# Package 'pathwayPCA'

December 13, 2018

**Type** Package

**Title** Integrative Pathway Analysis with Modern PCA Methodology and Gene Selection

**Version** 1.0.0.0

**Maintainer** Gabriel Odom <gabriel.odom@med.miami.edu>

**Description** Apply the Supervised PCA and Adaptive, Elastic-Net, Sparse PCA methods to extract principal components from each pathway. Use these pathway-specific principal components as the design matrix relating the response to each pathway. Return the model fit statistic p-values, and adjust these values for False Discovery Rate. Return a data frame of the pathways sorted by their adjusted p-values. This package has corresponding vignettes hosted in the "User Guides" page of <https://gabrielodom.github.io/pathwayPCA/index.html>, and the website for the development information is hosted at <https://github.com/gabrielodom/pathwayPCA>.

**License** GPL-2

**Depends** R (>= 2.10)

**Imports** lars,
methods,
parallel,
survival

**Suggests** knitr,
reshape2,
rmarkdown,
survminer,
tidyverse

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 6.1.1

**Collate** 'CreatePathwayCollection.R'
'createClass_OmicsPath.R'
'createClass_validOmics.R'
'accessClass_OmicsPath.R'
'createClass_OmicsSurv.R'
'accessClass_OmicsSurv.R'
'accessClass_OmicsRegCateg.R'
'createClass_OmicsCateg.R'
'createClass_OmicsReg.R'

'accessClass_OmicsPathData.R'
'accessClass_pathwayCollection.R'
'accessClass_pcOut.R'
'aesPC_calculate_AESPCA.R'
'aesPC_calculate_LARS.R'
'aesPC_extract_OmicsPath_PCs.R'
'aesPC_permtest_CoxPH.R'
'aesPC_permtest_GLM.R'
'aesPC_permtest_LM.R'
'aesPC_unknown_matrixNorm.R'
'aesPC_wrapper.R'
'createOmics_All.R'
'createOmics_CheckAssay.R'
'createOmics_CheckPathwayCollection.R'
'createOmics_CheckSampleIDs.R'
'createOmics_JoinPhenoAssay.R'
'createOmics_TrimPathwayCollection.R'
'createOmics_Wrapper.R'
'data_colonSubset.R'
'data_genesetSubset.R'
'data_wikipathways.R'
'pathwayPCA.R'
'printClass_Omics_All.R'
'printClass_pathwayCollection.R'
'superPC_model_CoxPH.R'
'superPC_model_GLM.R'
'superPC_model_LS.R'
'superPC_model_tStats.R'
'superPC_model_train.R'
'superPC_modifiedSVD.R'
'superPC_optimWeibullParams.R'
'superPC_optimWeibull_pValues.R'
'superPC_pathway_tControl.R'
'superPC_pathway_tScores.R'
'superPC_permuteSamples.R'
'superPC_wrapper.R'
'utils_adjust_and_sort_pValues.R'
'utils_load_test_data_onto_PCs.R'
'utils_multtest_pvalues.R'
'utils_read_gmt.R'
'utils_transpose_assay.R'
'utils_write_gmt.R'

**VignetteBuilder** knitr

**URL** https://gabrielodom.github.io/pathwayPCA/;
https://github.com/gabrielodom/pathwayPCA

**BugReports** https://github.com/gabrielodom/pathwayPCA/issues

# R topics documented:

---

`AESPCA_pVals` *Test pathway association with AES-PCA*

---

## Description

Given a supervised `OmicsPath` object (one of `OmicsSurv`, `OmicsReg`, or `OmicsCateg`), extract the first $k$ adaptive, elastic-net, sparse principal components (PCs) from each pathway-subset of the features in the -Omics assay design matrix, test their association with the response matrix, and return a data frame of the adjusted $p$-values for each pathway.

## Usage

```
AESPCA_pVals(object, numPCs = 1, numReps = 0L, parallel = FALSE,
  numCores = NULL, asPCA = FALSE, adjustpValues = TRUE,
  adjustment = c("Bonferroni", "Holm", "Hochberg", "SidakSS", "SidakSD",
  "BH", "BY", "ABH", "TSBH"), ...)

## S4 method for signature 'OmicsPathway'
AESPCA_pVals(object, numPCs = 1,
  numReps = 1000, parallel = FALSE, numCores = NULL, asPCA = FALSE,
  adjustpValues = TRUE, adjustment = c("Bonferroni", "Holm",
  "Hochberg", "SidakSS", "SidakSD", "BH", "BY", "ABH", "TSBH"), ...)
```

## Arguments

| | |
|---|---|
| object | An object of class OmicsPathway with a response matrix or vector. |
| numPCs | The number of PCs to extract from each pathway. Defaults to 1. |
| numReps | How many permutations to estimate the $p$-value? Defaults to 0 (that is, to estimate the $p$-value parametrically). If numReps > 0, then the non-parametric, permutation $p$-value will be returned based on the number of random samples specified. |
| parallel | Should the computation be completed in parallel? Defaults to FALSE. |
| numCores | If parallel = TRUE, how many cores should be used for computation? Internally defaults to the number of available cores minus 1. |
| asPCA | Should the computation return the eigenvectors and eigenvalues instead of the adaptive, elastic-net, sparse principal components and their corresponding loadings. Defaults to FALSE; this should be used for diagnostic or comparative purposes only. |
| adjustpValues | Should you adjust the $p$-values for multiple comparisons? Defaults to TRUE. |
| adjustment | Character vector of procedures. The returned data frame will be sorted in ascending order by the first procedure in this vector, with ties broken by the unadjusted $p$-value. If only one procedure is selected, then it is necessarily the first procedure. See the documentation for the ControlFDR function for the adjustment procedure definitions and citations. |
| ... | Dots for additional internal arguments. |

## Details

This is a wrapper function for the ExtractAESPCs, PermTestSurv, PermTestReg, and PermTestCateg functions.

