

Package ‘BEAMR’

July 21, 2025

Title Bootstrap Evaluation of Association Matrices

Version 1.1.0

Description A bootstrap-based approach to integrate multiple forms of high dimensional genomic data with multiple clinical endpoints. This method is used to find clinically meaningful groups of genomic features, such as genes or pathways. A manuscript describing this method is in preparation.

License GPL (>= 3)

Encoding UTF-8

RoxygenNote 7.3.2

Imports dplyr, ggmosaic, ggplot2, ggpubr, logistf, magrittr, MASS, purrr, rlist, stats, stringr, survival, survminer

Suggests rmarkdown

Depends R (>= 2.10)

LazyData true

URL <https://annaseffernick.github.io/BEAMR/>,
<https://github.com/annaSeffernick/BEAMR>

BugReports <https://github.com/annaSeffernick/BEAMR/issues>

NeedsCompilation no

Author Anna Eames Seffernick [aut, cre, cph] (ORCID:
<<https://orcid.org/0000-0003-0848-4604>>),
Stanley Pounds [aut],
Xueyuan Cao [aut]

Maintainer Anna Eames Seffernick <anna.seffernick@stjude.org>

Repository CRAN

Date/Publication 2024-07-27 16:00:06 UTC

Contents

beam_dat	2
beam_dat_sm	3
beam_specs	4
beam_specs_sm	4
beam_stats	5
beam_stats_sm	5
check_beam_specs	6
check_list_class	6
clean_Bmtx	7
clinf	7
compute_beam_stats	8
compute_feature_pvalues	9
compute_set_pvalues	9
extend_set_data	10
extract_beam_stats	11
find_id_clm	11
gen_beam_plot_list	12
get_id_index	13
omicann	14
omicdat	14
plot_beam_boot	15
plot_beam_clin	16
plot_feat_clin	17
prep_beam_data	18
prep_beam_plot	19
prep_beam_specs	20
print.beam.data	21
print.beam.stats	22
setdat	22
specs	23
subset_beam_result	23

Index	26
--------------	-----------

beam_dat

Pediatric T-ALL Clinical Data from COG trial AALL0434

Description

The beam.data object used in example beam analyses

Usage

beam_dat

Format

beam_dat:

A beam.data object, which is a list with the following elements:

main.data A data.frame with clinical/endpoint data.

mtx.data A list of the omics data matrices.

mtx.anns A list of omic annotation data.frames.

anns.mtch A data.frame with information to link mtx.data and mtx.anns.

set.data A data.frame with set.id, mtx.id, and row.id to link omic features to sets.

set.anns Optional data.frame with set annotation data.

boot.index A matrix with bootstrap indices.

Source

NA

beam_dat_sm

Pediatric T-ALL Clinical Data from COG trial AALL0434

Description

The smaller beam.data object used in the example for compute_beam_stats function

Usage

beam_dat_sm

Format

beam_dat_sm:

A beam.data object, which is a list with the following elements:

main.data A data.frame with clinical/endpoint data.

mtx.data A list of the omics data matrices.

mtx.anns A list of omic annotation data.frames.

anns.mtch A data.frame with information to link mtx.data and mtx.anns.

set.data A data.frame with set.id, mtx.id, and row.id to link omic features to sets.

set.anns Optional data.frame with set annotation data.

boot.index A matrix with bootstrap indices.

Source

NA

beam_specs	<i>Pediatric T-ALL BEAM Analysis Specs Data from COG trial AALL0434</i>
------------	---

Description

The beam.specs object used in example beam analyses

Usage

beam_specs

Format

beam_specs:

A data frame with 6 rows and 3 columns:

name Analysis name with omic and endpoint

mtx Name of omics matrix used in the analysis

mdl Regression model

Source

NA

beam_specs_sm	<i>Pediatric T-ALL BEAM Analysis Specs Data from COG trial AALL0434</i>
---------------	---

Description

The small beam.specs object used in example compute_beam_stats function.

