Package 'httrpathway'

April 8, 2025

Type Package

Title Pathway Scoring and Concentration Response for high-throughput transcriptomics (HTTr) data

Version 0.2

Author Thomas Sheffield

Maintainer Derik Haggard <haggard.derik@epa.gov>

Description

This package generates pathway (signature) scores with associated concentration response modeling. This package contains functions to create input files (log2-fold change, or (l2fc) matrices) and run concentration-response modeling of signature or gene/probe-level HTTr data.

Imports stats,
stringr,
grDevices,
graphics,
utils,
methods,
future.apply,
future,
moments,
numDeriv,
openxlsx,
parallel,
RColorBrewer,
reshape2,
data.table,
e1071,
mongolite,
testthat,
tcplfit2,
stringi,
matrixStats,
tidyr,
gplots,
ggplot2

biocViews GSVA (>= 1.52.3)

License MIT + file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 7.3.2

Suggests knitr,
rmarkdown,
httrlib

VignetteBuilder knitr

Contents

auc *Area Under the Curve*

Description

Compute AUC for an ROC curve.

Usage

```
auc(tpr, fpr)
```

Arguments

tpr	Vector of true positive rates.
fpr	Vector of false positive rates.

Details

Uses trapezoid rule numerical integration to approximate AUC. Will be more accurate with more fine-grained inputs.

Value

AUC

Examples

```
auc(c(0,.5,1), c(0,.5,1))
auc(c(0,1,1), c(0,.5,1))
```

buildFCMAT2 *Transpose and filter the fold change matrix FCMAT1 in long format into a gene x sample format. Function returns FCMAT2 and CHEM_DICT as a list R object*

Description

Transpose and filter the fold change matrix FCMAT1 in long format into a gene x sample format. Function returns FCMAT2 and CHEM_DICT as a list R object

Usage

```
buildFCMAT2(
  FCMAT1 = NULL,
  time = 24,
  media = "DMEM",
  output_dir = "../input/fcdata/",
  method = "gene",
  sample_type = "test sample",
  writing_to_disk = FALSE
)
```

Arguments

FCMAT1	The input data structure that contains the data frame from the httrlib buildFCMAT1() function or provided by the user in the same format. FCMAT1 should be a long format table with rows denoting genes/probes and an 'l2fc' column which contains log2 fold-change data from DESeq2 or another method.
time	The time in hours that the chemical dosing was run
media	The name of the media used
output_dir	The directory from which to read all of the raw files
method	Either "gene" or "probe_id". Specifying "gene" will average probes that target the same gene, whereas "probe_id" will keep all probes separate when making the FCMAT2 table.
sample_type	either "test sample" or "reference chemical" will determine what sample type the FCMAT2/CHEM_DICT tables will include
writing_to_disk	dictates whether all results should be saved to disk (path is defined by output_dir parameter)

Value

Global variables are created for the FC matrix (FCMAT2) and the chemical dictionary (CHEM_DICT) which translates from the sample key (sample_id_conc_time) to the individual components

list consisting of matrices FCMAT2 and CHEM_DICT

---

compareSignatureCutoffs

Generates a plot comparing two cutoff calculation methods

Description

Generates a plot comparing two cutoff calculation methods

Usage

```
compareSignatureCutoffs(empirical_cutoffs, cutoffs, nsig = NULL)
```

Arguments

empirical_cutoffs	output from the exportSignatureCutoffs function
cutoffs	calculated cutoffs provided by cutoffCalc function
nsig	number of signatures to compare

Value

ggplot object which is the scatterplot comparing the two methods

---

cutoffCalc *Calculate the signature-wise cutoffs based on the analytical method which does not break any correlations between genes and compare with already defined signature_cr object (see exportSignatureCutoffs function)*

Description

Calculate the signature-wise cutoffs based on the analytical method which does not break any correlations between genes and compare with already defined signature_cr object (see exportSignatureCutoffs function)

Usage

```
cutoffCalc(
  sigcatalog,
  sigset,
  pval = 0.05,
  seed = 12345,
  nlowconc = 2,
  mc.cores = 1,
  dtxsid.exclude = NULL,
  do.cov = T,
  do.compare = F,
  to.file = F,
  sigdbgenelist,
  FCMAT2,
  CHEM_DICT
)
```

Arguments

sigcatalog	The name of the signature catalog to use
sigset	The signature set
pval	The p-value for the baseline distribution

seed	Random seed
nlowconc	Only include the nlowconc concentrations for each chemical when calcualting the cutoff
mc.cores	Number of cores to use when running parallel
dtxsid.exclude	dtxsids to exclude, default NULL
do.cov	If TRUE, calculate the covariance matrix and store in a global
do.compare	If TRUE, compare the cutoffs with those from the original method with no gene-gene correlation
to.file	If TRUE, and do.compare=TRUE, send a plot of the comparison to a file
sigdbgenelist	full path to the signatureDB_genelist.RDS file
FCMAT2	Sample by gene matrix of log2(fold change)'s. Rownames are sample keys and colnames are genes.
CHEM_DICT	Dataframe with one row per sample key and seven columns: sample_key, sample_id, conc, time, casrn, name, dtxsid.