Please see our Quickstart Guide for this package: https://gabrielodom.github.io/pathwayPCA/articles/Supplement1-Quickstart_Guide.html

## Value

A results list with class aespcOut. This list has three components: a data frame of pathway details, pathway $p$-values, and potential adjustments to those values (pVals_df); a list of the first numPCs *score* vectors for each pathway (PCs_ls); and a list of the first numPCs feature loading vectors for each pathway (loadings_ls). The $p$-value data frame has columns:

- pathways : The names of the pathways in the Omics* object (given in object@trimPathwayCollection$pathways
- setsize : The number of genes in each of the original pathways (given in the object@trimPathwayCollection$se object).
- n_tested : The number of genes in each of the trimmed pathways (given in the object@trimPathwayCollection$ object).
- terms : The pathway description, as given in the object@trimPathwayCollection$TERMS object.
- rawp : The unadjusted $p$-values of each pathway.
- ... : Additional columns of adjusted $p$-values as specified through the adjustment argument.

The data frame will be sorted in ascending order by the method specified first in the adjustment argument. If adjustpValues = FALSE, then the data frame will be sorted by the raw $p$-values. If you have the suggested tidyverse package suite loaded, then this data frame will print as a tibble. Otherwise, it will print as a data frame.

## See Also

CreateOmics; ExtractAESPCs; PermTestSurv; PermTestReg; PermTestCateg; TabulatepValues; clusterApply

## Examples

```
## Not run:
  ###  Load the Example Data  ###
  data("colonSurv_df")
  data("colon_pathwayCollection")

  ###  Create an OmicsSurv Object  ###
  colon_Omics <- CreateOmics(
    assayData_df = colonSurv_df[, -(2:3)],
    pathwayCollection_ls = colon_pathwayCollection,
    response = colonSurv_df[, 1:3],
    respType = "surv"
  )

  ###  Calculate Pathway p-Values  ###
  colonSurv_pVals_df <- AESPCA_pVals(
    object = colon_Omics,
    numReps = 5000,
    parallel = TRUE,
    numCores = 16,
    adjustpValues = TRUE,
    adjustment = c("Hoch", "SidakSD")
  )

## End(Not run)
```

---

# colonSurv_df *Colon Cancer -Omics Data*

---

## Description

Subset of a colon cancer survival data set, with subject response and assay values.

## Usage

```
colonSurv_df
```

## Format

A subset of a data frame containing 656 of 2022 genes measured on 250 subjects. The first two columns are the Overall Survival time (`OS_time`) and death indicator (`OS_event`).

## Source

Xi Steven Chen

## colon_pathwayCollection *Gene Pathway Subset*

### Description

An example Canonical Pathways Gene Subset from the Broad Institute: File: `c2.cp.v6.0.symbols.gmt`.

### Usage

```
colon_pathwayCollection
```

### Format

A `pathwayCollection` list of two elements:

- `pathways` : A list of 15 character vectors. Each vector contains the names of the individual genes within that pathway as a vector of character strings.
- `TERMS` : A character vector of length 15 containing the names of the gene pathways.

### Details

This is a subset of 15 pathways from the Broad Institute pathways list. This subset contains seven pathways which are related to the response information in the `colonSurv_df` data file.

### Source

http://software.broadinstitute.org/gsea/msigdb/collections.jsp

## CreateOmics *Generation Wrapper function for* `-Omics*-class objects`

### Description

This function calls the `CreateOmicsPath`, `CreateOmicsSurv`, `CreateOmicsReg`, and `CreateOmicsCateg` functions to create valid objects of the classes `OmicsPathway`, `OmicsSurv`, `OmicsReg`, or `OmicsCateg`, respectively.

### Usage

```
CreateOmics(assayData_df, pathwayCollection_ls, response = NULL,
  respType = c("none", "survival", "regression", "categorical"),
  centerScale = c(TRUE, TRUE), minPathSize = 3, ...)
```

## Arguments

| | |
|---|---|
| assayData_df | An $N \times p$ data frame with named columns. |
| pathwayCollection_ls | A pathwayCollection list of known gene pathways with two or three elements:<br>• pathways : A named list of character vectors. Each vector contains the names of the individual genes within that pathway as a vector of character strings. The names contained in these vectors must have non-empty overlap with the *column names* of the assayData_df data frame. The names of the pathways (the list elements themselves) should be the a shorthand representation of the full pathway name.<br>• TERMS: A character vector the same length as the pathways list with the proper names of the pathways.<br>• description : An optional character vector the same length as the pathways list with additional information about the pathways.<br>If your gene pathways list is stored in a .gmt file, use the read_gmt function to import your pathways list as a pathwayCollection list object. |
| response | An optional response object. See "Details" for more information. Defaults to NULL. |
| respType | What type of response has been supplied. Options are "none", "survival", "regression", and "categorical". Defaults to "none" to match the default response = NULL value. |
| centerScale | Should the values in assayData_df be centered and scaled? Defaults to TRUE for centering and scaling, respectively. See scale for more information. |
| minPathSize | What is the smallest number of genes allowed in each pathway? Defaults to 3. |
| ... | Dots for additional arguments passed to the internal CheckAssay function. |

## Details

This function is a wrapper around the four CreateOmics* functions. The values supplied to the response function argument can be in a list, data frame, matrix, vector, Surv object, or any class which extends these. Because this function makes "best guess" type conversions based on the respType argument, this argument is mandatory if response is non-NULL. Further, it is the responsibility of the user to ensure that the coerced response contained in the resulting Omics object accurately reflects the supplied response.

For respType = "survival", response is assumed to be ordered by event time, then event indicator. For example, if the response is a data frame or matrix, this function assumes that the first column is the time and the second column the death indicator. If the response is a list, then this function assumes that the first entry in the list is the event time and the second entry the death indicator. The death indicator must be a logical or binary (0-1) vector, where 1 or TRUE represents a death and 0 or FALSE represents right-censoring.

Some of the pathways in the supplied pathways list will be removed, or "trimmed", during object creation. For the pathway-testing methods, these trimmed pathways will have $p$-values given as NA. For an explanation of pathway trimming, see the documentation for the IntersectOmicsPwyCollct function.

## Value

A valid object of class OmicsPathway, OmicsSurv, OmicsReg, or OmicsCateg.