Usage

beam_specs_sm

Format

beam_specs_sm:

A data frame with 2 rows and 3 columns:

name Analysis name with omic and endpoint

mtx Name of omics matrix used in the analysis

mdl Regression model

Source

NA

`beam_stats`*Pediatric T-ALL Clinical Data from COG trial AALL0434*

Description

The beam.stats object used in example beam analyses

Usage`beam_stats`**Format**`beam_stats:`

A beam.stats object, which contains the following objects

beam.stats A list of data.frames of association statistics for each omic-endpoint pair.

beam.specs A beam.specs object (data.frame with name, mtx, and mdl.)

beam.data The beam.data object.

Source

NA

`beam_stats_sm`*Pediatric T-ALL Clinical Data from COG trial AALL0434*

Description

The small beam.stats object used in example for compute_beam_stats function.

Usage`beam_stats_sm`**Format**`beam_stats_sm:`

A beam.stats object, which contains the following objects

beam.stats A list of data.frames of association statistics for each omic-endpoint pair.

beam.specs A beam.specs object (data.frame with name, mtx, and mdl.)

beam.data The beam.data object.

Source

NA

check_beam_specs	<i>Check that beam.specs satisfies all necessary conditions</i>
------------------	---

Description

Check that beam.specs satisfies all necessary conditions

Usage

```
check_beam_specs(beam.specs, mtx.names)
```

Arguments

beam.specs	A data.frame with column name, mtx, and mdl
mtx.names	A vector with the names of the data matrices (beam.data\$mtx.data)

Value

A data.frame of beam.specs if all conditions satisfied, otherwise throws an error

Examples

```
data(beam_dat)
data(beam_specs)
test_specs <- check_beam_specs(beam_specs, names(beam_dat$mtx.data))
```

check_list_class	<i>Check that each element of a list is of a required class</i>
------------------	---

Description

Check that each element of a list is of a required class

Usage

```
check_list_class(list.object, required.class)
```

Arguments

list.object	A list used in BEAMR analysis
required.class	Class for list elements, e.g. matrix

Value

Logical TRUE if list is of required class

Examples

```
data(omicdat)
check_list_class(omicdat, "matrix")
```

clean_Bmtx

Clean up bootstrap coefficient matrix

Description

Clean up bootstrap coefficient matrix

Usage

```
clean_Bmtx(B)
```

Arguments

B Matrix of bootstrap coefficients

Value

Matrix of cleaned bootstrap coefficients

Examples

```
data(beam_stats)
B.mtx <- beam_stats$beam_stats[[1]]
B.cln <- clean_Bmtx(B.mtx)
```

clinf

Pediatric T-ALL Clinical Data from COG trial AALL0434

Description

A subset of clinical data from pediatric and young adult t-lineage acute lymphoblastic leukemia patients in the Children's Oncology Group trial AALL0434, published in Liu et al., 2017 Nature Genetics

Usage

```
clinf
```

Format

clinf:

A data frame with 265 rows and 8 columns:

ID Subject ID

MRD29 Minimal residual disease measured at day 29

RNA.clm Key to match to RNA matrix

Lesion.clm Key to match Lesion matrix

Lesion.id Key to match Lesion matrix

RNA.id Key to match RNA matrix

EFS Event-free survival Surv object

OS Overall survival Surv object

Source

<https://www.nature.com/articles/ng.3909>

compute_beam_stats	<i>Compute bootstrap model coefficients for BEAM</i>
--------------------	--

Description

Compute bootstrap model coefficients for BEAM

Usage

```
compute_beam_stats(beam.data, beam.specs, stdize = TRUE)
```

Arguments

beam.data	Result of prep.beam.data
beam.specs	A data.frame of strings with columns name, mtx, mdl (string with R model with mtx.row)
stdize	Logical whether to standardize (center and scale) predictors or not. Default is TRUE.