Value

calculated cutoffs dataframe

`cutoffCalc.inner.empirical`

Inner function for the cutoff calculation based on the empirical distributions

---

Description

Inner function for the cutoff calculation based on the empirical distributions

Usage

```
cutoffCalc.inner.empirical(signature, pval, SIGSCOREMAT)
```

Arguments

signature	The name of the signature for which the cutoff is to be calculated
pval	The p-value for the baseline distribution
SIGSCOREMAT	The signature score matrix

Value

vector containing the signature, cutoff, sd (one standard deviation of the signature score matrix), bmed (median of the signature score matrix)

cutoffCalc.inner.fc

Inner function for the cutoff calculation based on the analytical method

Description

Inner function for the cutoff calculation based on the analytical method

Usage

```
cutoffCalc.inner.fc(parent, catalog, allgenes, pval)
```

Arguments

parent	The name of the signature parent for which the cutoff is to be calculated
catalog	The signature catalog
allgenes	The list of all the genes in the data set
pval	The p-value for the baseline distribution

Value

vector containing the parent (signature), cutoff, sd, bmed

cutoffCalcEmpirical

Calculate the signature-wise cutoffs based on the empirical distributions which does not break any correlations between genes

Description

Calculate the signature-wise cutoffs based on the empirical distributions which does not break any correlations between genes

Usage

```
cutoffCalcEmpirical(
  pval = 0.05,
  nlowconc = 2,
  mc.cores = 1,
  dtxsid.exclude = NULL,
  signaturescoremat,
  CHEM_DICT
)
```

Arguments

pval	The p-value for the baseline distribution
nlowconc	Only include the lowest nlowconc concentrations for each chemical
mc.cores	Number of cores to use when running parallel
dtxsid.exclude	dtxsids to exclude, default NULL
signaturescoremat	Dataframe with one row per chemical/conc/signature combination. Columns are: sample_id, dtxsid, casrn, name, time, conc, sigset, signature, size, signature_score
CHEM_DICT	Dataframe with one row per sample key and seven columns: sample_key, sample_id, conc, time, casrn, name, dtxsid.

Value

signature-wise cutoffs dataframe

---

exportSignatureCutoffs *Export the signature-wise cutoffs*

Description

Export the signature-wise cutoffs

Usage

```
exportSignatureCutoffs(signature_cr)
```

Arguments

signature_cr	dataframe returned by the signatureConcResp function

---

geneConcResp *Gene Concentration Response*

Description

Wrapper that performs concentration response modeling for gene/probe log2 fold-change values

Usage

```
geneConcResp(
  mc.cores = 20,
  to.file.path = NULL,
  pval = 0.05,
  aicc = F,
  fitmodels = c("cnst", "hill", "poly1", "poly2", "pow", "exp2", "exp3", "exp4", "e
  genefile = NULL,
  FCMAT2,
  CHEM_DICT
)
```

Arguments

mc.cores	Number of parallel cores to use.
to.file.path	when provided, path of RDS file where results are written to
pval	P-value cutoff between 0 and 1.
aicc	If aicc = T, corrected AIC is used instead of first order (regular) AIC.
fitmodels	Vector of models names to be used. Default is all of them.
genefile	An optional file (.xlsx) that can be used to filter concentration-response modeling for a subset of genes of interest
FCMAT2	chem/conc by gene or chem/conc by probe. Uses two lowest concentration of each column to estimate noise cutoff (as opposed to signature CR).
CHEM_DICT	Dataframe with one row per sample key and seven columns: sample_key, sample_id, conc, time, casrn, name, dtxsid.

Details

Uses two lowest concentration of each column to estimate noise cutoff (as opposed to signature CR).

Value

dataframe of concentration response modeling results

GSEA *My Gene Set Enrichment Analysis*

Description

Performs tweaked version of single sample GSEA.