## See Also

OmicsPathway, CreateOmicsPath, OmicsSurv, CreateOmicsSurv, OmicsCateg, CreateOmicsCateg OmicsReg, CreateOmicsReg, CheckAssay, CheckPwyColl, and IntersectOmicsPwyCollct

## Examples

```
## Not run:
  ###  Load the Example Data  ###
  data("colonSurv_df")
  data("colon_pathwayCollection")

  ###  Create an OmicsPathway Object  ###
  colon_OmicsPath <- CreateOmics(
    assayData_df = colonSurv_df[, -(2:3)],
    pathwayCollection_ls = colon_pathwayCollection
  )

  ###  Create an OmicsSurv Object  ###
  colon_OmicsSurv <- CreateOmics(
    assayData_df = colonSurv_df[, -(2:3)],
    pathwayCollection_ls = colon_pathwayCollection,
    response = colonSurv_df[, 1:3],
    respType = "surv"
  )

  ###  Create an OmicsReg Object  ###
  colon_OmicsReg <- CreateOmics(
    assayData_df = colonSurv_df[, -(2:3)],
    pathwayCollection_ls = colon_pathwayCollection,
    response = colonSurv_df[, 1:2],
    respType = "reg"
  )

  ###  Create an OmicsCateg Object  ###
  colon_OmicsCateg <- CreateOmics(
    assayData_df = colonSurv_df[, -(2:3)],
    pathwayCollection_ls = colon_pathwayCollection,
    response = colonSurv_df[, c(1,3)],
    respType = "cat"
  )

## End(Not run)
```

---

# CreateOmicsPath *Generation functions for* -Omics*-class objects

---

## Description

These functions create valid objects of class OmicsPathway, OmicsSurv, OmicsReg, or OmicsCateg.

## Usage

```
CreateOmicsPath(assayData_df, sampleIDs_char, pathwayCollection_ls)

CreateOmicsSurv(assayData_df, sampleIDs_char, pathwayCollection_ls,
  eventTime_num, eventObserved_lgl)

CreateOmicsReg(assayData_df, sampleIDs_char, pathwayCollection_ls,
  response_num)

CreateOmicsCateg(assayData_df, sampleIDs_char, pathwayCollection_ls,
  response_fact)
```

## Arguments

| | |
|---|---|
| `assayData_df` | An $N \times p$ data frame with named columns. |
| `sampleIDs_char` | A character vector with the N sample names. |
| `pathwayCollection_ls` | A `pathwayCollection` list of known gene pathways with two or three elements:<br>• `pathways` : A named list of character vectors. Each vector contains the names of the individual genes within that pathway as a vector of character strings. The names contained in these vectors must have non-empty overlap with the *column names* of the `assayData_df` data frame. The names of the pathways (the list elements themselves) should be the a shorthand representation of the full pathway name.<br>• `TERMS`: A character vector the same length as the `pathways` list with the proper names of the pathways.<br>• `description` : An optional character vector the same length as the `pathways` list with additional information about the pathways. |
| `eventTime_num` | A `numeric` vector with $N$ observations corresponding to the last observed time of follow up. |
| `eventObserved_lgl` | A `logical` vector with $N$ observations indicating right-censoring. The values will be `FALSE` if the observation was censored (i.e., we did not observe an event). |
| `response_num` | A `numeric` vector of length $N$: the dependent variable in an ordinary regression exercise. |
| `response_fact` | A `factor` vector of length $N$: the dependent variable of a generalized linear regression exercise. |

## Details

Please note that the classes of the parameters are *not* flexible. The -Omics assay data *must* be or extend the class `data.frame`, and the response values (for a survival-, regression-, or categorical-response object) *must* match their expected classes *exactly*. The reason for this is to encourage the end user to pay attention to the quality and format of their input data. Because the functions internal to this package have only been tested on the classes described in the Arguments section, these class checks prevent unexpected errors (or worse, incorrect computational results without an error). These draconian input class restrictions protect the accuracy of your data analysis.

## Value

A valid object of class `OmicsPathway`, `OmicsSurv`, `OmicsReg`, or `OmicsCateg`.

### OmicsPathway

Valid OmicsPathway objects will have no response information, just the mass spectrometry or bioassay ("design") matrix and the pathway list. OmicsPathway objects should be created only when unsupervised pathway extraction is needed (not possible with Supervised PCA). Because of the missing response, no pathway testing can be performed on an OmicsPathway object.

### OmicsSurv

Valid OmicsSurv objects will have two response vectors: a vector of the most recently recorded follow-up times and a logical vector if that time marks an event (TRUE: observed event; FALSE: right- censored observation).

### OmicsReg and OmicsCateg

Valid OmicsReg and OmicsCateg objects with have one response vector of continuous (numeric) or categorial (factor) observations, respectively.

### See Also

OmicsPathway, OmicsSurv, OmicsReg, and OmicsCateg

### Examples

```
# DO NOT CALL THESE FUNCTIONS DIRECTLY. USE CreateOmics() INSTEAD.
```

---

## CreatePathwayCollection

*Manually Create a* pathwayCollection*-class Object.*

---

### Description

Manually create a pathwayCollection list similar to the output of the read_gmt function.

### Usage

```
CreatePathwayCollection(pathways, TERMS, ...)
```

### Arguments

| | |
|---|---|
| pathways | A named list of character vectors. Each vector should contain the names of the individual genes or proteins within that pathway as a vector of character strings. The names contained in these vectors should have non-empty overlap with the feature names of the assay data frame that will be paired with this list in the subsequent analysis. The names of the pathways (the list elements themselves) should be the a shorthand representation of the full pathway name, but this is not required. |
| TERMS | A character vector the same length as the pathways list with the proper names of the pathways. |
| ... | Additional vectors or data components related to the pathways list. These values should be passed as a name-value pair. See "Details" for more information. |

## Details

This function checks the pathwy list and pathway term inputs and then creates a pathwayCollection object from them. Pass additional list elements (such as the description of each pathway) using the form tag = value through the ... argument (as in the list function). Because some functions in the pathwayPCA package add and edit elements of pathwayCollection objects, please do not create pathwayCollection list items named setsize or n_tested.

## Value

A list object with class pathwayCollection.