Value

A beam.stats object, which is a list with beam.stats (the association matrices), the beam.specs, and the beam.data

Examples

```
data(beam_dat_sm)
data(beam_specs_sm)
test.beam.stats <- compute_beam_stats(beam.data=beam_dat_sm,
                                     beam.specs=beam_specs_sm, stdize=TRUE)
```

`compute_feature_pvalues`*Compute feature level p-values from BEAM statistics*

Description

Compute feature level p-values from BEAM statistics

Usage

```
compute_feature_pvalues(beam.stats)
```

Arguments

<code>beam.stats</code>	A beam.stats object, which is a list with beam.stats (the association matrices), the beam.specs, and the beam.data
-------------------------	--

Value

A list of feature level p-values, with each entry a data frame for a different omics/endpoint association, with columns id, gene, beta, p, q

Examples

```
data(beam_stats)
test.featt.pvals <- compute_feature_pvalues(beam.stats=beam_stats)
```

`compute_set_pvalues` *Compute BEAMR p-values for sets*

Description

Compute BEAMR p-values for sets

Usage

```
compute_set_pvalues(  
  beam.stats,  
  peel = FALSE,  
  z = TRUE,  
  alpha = 0.1,  
  mess.freq = 25  
)
```

Arguments

beam.stats	A beam.stats object from compute_beam_stats function
peel	Logical indicating whether to peel in p-value calculation
z	Logical indicating whether to z-scale each vector of one coefficient estimate across bootstraps before analysis
alpha	Maximum depth to peel (reduces computing time); default 0.1.
mess.freq	Message frequency; default 25.

Value

A list with a data.frame of set p-values from BEAMR analysis, a data.frame of summary row p-values, and a data frame of set matching.

Examples

```
data(beam_stats_sm)
test.pvals <- compute_set_pvalues(beam.stats=beam_stats_sm)
```

extend_set_data	<i>Extend set definition data with genes on the same row separated by commas, semicolons, slashes, etc</i>
-----------------	--

Description

Extend set definition data with genes on the same row separated by commas, semicolons, slashes, etc

Usage

```
extend_set_data(set.data, sep)
```

Arguments

set.data	A data frame with set definition data.
sep	Punctuation to split on.

Value

A data frame.

Examples

```
data(setdat)
extend_set_data(setdat, sep=",")
```

extract_beam_stats	<i>Extract beam stats for a specific set</i>
--------------------	--

Description

Extract beam stats for a specific set

Usage

```
extract_beam_stats(beam.stats, set.id)
```

Arguments

beam.stats	A beam.stats object, which is a list with beam.stats (the association matrices), the beam.specs, and the beam.data
set.id	A character of a set id name (an entry in in beam.data\$set.data\$set.id)

Value

A matrix with with estimated associations for each endpoint and each omic feature linked to the set

Examples

```
data(beam_stats)
test.stats <- extract_beam_stats(beam_stats, set.id="ENSG00000099810")
```

find_id_clm	<i>Find the column of mtch.data with the most rows containing an element of ids</i>
-------------	---

Description

Find the column of mtch.data with the most rows containing an element of ids

Usage

```
find_id_clm(mtch.data, ids)
```

Arguments

mtch.data	A data.frame
ids	A vector of row ids to match

Value

A vector of column names with the most matches.

Examples

```
data(omicann)
data(omicdat)
lsn.data <- omicann[[1]]
mtx.rows <- rownames(omicdat[[1]])
test <- find_id_clm(lsn.data,mtx.rows)
```

gen_beam_plot_list	<i>Generate BEAM Plot List</i>
--------------------	--------------------------------

Description

Internal function: generate a list of clinical feature plots.

