniRAVmodel, maxFrom = "avgLoading")
```

---

validatedSignatures *Validation result in data frame*

Description

Validation result in data frame

Usage

```
validatedSignatures(
  val_all,
  num.out = 5,
  scoreCutoff = NULL,
  swCutoff = NULL,
  clsizeCutoff = NULL,
  indexOnly = FALSE,
  whichPC = NULL
)
```

Arguments

val_all	An output matrix from validate function. If this input is from multiple datasets, only scoreCutoff argument will be considered and other inputs will be ignored.
num.out	A number of highly validated RAVs to output. Default is 5. If any of the cutoff parameters are provided, num.out or the number of filtered RAVs, whichever smaller, will be chosen.
scoreCutoff	A numeric value for the minimum correlation. For multi-studies case, the default is 0.7.
swCutoff	A numeric value for the minimum average silhouette width.
clsizeCutoff	An integer value for the minimum cluster size.
indexOnly	A logical. Under the default (= FALSE), the detailed information on validated RAVs, such as score, average silhouette width, cluster size, is printed. If it is set TRUE, only the RAV number will be printed.
whichPC	An integer value between 1 and 8. PC number of your data to check the validated signatures with. Under the default (NULL), it outputs top scored signatures with any PC of your data.

Value

A subset of the input matrix, which meets the given condition.

Examples

```
data(miniRAVmodel)
library(bcellViper)
data(bcellViper)
val_all <- validate(dset, miniRAVmodel)
validatedSignatures(val_all, num.out = 3, scoreCutoff = 0)
```

Index