## See Also

read_gmt

## Examples

```
data("colon_pathwayCollection")

CreatePathwayCollection(
  pathways = colon_pathwayCollection$pathways,
  TERMS = colon_pathwayCollection$TERMS
)
```

---

# getPathPCLs *Extract PCs and Loadings from a* superpcOut- *or* aespcOut-*class Object.*

---

## Description

Given an object of class aespcOut or superpcOut, as returned by the functions AESPCA_pVals or SuperPCA_pVals, respectively, and the name or unique ID of a pathway, return a data frame of the principal components and a data frame of the loading vectors corresponding to that pathway.

## Usage

```
getPathPCLs(pcOut, pathway_char, ...)

## S3 method for class 'superpcOut'
getPathPCLs(pcOut, pathway_char, ...)

## S3 method for class 'aespcOut'
getPathPCLs(pcOut, pathway_char, ...)
```

## Arguments

| | |
|---|---|
| pcOut | An object of classes superpcOut or aespcOut as returned by the SuperPCA_pVals or AESPCA_pVals functions, respectively. |
| pathway_char | A character string of the name or unique identifier of a pathway |
| ... | Dots for additional arguments (currently unused). |

## Details

Match the supplied pathway character string to either the `pathways` or `terms` columns of the `pVals_df` data frame within the `pcOut` object. Then, subset the `loadings_ls` and `PCs_ls` lists for their entries which match the supplied pathway. Finally, return a list of the PCs, loadings, and the pathway ID and name.

## Value

A list of four elements:

- `PCs` : A data frame of the principal components
- `Loadings` : A matrix of the loading vectors with features in the row names
- `pathway` : The unique pathway identifier for the `pcOut` object
- `term` : The name of the pathway

NULL

NULL

## Examples

```
## Not run:

  ###  Load Data  ###
  data("colonSurv_df")
  data("colon_pathwayCollection")

  ###  Create -Omics Container  ###
  colon_Omics <- CreateOmics(
    assayData_df = colonSurv_df[, -(2:3)],
    pathwayCollection_ls = colon_pathwayCollection,
    response = colonSurv_df[, 1:3],
    respType = "survival"
  )

  ###  Calculate Supervised PCA Pathway p-Values  ###
  colon_superpc <- SuperPCA_pVals(
    colon_Omics,
    numPCs = 2,
    parallel = TRUE,
    adjustment = "BH"
  )

  ###  Extract PCs and Loadings  ###
  getPathPCLs(
    colon_superpc,
    "KEGG_PENTOSE_PHOSPHATE_PATHWAY"
  )


## End(Not run)
```

LoadOntoPCs *Calculate Test Data PCs from Training-Data Estimated Loadings*

## Description

Given a list of loading vectors from a training data set, calculate the PCs of the test data set.

## Usage

```
LoadOntoPCs(design_df, loadings_ls)
```

## Arguments

| | |
|---|---|
| design_df | A test data frame with rows as samples and named features as columns |
| loadings_ls | A list of $p \times d$ loading vectors or matrices as returned by either the SuperPCA_pVals, AESPCA_pVals, or ExtractAESPCs functions. These lists of loadings will have feature names as their row names. Such feature names must match a subset of the column names of design_df exactly, as pathway-specific test-data subsetting is performed by column name. |

## Details

This function takes in a list of loadings and a training-centered test data set, applies over the list of loadings, subsets the columns of the test data by the row names of the loading vectors, right-multiplies the test-data subset matrix by the loading vector / matrix, and returns a data frame of the test-data PCs for each loading vector.

## Value

A data frame with the PCs from each pathway concatenated by column. If you have the tidyverse loaded, this object will display as a tibble.

## Examples

```
## Not run:

  ###  Load the Data  ###
  data("colonSurv_df")
  data("colon_pathwayCollection")

  ###  Create -Omics Container  ###
  colon_Omics <- CreateOmics(
    assayData_df = colonSurv_df[, -(2:3)],
    pathwayCollection_ls = colon_pathwayCollection,
    response = colonSurv_df[, 1:3],
    respType = "survival"
  )

  ###  Extract SuperPCs  ###
  colon_superpc <- SuperPCA_pVals(
    colon_Omics,
    parallel = TRUE,
    adjustment = "BH"
```

```
)

###  Project Data onto Pathway First PCs  ###
LoadOntoPCs(
  design_df = colonSurv_df,
  loadings_ls = colon_superpc$loadings_ls
)


## End(Not run)
```

---

`OmicsCateg-class` *An S4 class for categorical responses within an* `OmicsPathway` *object*

---

## Description

This creates the `OmicsCateg` class which extends the `OmicsPathway` master class.

## Slots

`assayData_df` An $N \times p$ data frame with named columns.

`pathwayCollection` A list of known gene pathways with three or four elements:

- `pathways` : A named list of character vectors. Each vector contains the names of the individual genes within that pathway as a vector of character strings. The names contained in these vectors must have non-empty overlap with the *column names* of the `assayData_df` data frame. The names of the pathways (the list elements themselves) should be the a shorthand representation of the full pathway name.
- `TERMS` : A character vector the same length as the `pathways` list with the proper names of the pathways.
- `description` : An optional character vector the same length as the `pathways` list with additional information about the pathways.
- `setsize` : A named integer vector the same length as the `pathways` list with the number of genes in each pathway. This list item is calculated during the creation step of a `CreateOmics` function call.

`response` A `factor` vector of length $N$: the dependent variable of a generalized linear regression exercise. Currently, we support binary factors only. We expect to extend support to n-ary responses in the next package version.

## See Also

`OmicsPathway`, `CreateOmics`

## OmicsPathway-class *An S4 class for mass spectrometry or bio-assay data and gene pathway lists*

### Description

An S4 class for mass spectrometry or bio-assay data and gene pathway lists

### Slots

`assayData_df` An $N \times p$ data frame with named columns.

`sampleIDs_char` A character vector with the N sample names.

`pathwayCollection` A list of known gene pathways with three or four elements:

- `pathways` : A named list of character vectors. Each vector contains the names of the individual genes within that pathway as a vector of character strings. The names contained in these vectors must have non-empty overlap with the *column names* of the `assayData_df` data frame. The names of the pathways (the list elements themselves) should be the a shorthand representation of the full pathway name.
- `TERMS` : A character vector the same length as the `pathways` list with the proper names of the pathways.
- `description` : An optional character vector the same length as the `pathways` list with additional information about the pathways.
- `setsize` : A named integer vector the same length as the `pathways` list with the number of genes in each pathway. This list item is calculated during the creation step of a `CreateOmics` function call.