Usage

```
GSEA(
  X,
  geneSets,
  min.sz = 1,
  max.sz = Inf,
  alpha = 0.25,
  verbose = T,
  useranks = T
)
```

Arguments

X	Transposed FCMAT2; i.e a gene by sample matrix of l2fc's including rownames and colnames. Equivalent to expr in gsva.
geneSets	Named list of signature definitions. Each element is a vector of gene names. Each element name is a signature name.E quivalent to gset.idx.list in gsva.
min.sz	Minimum signature size (deprecated).
max.sz	Maximum signature size (deprecated)
alpha	Power of R to use. Higher alpha will upweight more extreme ranks relative to middle ranks.
verbose	verbose = T prints gene set length message.
useranks	useranks = T uses ranks as in ssGSEA, while useranks = F uses the bare fold changes instead.

Details

Based on the GSVA ssGSEA code. Main changes are: NAs are now handled correctly and rank is now centered on zero instead of beginning at one. Since signature sizes are undercounted here due to missing values, they are assessed more accurately in signatureScoreCoreGSEA and limits are enforced after scoring.

Value

Outputs signature by sample matrix of signature scores.

Examples

```
geneSets = list(signature1 = c("ABC", "DEF"), signature2 = c("ABC", "GHI"))
X = matrix(c(1:3,3:1), nrow = 3)
colnames(X) = c("Sample1", "Sample2")
rownames(X) = c("ABC", "DEF", "GHI")
GSEA(X,geneSets)
```

`myrndwalk` *GSVA:::rndwalk but able to handle sum(Ralpha) = 0*

Description

GSVA:::rndwalk but able to handle sum(Ralpha) = 0

Usage

```
myrndwalk(gSetIdx, geneRanking, j, R, alpha)
```

Arguments

`gSetIdx`	gSetIdx
`geneRanking`	geneRanking
`j`	j
`R`	R
`alpha`	alpha

`printCurrentFunction` *Print the name of the current function*

Description

Print the name of the current function

Usage

```
printCurrentFunction(comment.string = NA)
```

Arguments

`comment.string`	An optional string to be printed

Value

nothing

R2 *R Squared*

Description

Calculate coefficient of determination.

Usage

```
R2(y, pred)
```

Arguments

y	Vector of actual values.
pred	Vector of corresponding predicted values.

Details

Note that order matters: R2(x,y) != R2(y,x) in general.

Value

Coefficient of determination.

Examples

```
R2(c(1:10), c(1:10*.8))
R2(c(1:10*.8), c(1:10))
```

randomdata *Randomized Null Data*

Description

Generate randomized null data based on actual data.

Usage

```
randomdata(
  nchem = 1000,
  seed = 12345,
  maxconc = 1e+06,
  nlowconc = 2,
  dtxsid.exclude = NULL,
  FCMAT2,
  CHEM_DICT,
```

```
  output_dir = "../data/",
  writing_to_disk = FALSE
)
```

Arguments

nchem	Number of null chemicals. Number of null samples is approximately eight times this value.
seed	Random seed.
maxconc	Only use concentrations less than maxconc, default 1000000
nlowconc	If not NULL, only include the lowest nlowconc concentrations for each chemical
dtxsid.exclude	dtxsids to exclude, default NULL
FCMAT2	Sample by gene matrix of log2(fold change)'s. Rownames are sample keys and colnames are genes.
CHEM_DICT	Data frame with one row per sample key and seven columns: sample_key, sample_id, conc, time, casrn, name, dtxsid.
output_dir	full path to directory/file where the CHEM_DICT/FCMAT2 data frames should be saved
writing_to_disk	boolean, TRUE if the resulting CHEM_DICT/FCMAT2 data frames are to be saved to disk

Details

Randomization is performed by sampling the quantile function for each gene in the actual data. The null set will have roughly the same distribution of values for each gene in the actual data.

Value

list of the randomized FCMAT2 and CHEM_DICT data frame

RMSE *Root-mean-square-error*

Description

Computes root-mean-square-error between two vectors.

Usage

```
RMSE(x, y)
```

Arguments

x	First vector.
y	Second vector.

Value

RMSE

Examples

```
RMSE(1:3, c(1,3,5))
```

---

rowMean *Utility function used by cutoffCalc.inner.fc*

Description

Utility function used by cutoffCalc.inner.fc

Usage

```
rowMean(x)
```

Arguments

x	= matrix or data.frame

Value

vector of mean value of each row

---

runAllSignatureCR *Run All Pathway Concentration Response (P-Value)*

Description

Driver for signature scoring and concentration response (CR).