Usage

```
gen_beam_plot_list(
  beam.result,
  beam.specs,
  beam.feats.pvals,
  number.pairs = 1,
  set.id,
  feat.id = NULL,
  title.size = 10,
  pair.order = "both",
  endpt.order = NULL
)
```

Arguments

beam.result	Result of prep.beam.data
beam.specs	A data.frame of strings with columns name, mtx, mdl, plot
beam.feats.pvals	List of feature-level p-values from compute_feature_pvalues
number.pairs	Numeric; number of features to display in clinical plots, ordered by significance
set.id	A character with set name; must be in beam.result\$beam.data\$set.data\$set.id
feat.id	Default NULL; a character with feature name; must be in beam.result\$beam.data\$set.data\$row.id
title.size	A numeric. Specify the size of individual plot titles. Default is 10.
pair.order	One of c("both", "omic", "endpoint"). Default is "both." Specify how to choose feature-endpoint plots to include. If "both", find the best (based on q, p, effect size) feature-omic pair for each type of omic and each endpoint separately. If "omic", within each omic, find the best feature-endpoint pair and then plot this feature with all endpoints. If "endpoint", need to specify endpt.order as the name of chosen endpoint. Then, within each omic, find the feature with best association with the selected endpoint, and plot this feature for all endpoints.
endpt.order	Default NULL. If pair.order="endpoint", specify character with endpoint name (from beam.specs\$name, after the period).

Value

A list of plots for the specified set and/or feature.

Examples

```
data(beam_stats)
test.featt.pvals <- compute_feature_pvalues(beam.stats=beam_stats)
plot.specs <- prep_beam_plot(beam.data=beam_stats$beam.data,
                             beam.specs=beam_stats$beam.specs)
plot.list <- gen_beam_plot_list(beam.result=beam_stats, beam.specs=plot.specs,
                                beam.featt.pvals=test.featt.pvals,
                                number.pairs=1, set.id="ENSG00000099810",
                                feat.id=NULL, title.size=11,
                                pair.order="omic", endpt.order=NULL)
```

get_id_index	<i>For each row of the data.frame main.data, find the index of the matching element in vector ids</i>
--------------	---

Description

For each row of the data.frame main.data, find the index of the matching element in vector ids

Usage

```
get_id_index(mtch.data, ids, warn = TRUE)
```

Arguments

mtch.data	A data.frame to be linked with the ids
ids	A vector of ids to be linked in mtch.data
warn	A logical value whether to include warnings with results

Value

A data.frame with matching id index

Examples

```
data(clinf)
data(omicdat)
mtx.clms <- colnames(omicdat[[1]])
id_index <- get_id_index(clinf,mtx.clms)
```

omicann

Pediatric T-ALL Omics Annotation Data from COG trial AALL0434

Description

A subset of genomic lesion and RNA expression data from pediatric and young adult t-lineage acute lymphoblastic leukemia patients in the Children's Oncology Group trial AALL0434, published in Liu et al., 2017 Nature Genetics. This is the annotation mapping feature id to gene name given by Ensembl ID.

Usage

```
omicann
```

Format

```
omicann:
```

A list with two data frames of omics annotation.

Lesion A dataframe with 20 rows and 2 columns with lesion ID and Ensembl ID.

RNA A dataframe with 20 rows and 2 columns with feature ID and Ensembl ID.

Source

<https://www.nature.com/articles/ng.3909>

omicdat

Pediatric T-ALL Omics Data from COG trial AALL0434

Description

A subset of genomic lesion and RNA expression data from pediatric and young adult t-lineage acute lymphoblastic leukemia patients in the Children's Oncology Group trial AALL0434, published in Liu et al., 2017 Nature Genetics

Usage

```
omicdat
```

Format

```
omicdat:
```

A list with two dataframes of omic data for each subject

Lesion A dataframe with 20 rows and 265 columns indicating presence of lesion.

RNA A dataframe with 20 rows and 265 columns with expression data.

plot_beam_clin

*Plot BEAM Sets***Description**

plot_beam_clin produces a matrix of feature level clinical plots for a set. Users can specify which omic/endpoint pairs they want to see as well as the number of features from the set. Default is all omic/endpoint pairs and the top feature (smallest feature-level p-value).