`trimPathwayCollection` A subset of the list stored in the `pathwayCollection` slot. This list will have pathways that only contain genes that are present in the assay data frame.

### See Also

`CreateOmics`

## OmicsReg-class *An S4 class for continuous responses within an* `OmicsPathway` *object*

### Description

This creates the `OmicsReg` class which extends the `OmicsPathway` master class.

### Slots

`assayData_df` An $N \times p$ data frame with named columns.

`pathwayCollection` A list of known gene pathways with three or four elements:

- `pathways` : A named list of character vectors. Each vector contains the names of the individual genes within that pathway as a vector of character strings. The names contained in these vectors must have non-empty overlap with the *column names* of the `assayData_df` data frame. The names of the pathways (the list elements themselves) should be the a shorthand representation of the full pathway name.

- TERMS : A character vector the same length as the pathways list with the proper names of the pathways.
- description : An optional character vector the same length as the pathways list with additional information about the pathways.
- setsize : A named integer vector the same length as the pathways list with the number of genes in each pathway. This list item is calculated during the creation step of a CreateOmics function call.

response A numeric vector of length $N$: the dependent variable in a regression exercise.

## See Also

OmicsPathway, CreateOmics

---

# OmicsSurv-class *An S4 class for survival responses within an* OmicsPathway *object*

---

## Description

This creates the OmicsSurv class which extends the OmicsPathway master class.

## Slots

assayData_df An $N \times p$ data frame with named columns.

pathwayCollection A list of known gene pathways with three or four elements:

- pathways : A named list of character vectors. Each vector contains the names of the individual genes within that pathway as a vector of character strings. The names contained in these vectors must have non-empty overlap with the *column names* of the assayData_df data frame. The names of the pathways (the list elements themselves) should be the a shorthand representation of the full pathway name.
- TERMS : A character vector the same length as the pathways list with the proper names of the pathways.
- description : An optional character vector the same length as the pathways list with additional information about the pathways.
- setsize : A named integer vector the same length as the pathways list with the number of genes in each pathway. This list item is calculated during the creation step of a CreateOmics function call.

eventTime A numeric vector with $N$ observations corresponding to the last observed time of follow up.

eventObserved A logical vector with $N$ observations indicating right-censoring. The values will be FALSE if the observation was censored (i.e., we did not observe an event).

## See Also

OmicsPathway, CreateOmics

## pathwayPCA *Extract and Test the Significance of Pathway-Specific Principal Components*

### Description

To introduce this package, please see our "Integrative Pathway Analysis" vignette: https://gabrielodom.github.io/pathwayPCA/articles//Introduction_to_pathwayPCA.html.

The pathwayPCA package has three main components:

- Import and Tidy Data: https://gabrielodom.github.io/pathwayPCA/articles/Supplement2-Importing_Data.html
- Create Omics Data Objects https://gabrielodom.github.io/pathwayPCA/articles/Supplement3-Create_Omics_Objects.html
- Test Pathway Significance https://gabrielodom.github.io/pathwayPCA/articles/Supplement4-Methods_Walkthrough.html
- Analyze and Visualize Results https://gabrielodom.github.io/pathwayPCA/articles/Supplement5-Analyse_Results.html

For an overview of these four topics in context, please see our Quickstart Guide: https://gabrielodom.github.io/pathwayPCA/articles/Supplement1-Quickstart_Guide.html

## read_gmt *Read a* .gmt *file in as a* pathwayCollection *object*

### Description

Read a pathways list file in Gene Matrix Transposed (.gmt) format, with special performance consideration for large files. Present this object as a pathwayCollection object.

### Usage

```
read_gmt(file, description = FALSE, delim = "\t")
```

### Arguments

| | |
|---|---|
| file | A path to a file or a connection. This file must be a .gmt file, otherwise input will likely be nonsense. See the "Details" section for more information. |
| description | Should the "description" field (the second field in the .gmt file on each line) be included in the output? Defaults to FALSE. |
| delim | The .gmt delimiter. As proper .gmt files are tab delimited, this defaults to "\t". |

### Details

This function uses R's readChar function to improve character input performance over readLines (and far improve input performance over scan).

See the Broad Institute's "Data Formats" page for a description of the Gene Matrix Transposed file format: https://software.broadinstitute.org/cancer/software/gsea/wiki/index.php/Data_formats#GMT:_Gene_Matrix_Transposed_file_format_.28.2A.gmt.29

## Value

A pathwayCollection list of pathways. This list has three elements:

- pathways : A named list of character vectors. Each vector contains the names of the individual genes within that pathway as a vector of character strings.
- TERMS : A character vector the same length as the pathways list with the proper names of the pathways.
- description : (OPTIONAL) A character vector the same length as the pathways list with a note on that pathway (for the .gmt file included with this package, this field contains hyperlinks to the MSigDB description card for that pathway). This field is included when description = TRUE.

## See Also

print.pathwayCollection; write_gmt

## Examples

```
## Not run:
  # If you have installed the package:
  data_path <- system.file("extdata", "c2.cp.v6.0.symbols.gmt",
                           package = "pathwayPCA", mustWork = TRUE)
  geneset_ls <- read_gmt(data_path)

  # If you are using the development version from GitHub:
  geneset_ls <- read_gmt("inst/extdata/c2.cp.v6.0.symbols.gmt")

## End(Not run)
```

---

# SubsetOmicsPath *Access and Edit Assay or* pathwayCollection *Values in* Omics* *Objects*

---

## Description

"Get" or "Set" the values of the assayData_df, sampleIDs_char, or pathwayCollection slots of an object of class OmicsPathway or a class that extends this class (OmicsSurv, OmicsReg, or OmicsCateg).