Usage

```
runAllSignatureCR(
  dataset,
  sigset,
  sigcatalog = "../inst/extdata/signatureDB_master_catalog_2022-05-16.xlsx",
  method,
  bmr_scale = 1.349,
  normfactor = 7500,
  minsigsize = 10,
  pval = 0.05,
  nlowconc = 2,
  mc.cores = 1,
```

```
  fitmodels = c("cnst", "hill", "poly1", "poly2", "pow", "exp2", "exp3", "exp4", "e
  FCMAT2,
  CHEM_DICT,
  sigdbgenelist = "../inst/extdata/signatureDB_genelists.RDS"
)
```

Arguments

dataset	Name of data set.
sigset	Name of signature set.
sigcatalog	Name of the signature catalog
method	Pathway scoring method in c("fc", "gsva", "gsea")
bmr_scale	benchmark response (bmr) scaling factor. Default = 1.349
normfactor	Factor to scale the native units up by to get onto a reasonable plotting value (~ -1 to 1)
minsigsize	Minimum signature size.
pval	P-value to use for noise estimation.
nlowconc	Only include the lowest nlowconc concentrations for each chemical
mc.cores	Vector with two values: number of cores to use for signature scoring and number of cores to use for CR. CR can usually handle the maximum number, but gsva scoring might require a smaller number to avoid memory overflow.
fitmodels	Vector of model names to run conc/resp with. "cnst" should always be chosen.
FCMAT2	Sample by gene matrix of log2(fold change)'s. Rownames are sample keys and colnames are genes.
CHEM_DICT	Dataframe with one row per sample key and seven columns: sample_key, sample_id, conc, time, casrn, name, dtxsid.
sigdbgenelist	full path to signature DB gene list file; default is repo version

Value

dataframe with signature concentration-response results

---

signatureCatalogLoader

Read in signature catalog

Description

Read in signature catalog

Usage

```
signatureCatalogLoader(
  sigset = "wgcna",
  sigcatalog = "../inst/extdata/signatureDB_master_catalog_2022-05-16.xlsx",
  sigdbgenelist = "../inst/extdata/signatureDB_genelists.RDS"
)
```

Arguments

sigset	Name of the signature set to use; derived from the sigcatalog
sigcatalog	full path to signature catalog xlsx file; default is repo version
sigdbgenelist	full path to signature DB gene list file; default is repo version

Value

the trimmed signature table

---

signatureConcResp *Pathway Concentration Response (P-value)*

Description

Performs signature concentration response using p-value based cutoffs.

Usage

```
signatureConcResp(
  dataset,
  sigset,
  sigcatalog = "../inst/extdata/signatureDB_master_catalog_2022-05-16.xlsx",
  method,
  bmr_scale = 1.349,
  mc.cores = 1,
  pval = 0.05,
  nlowconc = 2,
  aicc = F,
  dtxsid.exclude = NULL,
  minsigsize = 10,
  fitmodels = c("cnst", "hill", "poly1", "poly2", "pow", "exp2", "exp3", "exp4", "e
  signaturescoremat,
  sigdbgenelist = "../inst/extdata/signatureDB_genelists.RDS",
  CHEM_DICT
)
```

Arguments

dataset	Name of the data set.
sigset	Name of the signature set.
sigcatalog	Name of the signature catalog file
method	Pathway scoring method in c("fc", "gsva", "gsea")
bmr_scale	bmr scaling factor. Default = 1.349
mc.cores	Number of cores to parallelize with.
pval	Desired cutoff p-value.
nlowconc	Only include the lowest nlowconc concentrations for each chemical
aicc	aicc = T uses corrected AIC to choose winning method; otherwise regular AIC
dtxsid.exclude	dtxsids to exclude, default NULL
minsigsize	Minimum allowed signature size. Sample/signature combinations with less than this number of non-missing l2fc's will be discarded.
fitmodels	Vector of model names to use. Probably should include "cnst".
signaturescoremat	dataframe returned by the upstream signatureScoreMerge function
sigdbgenelist	full path to signature DB gene list file; default is repo version
CHEM_DICT	Dataframe with one row per sample key and seven columns: sample_key, sample_id, conc, time, casrn, name, dtxsid.

Details

dataset should have already been scored using signatureScore and signatureScoreMerge and the given sigset and method. This function prepares signatureScore output for CR processing, calls tcplfit2::concRespCore, and formats the output.

Value

dataframe with signature CR output.

---

signatureScore *Signature Score*

Description

Computes and saves signature scores.