Usage

```
plot_beam_clin(
  beam.result,
  beam.specs = NULL,
  beam.set.pvals,
  beam.feats.pvals,
  set.id,
  gene.name = NULL,
  pair.type = NULL,
  number.pairs = 1,
  pair.order = "both",
  endpt.order = NULL,
  n.col = NULL,
  n.row = NULL,
  title.size = 10
)
```

Arguments

beam.result	A beam.stats object from compute_beam_stats
beam.specs	A data.frame. Default NULL, in which case beam.result\$beam.specs is used. Otherwise can input other beam.specs data.frame that must contain name, mtx, mdl, plot columns.
beam.set.pvals	A list containing BEAMR set p-values from compute_set_pvalues.
beam.feats.pvals	A list containing feature-level p-values from compute_feature_pvalues.
set.id	A character specifying the name of a set. Must be in beam.result\$beam.data\$set.data
gene.name	A character specifying a Gene Name/Symbol for the set. Default is NULL
pair.type	A character vector. Default NULL, in which case clinical plots for all omic/endpoint pairs are produced. Otherwise specify pairs from beam.stats\$beam.specs\$name
number.pairs	A numeric. Default 1, in which case only feature with best simple test for each pair is plotted. If >1, show top n simple plots ordered by feature-level p-value
pair.order	One of c("both", "omic", "endpoint"). Default is "both." Specify how to choose feature-endpoint plots to include. If "both", find the best (based on q, p, effect

	size) feature-omic pair for each type of omic and each endpoint separately. If "omic", within each omic, find the best feature-endpoint pair and then plot this feature with all endpoints. If "endpoint", need to specify endpt.order as the name of chosen endpoint. Then, within each omic, find the feature with best association with the selected endpoint, and plot this feature for all endpoints.
endpt.order	Default NULL. If pair.order="endpoint", specify character with endpoint name (from beam.specs\$name, after the period).
n.col	A numeric. Specify the number of columns for the plot layout; default NULL will use the number of omics types.
n.row	A numeric. Specify the number of rows for the plot layout; default NULL will automatically define the number of rows after number of columns specified.
title.size	A numeric. Specify the size of individual plot titles. Default is 10.

Value

A figure (ggarrange object)

Examples

```
data(beam_stats)
test.pvals <- compute_set_pvalues(beam.stats=beam_stats)
test.featt.pvals <- compute_feature_pvalues(beam.stats=beam_stats)
plot.specs <- prep_beam_plot(beam.data=beam_stats$beam.data,
                             beam.specs=beam_stats$beam.specs)
test.plot <- plot_beam_clin(beam.result=beam_stats, beam.specs=plot.specs,
                           beam.set.pvals=test.pvals,
                           beam.featt.pvals=test.featt.pvals,
                           set.id="ENSG00000099810", gene.name="MTAP",
                           pair.type=NULL, number.pairs=1, n.col=4,
                           n.row=NULL, title.size=11,
                           pair.order="omic", endpt.order=NULL)
```

plot_feat_clin

Plot BEAM Feature

Description

plot_feat_clin produces a matrix of feature level clinical plots for a specific feature.

Usage

```
plot_feat_clin(
  feat.id,
  beam.result,
  beam.specs = NULL,
  beam.set.pvals,
  beam.featt.pvals,
```

```

    n.row = NULL,
    n.col = NULL
  )