## Usage

```
getAssay(object, ...)

getAssay(object) <- value

getSampleIDs(object, ...)

getSampleIDs(object) <- value

getPathwayCollection(object, ...)
```

```
getPathwayCollection(object) <- value

getTrimPathwayCollection(object, ...)

## S4 method for signature 'OmicsPathway'
getAssay(object, ...)

## S4 replacement method for signature 'OmicsPathway'
getAssay(object) <- value

## S4 method for signature 'OmicsPathway'
getSampleIDs(object, ...)

## S4 replacement method for signature 'OmicsPathway'
getSampleIDs(object) <- value

## S4 method for signature 'OmicsPathway'
getPathwayCollection(object, ...)

## S4 replacement method for signature 'OmicsPathway'
getPathwayCollection(object) <- value

## S4 method for signature 'OmicsPathway'
getTrimPathwayCollection(object, ...)
```

## Arguments

| | |
|---|---|
| object | An object of or extending OmicsPathway-class: that class, OmicsSurv-class, OmicsReg-class, or OmicsCateg-class. |
| ... | Dots for additional internal arguments (currently unused). |
| value | The replacement object to be assigned to the specified slot. |

## Details

These functions can be useful to set or extract the assay data or pathways list from an Omics*-class object. However, we recommend that users simply create a new, valid Omics* object instead of modifying an existing one. The validity of edited objects is checked with the ValidOmicsSurv, ValidOmicsCateg, or ValidOmicsReg functions.

Further, because the pathwayPCA methods require a cleaned (trimmed) pathway collection, the trimPathwayCollection slot is read-only. Users may only edit this slot by updating the pathway collection provided to the pathwayCollection slot. Despite this functionality, we **strongly** recommend that users create a new object with the updated pathway collection, rather than attempting to overwrite the slots within an existing object. See IntersectOmicsPwyCollct for details on trimmed pathway collection.

## Value

The "get" functions return the objects in the slots specified: getAssay returns the assayData_df data frame object, getSampleIDs returns the sampleIDs_char character vector, getPathwayCollection returns the pathwayCollection list object, and getTrimPathwayCollection returns the trimPathwayCollection.

These functions can extract these values from any valid OmicsPathway, OmicsSurv, OmicsReg, or OmicsCateg object.

The "set" functions enable the user to edit or replace objects in the assayData_df, sampleIDs_char, or pathwayCollection slots for any OmicsPathway, OmicsSurv, OmicsReg, or OmicsCateg objects, provided that the new values do not violate the validity checks of their respective objects. Because the slot for trimPathwayCollection is filled upon object creation, and to ensure that this pathway collection is "clean", there is no "set" function for the trimmed pathway collection slot. Instead, users can update the pathway collection, and the trimmed pathway collection will be updated automatically. See "Details" for more information on the "set" functions.

## See Also

CreateOmics

## Examples

```
## Not run:
  data("colonSurv_df")
  data("colon_pathwayCollection")

  colon_Omics <- CreateOmics(
    assayData_df = colonSurv_df[, -(2:3)],
    pathwayCollection_ls = colon_pathwayCollection,
    response = colonSurv_df[, 1:3],
    respType = "survival"
  )

  getAssay(colon_OmicsSurv)
  getPathwayCollection(colon_OmicsSurv)

  getAssay(colon_OmicsSurv) <- newAssay_df
  getPathwayCollection(colon_OmicsSurv) <- new_pathwayCollection

## End(Not run)
```

---

# SubsetOmicsResponse *Access and Edit Response of an* OmicsReg *or* OmicsReg *Object*

---

## Description

"Get" or "Set" the values of the response_num or response_fact slots of an object of class OmicsReg or OmicsReg, respectively.

## Usage

```
getResponse(object, ...)

getResponse(object) <- value

## S4 method for signature 'OmicsPathway'
getResponse(object, ...)
```

```
## S4 replacement method for signature 'OmicsPathway'
getResponse(object) <- value
```

## Arguments

| | |
|---|---|
| object | An object of class OmicsReg-class or OmicsCateg-class. |
| ... | Dots for additional internal arguments (currently unused). |
| value | The replacement object to be assigned to the response slot. |

## Details

These functions can be useful to set or extract the response vector from an object of class OmicsReg or OmicsReg. However, we recommend that users simply create a new, valid object instead of modifying an existing one. The validity of edited objects is checked with their respective ValidOmicsCateg or ValidOmicsReg function. Because both classes have a response slot, we set this method for the parent class, OmicsPathway-class.

## Value

The "get" functions return the objects in the slots specified: getResponse returns the response_num vector from objects of class OmicsReg and the response_fact vector from objects of class OmicsCateg. These functions can extract these values from any valid object of those classes.

The "set" functions enable the user to edit or replace the object in the response_num slot for any OmicsReg object or response_fact slot for any OmicsCateg object, provided that the new values do not violate the validity check of such an object. See "Details" for more information.

## See Also

CreateOmics

## Examples

```
## Not run:
  data("colonSurv_df")
  data("colon_pathwayCollection")

  colon_Omics <- CreateOmics(
    assayData_df = colonSurv_df[, -(2:3)],
    pathwayCollection_ls = colon_pathwayCollection,
    response = colonSurv_df[, c(1, 3)],
    respType = "reg"
  )

  getResponse(colon_Omics)
  getResponse(colon_Omics) <- newResponse_num

## End(Not run)
```

SubsetOmicsSurv *Access and Edit Event Time or Indicator in an* OmicsSurv *Object*

## Description

"Get" or "Set" the values of the eventTime_num or eventObserved_lgl slots of an object of class OmicsSurv.

## Usage

```
getEventTime(object, ...)

getEventTime(object) <- value

getEvent(object, ...)

getEvent(object) <- value

## S4 method for signature 'OmicsSurv'
getEventTime(object, ...)

## S4 replacement method for signature 'OmicsSurv'
getEventTime(object) <- value

## S4 method for signature 'OmicsSurv'
getEvent(object, ...)

## S4 replacement method for signature 'OmicsSurv'
getEvent(object) <- value
```

## Arguments

| | |
|---|---|
| object | An object of class OmicsSurv-class. |
| ... | Dots for additional internal arguments (currently unused). |
| value | The replacement object to be assigned to the specified slot. |

## Details

These functions can be useful to set or extract the event time or death indicator from an OmicsSurv object. However, we recommend that users simply create a new, valid OmicsSurv object instead of modifying an existing one. The validity of edited objects is checked with the ValidOmicsSurv function.