```

Arguments

<code>feat.id</code>	A character specifying the name of a feature. Must be in <code>beam.result\$beam.data\$set.data</code>
<code>beam.result</code>	A <code>beam.stats</code> object from <code>compute_beam_stats</code>
<code>beam.specs</code>	A <code>data.frame</code> . Default <code>NULL</code> , in which case <code>beam.result\$beam.specs</code> is used. Otherwise can input other <code>beam.specs</code> <code>data.frame</code> that must contain <code>name</code> , <code>mtx</code> , <code>mdl</code> , <code>plot</code> columns.
<code>beam.set.pvals</code>	A list containing BEAMR set p-values from <code>compute_set_pvalues</code> .
<code>beam.featt.pvals</code>	A list containing feature-level p-values from <code>compute_feature_pvalues</code> .
<code>n.row</code>	A numeric. Specify the number of rows for the plot layout; default <code>NULL</code> will automatically define the number of rows after number of columns specified.
<code>n.col</code>	A numeric. Specify the number of columns for the plot layout; default <code>NULL</code> will use the number of omics types.

Value

A figure (ggarrange object)

Examples

```

data(beam_stats)
test.pvals <- compute_set_pvalues(beam.stats=beam_stats)
test.featt.pvals <- compute_feature_pvalues(beam.stats=beam_stats)
plot.specs <- prep_beam_plot(beam.data=beam_stats$beam.data, beam.specs=beam_stats$beam.specs)
test.plot <- plot_feat_clin(beam.result=beam_stats, beam.specs=plot.specs,
                           beam.set.pvals=test.pvals, beam.featt.pvals=test.featt.pvals,
                           feat.id="ENSG00000227443_loss",
                           n.col=2, n.row=NULL)

```

<code>prep_beam_data</code>	<i>Prepare data for BEAM analysis</i>
-----------------------------	---------------------------------------

Description

Prepare data for BEAM analysis

Usage

```
prep_beam_data(
  main.data,
  mtx.data,
  mtx.anns = NULL,
  set.data = NULL,
  set.anns = NULL,
  n.boot = 1000,
  seed = NULL
)
```

Arguments

main.data	A data.frame
mtx.data	A list, each element is a matrix
mtx.anns	A list, each element is a data.frame
set.data	A data.frame with columns set.id, mtx.id, row.id
set.anns	A data frame with set.id and other columns
n.boot	Number of bootstraps
seed	Initial seed for random number generation

Value

A beam.data object, which is a list with main.data, mtx.data, mtx.anns, anns.mtch, set.data, set.anns, and boot.index

Examples

```
data(clinf)
data(omicdat)
data(omicann)
data(setdat)
test.beam.data <- prep_beam_data(main.data=clinf, mtx.data=omicdat,
                                mtx.anns=omicann, set.data=setdat,
                                set.anns=NULL, n.boot=10, seed=123)
```

prep_beam_plot	<i>Prepare for BEAM plotting</i>
----------------	----------------------------------

Description

Add a "plot" column to beam.specs, which includes string of plot commands.

Usage

```
prep_beam_plot(beam.data, beam.specs)
```

Arguments

beam.data	Result of prep.beam.data
beam.specs	A data.frame of strings with columns name, mtx, mdl (string with R model with mtx.row)

Value

An updated beam.specs object that includes the column "plot"

Examples

```
data(clinf)
data(omicdat)
data(omicann)
data(setdat)
test.beam.data <- prep_beam_data(main.data=clinf, mtx.data=omicdat,
                                mtx.anns=omicann, set.data=setdat,
                                set.anns=NULL, n.boot=10, seed=123)
specs <- prep_beam_specs(beam.data=test.beam.data, endpts=c("MRD29", "EFS", "OS"),
                          firth=TRUE)
plot.specs <- prep_beam_plot(beam.data=test.beam.data, beam.specs=specs)
```

prep_beam_specs	<i>Prepare beam.specs</i>
-----------------	---------------------------

Description

Prepare the beam.specs data.frame for BEAM model fitting. Specifies the univariate models needed to compute the BEAMR set p-values.