## Value

The "get" functions return the objects in the slots specified: getEventTime returns the eventTime_num vector object and getEvent returns the eventObserved_lgl vector object. These functions can extract these values from any valid OmicsSurv object.

The "set" functions enable the user to edit or replace objects in the eventTime_num or eventObserved_lgl slots for any OmicsSurv object, provided that the new values do not violate the validity check of an OmicsSurv object. See "Details" for more information.

## See Also

CreateOmics

## Examples

```
## Not run:
  data("colonSurv_df")
  data("colon_pathwayCollection")

  colon_Omics <- CreateOmics(
    assayData_df = colonSurv_df[, -(2:3)],
    pathwayCollection_ls = colon_pathwayCollection,
    response = colonSurv_df[, 1:3],
    respType = "survival"
  )

  getEventTime(colon_Omics)
  getEvent(colon_Omics)

  getEventTime(colon_Omics) <- newTime_num
  getEvent(colon_Omics) <- newEvent_lgl

## End(Not run)
```

---

# SubsetPathwayCollection

*Subset a* pathwayCollection*-class Object.*

---

## Description

The subset method for pathways lists as returned by the read_gmt function.

## Usage

```
## S3 method for class 'pathwayCollection'
x[[name_char]]
```

## Arguments

| | |
|---|---|
| x | An object of class pathwayCollection. |
| name_char | The name of a pathway in the collection or its unique ID. |

## Details

This function finds the index matching the name_char argument to the TERMS field of the pathwayCollection-class Object, then subsets the pathways list, TERMS vector, description vector, and setsize vector by this index. If you subset a trimmed pathwayCollection object, and the function errors with "Pathway not found.", then the pathway specified has been trimmed from the pathway collection.

Also, this function does not allow for users to overwrite any portion of a pathway collection. These objects should rarely, if ever, be changed. If you absolutely must change the components of a pathwayCollection object, then create a new one with the codeCreatePathwayCollection function.

## Value

A list of the pathway name (`Term`), unique ID (`pathID`), contents (`IDs`), description (`description`), and number of features (`Size`).

## Examples

```
data("colon_pathwayCollection")
colon_pathwayCollection[["KEGG_RETINOL_METABOLISM"]]
```

---

# SubsetPathwayData *Subset Pathway-Specific Data*

---

## Description

Given an `Omics` object and the name of a pathway, return the -omes in the assay and the response as a (tibble) data frame.

## Usage

```
SubsetPathwayData(object, pathName, ...)

## S4 method for signature 'OmicsPathway'
SubsetPathwayData(object, pathName, ...)
```

## Arguments

| | |
|---|---|
| `object` | An object of class `OmicsPathway`, or an object extending this class. |
| `pathName` | The name of a pathway contained in the pathway collection in the object. |
| `...` | Dots for additional internal arguments (currently unused). |

## Details

This function subsets the assay by the matching gene symbols or IDs in the specified pathway.

## Value

A data frame of the columns of the assay in the `Omics` object which are listed in the specified pathway, with a leading column for sample IDs. If the `Omics` object has response information, these are also included as the first column(s) of the data frame, after the sample IDs. If you have the suggested `tidyverse` package suite loaded, then this data frame will print as a `tibble`. Otherwise, it will print as a data frame.

## Examples

```
## Not run:
  data("colonSurv_df")
  data("colon_pathwayCollection")

  colon_Omics <- CreateOmics(
    assayData_df = colonSurv_df[, -(2:3)],
    pathwayCollection_ls = colon_pathwayCollection,
```

```
  response = colonSurv_df[, 1:3],
  respType = "survival"
)

SubsetPathwayData(
  colon_Omics,
  "KEGG_RETINOL_METABOLISM"
)


## End(Not run)
```

---

SuperPCA_pVals *Test pathways with Supervised PCA*

---

## Description

Given a supervised OmicsPath object (one of OmicsSurv, OmicsReg, or OmicsCateg), extract the first $k$ principal components (PCs) from each pathway-subset of the -Omics assay design matrix, test their association with the response matrix, and return a data frame of the adjusted $p$-values for each pathway.

## Usage

```
SuperPCA_pVals(object, n.threshold = 20, numPCs = 1,
  parallel = FALSE, numCores = NULL, adjustpValues = TRUE,
  adjustment = c("Bonferroni", "Holm", "Hochberg", "SidakSS", "SidakSD",
  "BH", "BY", "ABH", "TSBH"), ...)

## S4 method for signature 'OmicsPathway'
SuperPCA_pVals(object, n.threshold = 20,
  numPCs = 1, parallel = FALSE, numCores = NULL,
  adjustpValues = TRUE, adjustment = c("Bonferroni", "Holm",
  "Hochberg", "SidakSS", "SidakSD", "BH", "BY", "ABH", "TSBH"), ...)
```

## Arguments

| | |
|---|---|
| object | An object of superclass OmicsPathway with a response matrix or vector. |
| n.threshold | The number of bins into which to split the feature scores in the fit object returned internally by the superpc.train function to the pathway_tScores and pathway_tControl functions. Defaults to 20. Smaller values may result in less accurate pathway $p$-values while larger values increase computation time. |
| numPCs | The number of PCs to extract from each pathway. Defaults to 1. |
| parallel | Should the computation be completed in parallel? Defaults to FALSE. |
| numCores | If parallel = TRUE, how many cores should be used for computation? Internally defaults to the number of available cores minus 1. |
| adjustpValues | Should you adjust the $p$-values for multiple comparisons? Defaults to TRUE. |

| | |
|---|---|
| adjustment | Character vector of procedures. The returned data frame will be sorted in ascending order by the first procedure in this vector, with ties broken by the unadjusted $p$-value. If only one procedure is selected, then it is necessarily the first procedure. See the documentation for the ControlFDR function for the adjustment procedure definitions and citations. |
| ... | Dots for additional internal arguments. |

## Details

This is a wrapper function for the pathway_tScores, pathway_tControl, OptimGumbelMixParams, GumbelMixpValues, and TabulatepValues functions.