Usage

```
prep_beam_specs(
  beam.data,
  endpts,
  firth = TRUE,
  adjvars = NULL,
  endptmdl = NULL
)
```

Arguments

beam.data	A beam.data object from prep_beam_data
endpts	A vector of endpoint variable names in main.data
firth	A logical value. If TRUE (default) fit Firth penalized Cox model to account for monotone likelihood in the presence of rare events or predictors. If FALSE fit usual Cox model.

adjvars	Default NULL, optional vector of adjustment variable names in main.data
endptmdl	Optional model specification data.frame with endpoint name column called "endpt" and model string column called "mdl"

Value

The beam.specs object, a data.frame specifying the omics-endpoint association models to be fit

Examples

```
data(clinf)
data(omicdat)
data(omicann)
data(setdat)
test.beam.data <- prep_beam_data(main.data=clinf, mtx.data=omicdat,
                                mtx.anns=omicann, set.data=setdat,
                                set.anns=NULL, n.boot=10, seed=123)

#Without adjustment
prep_beam_specs(beam.data=test.beam.data, endpts=c("MRD29", "OS", "EFS"),
                firth=TRUE)
# With adjustment
prep_beam_specs(beam.data=test.beam.data, endpts=c("OS", "EFS"),
                adjvars=c("MRD29"), firth=TRUE)
```

print.beam.data	<i>Print summary information about a beam.data object</i>
-----------------	---

Description

Print summary information about a beam.data object

Usage

```
## S3 method for class 'beam.data'
print(x, ...)
```

Arguments

x	An object of class "beam.data"
...	Other arguments passed to or from other methods

Value

Messages about the beam.data object

Examples

```
data(beam_dat)
print(beam_dat)
```

<code>print.beam.stats</code>	<i>Print summary information about beam.stats object</i>
-------------------------------	--

Description

Print summary information about beam.stats object

Usage

```
## S3 method for class 'beam.stats'
print(x, ...)
```

Arguments

<code>x</code>	An object of class "beam.stats"
<code>...</code>	Other arguments passed to or from other methods

Value

Messages about the beam.data object

Examples

```
data(beam_stats)
print(beam_stats)
```

<code>setdat</code>	<i>Map of Pediatric Data from COG trial AALL0434</i>
---------------------	--

Description

Map between annotation and omic data for a subset of clinical data from pediatric and young adult t-lineage acute lymphoblastic leukemia patients in the Children's Oncology Group trial AALL0434, published in Liu et al., 2017 Nature Genetics

Usage

```
setdat
```

Format

setdat:

A data frame with 40 rows and 3 columns

set.id Ensembl ID that defines gene-feature set

mtx.id Name of omic matrix where corresponding feature data can be found

row.id Feature name in corresponding omic matrix

Source

<https://www.nature.com/articles/ng.3909>

specs	<i>Pediatric T-ALL BEAMR Analysis Specs Data from COG trial AALL0434</i>
-------	--

Description

The beam.specs object used in example beam analyses

Usage

specs

Format

specs:
A data frame with 6 rows and 3 columns:
name Analysis name with omic and endpoint
mtx Name of omics matrix used in the analysis
mdl Regression model

Source

NA

subset_beam_result	<i>Subset beam.stats Result</i>
--------------------	---------------------------------

Description

Filter the beam.stats object from compute_beam_stats with various filtering criteria. Default is to filter to top 50 sets with smallest q-value. At least one filtering criteria must be specified. Can also use intersection or union of multiple criteria.

Usage

```
subset_beam_result(
  beam.result,
  beam.set.pvals = NULL,
  beam.feats.pvals = NULL,
  mtx.rows = NULL,
  set.ids = NULL,
  endpts = NULL,
  omics = NULL,
  p.limit = NULL,
  q.limit = NULL,
  p.feats.limit = NULL,
  q.feats.limit = NULL,
  intersect = TRUE,
  recalc = FALSE
)
```