Please see our Quickstart Guide for this package: https://gabrielodom.github.io/pathwayPCA/articles/Supplement1-Quickstart_Guide.html

## Value

A data frame with columns:

- pathways : The names of the pathways in the Omics* object (given in object@trimPathwayCollection$pathways
- setsize : The number of genes in each of the original pathways (given in the object@trimPathwayCollection$se object).
- terms : The pathway description, as given in the object@trimPathwayCollection$TERMS object.
- rawp : The unadjusted $p$-values of each pathway.
- ... : Additional columns as specified through the adjustment argument.

The data frame will be sorted in ascending order by the method specified first in the adjustment argument. If adjustpValues = FALSE, then the data frame will be sorted by the raw $p$-values. If you have the suggested tidyverse package suite loaded, then this data frame will print as a tibble. Otherwise, it will print as a data frame.

## See Also

CreateOmics; TabulatepValues; pathway_tScores; pathway_tControl; OptimGumbelMixParams; GumbelMixpValues; clusterApply

## Examples

```
## Not run:
  ###  Load the Example Data  ###
  data("colonSurv_df")
  data("colon_pathwayCollection")

  ###  Create an OmicsSurv Object  ###
  colon_OmicsSurv <- CreateOmics(
    assayData_df = colonSurv_df[, -(2:3)],
    pathwayCollection_ls = colon_pathwayCollection,
    response = colonSurv_df[, 1:3],
    respType = "surv"
  )

  ###  Calculate Pathway p-Values  ###
  colonSurv_pVals_df <- SuperPCA_pVals(
```

```
    object = colon_OmicsSurv,
    parallel = TRUE,
    numCores = 16,
    adjustpValues = TRUE,
    adjustment = c("Hoch", "SidakSD")
  )

## End(Not run)
```

---

`TransposeAssay` *Transpose an Assay (Data Frame)*

---

## Description

Transpose an object of class `data.frame` that contains assay measurements while preserving row (feature) and column (sample) names.

## Usage

```
TransposeAssay(assay_df, omeNames = c("firstCol", "rowNames"),
  stringsAsFactors = FALSE)
```

## Arguments

| | |
|---|---|
| `assay_df` | A data frame with numeric values to transpose |
| `omeNames` | Are the data feature names in the first column or in the row names of `df`? Defaults to the first column. If the feature names are in the row names, this function assumes that these names are accesible by the `rownames` function called on `df`. |
| `stringsAsFactors` | Should columns containing string information be coerced to factors? Defaults to `FALSE`. |

## Details

This function is designed to transpose "tall" assay data frames (where genes or proteins are the rows and patient or tumour samples are the columns). This function also transposes the row (feature) names to column names and the column (sample) names to row names. Notice that all rows and columns (other than the feature name column, as applicable) are numeric.

Recall that data frames require that all elements of a single column to have the same `class`. Therefore, sample IDs of a "tall" data frame **must** be stored as the column names rather than in the first row.

## Value

The transposition of `df`, with row and column names preserved and reversed.

## Examples

```
x_mat <- matrix(rnorm(5000), ncol = 20, nrow = 250)
rownames(x_mat) <- paste0("gene_", 1:250)
colnames(x_mat) <- paste0("sample_", 1:20)
x_df <- as.data.frame(x_mat, row.names = rownames(x_mat))

TransposeAssay(x_df, omeNames = "rowNames")
```

---

# wikipwsHS_Entrez_pathwayCollection

*Wikipathways Homosapiens EntrezIDs*

---

## Description

A pathwayCollection object containing the homosapiens pathways list from Wikipathways (https://www.wikipathways.org/).

## Usage

```
wikipwsHS_Entrez_pathwayCollection
```

## Format

A pathwayCollection list of three elements:

- pathways : A named list of 443 character vectors. Each vector contains the Entrez Gene IDs of the individual genes within that pathway as a vector of character strings. The names are the shorthand pathway names.
- TERMS : A character vector of length 443 containing the shorthand names of the gene pathways.
- description : A character vector of length 443 containing the full names of the gene pathways.

## Details

This pathwayCollection was sent to us from Dr. Alexander Pico at the Gladstone Institute (https://gladstone.org/our-science/people/alexander-pico).

## Source

Dr. Alexander Pico, Wikipathways

write_gmt *Write a* pathwayCollection *Object to a* .gmt *File*

## Description

Write a pathwayCollection object as a pathways list file in Gene Matrix Transposed (.gmt) format.

## Usage

```
write_gmt(pathwayCollection, file)
```

## Arguments

pathwayCollection
: A pathwayCollection list of pathways. This list contains the following two or three elements:
  - pathways : A named list of character vectors. Each vector contains the names of the individual genes within that pathway as a vector of character strings. Genes can be represented by HGNC gene symbols, Entrez IDs, Ensembl IDs, GO terms, etc.
  - TERMS : A character vector the same length as the pathways list with the proper names of the pathways.
  - description : An optional character vector the same length as the pathways list with a note on that pathway (such as a url to the description of the pathway). If this element of the pathwayCollection is NULL, then the file will be written with "" (the empty character string) as its second field in each line.

file
: Either a character string naming a file or a connection open for writing. File names should end in .gmt for clarity

## Details

See the Broad Institute's "Data Formats" page for a description of the Gene Matrix Transposed file format: https://software.broadinstitute.org/cancer/software/gsea/wiki/index.php/Data_formats#GMT:_Gene_Matrix_Transposed_file_format_.28.2A.gmt.29

## See Also

print.pathwayCollection; read_gmt

## Examples

```
## Not run:
  # Toy pathway set
  toy_pathwayCollection <- list(
    pathways = list(
      c("C1orf27", "NR5A1", "BLOC1S4", "C4orf50"),
      c("TARS2", "DUSP5", "GPR88"),
      c("TRX-CAT3-1", "LINC01333", "LINC01499", "LINC01046", "LINC01149")
    ),
```

```
  TERMS = c("C-or-f_paths", "randomPath2", "randomLINCs"),
  description = c("these are", "totally made up", "pathways")
)
class(toy_pathwayCollection) <- c("pathwayCollection", "list")
print(toy_pathwayCollection)

write_gmt(toy_pathwayCollection, file = "example_pathway.gmt")

## End(Not run)
```

# Index