Arguments

<code>beam.result</code>	A beam.stats object from <code>compute_beam_stats</code>
<code>beam.set.pvals</code>	A list containing BEAMR set p-values from <code>compute_set_pvalues</code> ; required if <code>p.limit</code> or <code>q.limit</code> are specified.
<code>beam.feats.pvals</code>	A list containing feature-level p-values from <code>compute_feature_pvalues</code> ; required if <code>p.feats.limit</code> or <code>q.feats.limit</code> are specified.
<code>mtx.rows</code>	A list of vectors of feature names corresponding to <code>row.id</code> in <code>set.data</code> . List names correspond to <code>mtx.id</code> in <code>set.data</code> . If specified, filter to all sets containing at least one of these features.
<code>set.ids</code>	A character vector of <code>set.ids</code> . If specified, filter to these sets.
<code>endpts</code>	A character vector of endpoint names. If specified, filter to sets that correspond to these endpoints.
<code>omics</code>	A character vector of omics names. If specified, filter to sets that correspond to these omics.
<code>p.limit</code>	A numeric value. If specified, determine <code>mtx.rows</code> that are below this threshold if $p < 1$ or top p sets if $p > 1$.
<code>q.limit</code>	A numeric value. If specified, determine <code>mtx.rows</code> that are below this threshold if $q < 1$ or top q sets if $q > 1$.
<code>p.feats.limit</code>	A numeric value. If specified, determine <code>mtx.rows</code> that are below this threshold if $p.feats < 1$ or top $p.feats$ sets if $p.feats > 1$ (feature p-values).
<code>q.feats.limit</code>	A numeric value. If specified, determine <code>mtx.rows</code> that are below this threshold if $q.feats < 1$ or top $q.feats$ sets if $q.feats > 1$.
<code>intersect</code>	A logical value. Default is TRUE. If TRUE, use intersection of all specified criteria. If FALSE use union of all specified criteria.
<code>recalc</code>	A logical value. Default is FALSE. If TRUE, recalculate p-values. If FALSE use original set p-values..

Value

A list with filtered beam.stats object, updated beam.set.pvals, and filtered beam.feat.pvals.

Examples

```
data(bam_stats)
test.pvals <- compute_set_pvalues(bam_stats=bam_stats)
test.feat.pvals <- compute_feature_pvalues(bam_stats=bam_stats)
filt.bam_stats <- subset_beam_result(bam_stats, test.pvals, test.feat.pvals,
                                   endpts=c("EFS", "OS"), q.limit=10, intersect=TRUE,
                                   recalc=FALSE)
```

Index

* datasets

- beam_dat, [2](#)
- beam_dat_sm, [3](#)
- beam_specs, [4](#)
- beam_specs_sm, [4](#)
- beam_stats, [5](#)
- beam_stats_sm, [5](#)
- clinf, [7](#)
- omicann, [14](#)
- omicdat, [14](#)
- setdat, [22](#)
- specs, [23](#)

- beam_dat, [2](#)
- beam_dat_sm, [3](#)
- beam_specs, [4](#)
- beam_specs_sm, [4](#)
- beam_stats, [5](#)
- beam_stats_sm, [5](#)

- check_beam_specs, [6](#)
- check_list_class, [6](#)
- clean_Bmtx, [7](#)
- clinf, [7](#)
- compute_beam_stats, [8](#)
- compute_feature_pvalues, [9](#)
- compute_set_pvalues, [9](#)

- extend_set_data, [10](#)
- extract_beam_stats, [11](#)

- find_id_clm, [11](#)

- gen_beam_plot_list, [12](#)
- get_id_index, [13](#)

- omicann, [14](#)
- omicdat, [14](#)

- plot_beam_boot, [15](#)
- plot_beam_clin, [16](#)

- plot_feat_clin, [17](#)
- prep_beam_data, [18](#)
- prep_beam_plot, [19](#)
- prep_beam_specs, [20](#)
- print.beam.data, [21](#)
- print.beam.stats, [22](#)

- setdat, [22](#)
- specs, [23](#)
- subset_beam_result, [23](#)