Usage

```
signatureScore(
  FCMAT2,
  CHEM_DICT,
  sigset,
  sigcatalog = "../inst/extdata/signatureDB_master_catalog_2022-05-16.xlsx",
  method,
  normfactor = 7500,
  mc.cores = 1,
  minsigsize = 10,
  sigdbgenelist = "../inst/extdata/signatureDB_genelists.RDS"
)
```

Arguments

FCMAT2	Sample by gene matrix of log2(fold change)'s. Rownames are sample keys and colnames are genes.
CHEM_DICT	Dataframe with one row per sample key and seven columns: sample_key, sample_id, conc, time, casrn, name, dtxsid.
sigset	Name of signature set.
sigcatalog	full path to signature catalog xlsx file; default is repo version
method	Signature scoring method in c("fc", "gsva", "gsea")
normfactor	Value passed to the plotting code to scale the y values
mc.cores	Number of cores to use.
minsigsize	Minimum allowed signature size BEFORE accounting for missing values.
sigdbgenelist	full path to signature DB gene list file; default is repo version

Details

signatureScore is a driver for various scoring methods. The three that are currently available are "gsva", "gsea", "fc". Beware running out of memory on large runs with gsva, Linux, and many cores – ensure your system has enough memory allocated depending on data size. Signature size is counted according to number of genes in the signature that are also in the column names of FCMAT2. However, each method performs a more rigorous size count internally that accounts for missing values and adds this to the output.

Value

Returns data frame of signature scores

signatureScoreCoreFC *Signature Score Core - FC*

Description

Computes fold change signature scores.

Usage

```
signatureScoreCoreFC(
  fcmat,
  sigset,
  chem_dict,
  signature_data,
  ngenemax = NULL,
  verbose = F
)
```

Arguments

fcmat	Expects FCMAT2. Sample by gene matrix of log2(fold change)'s. Rownames are sample keys and colnames are genes.
sigset	Name of signature set.
chem_dict	Expects CHEM_DICT object. Dataframe with one row per sample key and seven columns: sample_key, sample_id, conc, time, casrn, name, dtxsid.
signature_data	Named list of gene name vectors. Each element is one signature, defined by the genes it contains.
ngenemax	If ngene is not NULL, then only the most extreme n genes of the signature will be used for the "in" set
verbose	If TRUE, write extra diagnostic output

Details

This fast implementation of fold change signature scores uses matrix multiplication. The score is simply: mean(fold change of genes in signature) - mean(fold change of genes outside signature).

Value

Dataframe with one row per chemical/conc/signature combination. Columns are: sample_id, dtxsid, casrn, name, time, conc, sigset, signature, size (signature size accounting for missing values), mean_fc_scaled_in, mean_fc_scaled_out, signature_score.

`signatureScoreCoreGSEA` *Signature Score Core - GSEA*

Description

Computes signature scores for gsea.

Usage

```
signatureScoreCoreGSEA(
  sk.list,
  normfactor = 7500,
  sigset,
  fcmat,
  chem_dict,
  signature_data,
  mc.cores = 1,
  normalization = T,
  useranks = T
)
```

Arguments

`sk.list`	Sample keys to use; should correspond to fcmat rownames.
`normfactor`	= proceed on first 1/normfactor of GSEA data
`sigset`	Name of signature set.
`fcmat`	Expects FCMAT2. Sample by gene matrix of log2(fold change)'s. Rownames are sample keys and colnames are genes.
`chem_dict`	Expects CHEM_DICT object. Dataframe with one row per sample key and seven columns: sample_key, sample_id, conc, time, casrn, name, dtxsid.
`signature_data`	Named list of gene name vectors. Each element is one signature, defined by the genes it contains.
`mc.cores`	Number of cores to use.
`normalization`	normalization = T normalizes final scores.
`useranks`	useranks = T uses score ranks for weighting; otherwise, fold changes are used for weights.

Details

This function is a parallelized wrapper for gsea, which does the actual scoring. gsea method uses ranks and normalizationby default. Normalization divides final scores by difference between max and min score. Without normalization, scores from individual samples have no impact on each other.

Value

Dataframe with one row per chemical/conc/signature combination. Columns are: sample_id, dtxsid, casrn, name, time, conc, sigset, signature, size, signature_score

---

`signatureScoreCoreGSVA`

Signature Score Core - GSVA

Description

Computes GSVA signature scores.

Usage

```
signatureScoreCoreGSVA(
  sk.list,
  sigset = "FILTERED",
  fcmat,
  chem_dict,
  signature_data,
  mc.cores = 1
)
```

Arguments

`sk.list`	Sample keys to use; should correspond to fcmat rownames.
`sigset`	Name of signature set.
`fcmat`	Expects FCMAT2. Sample by gene matrix of log2(fold change)'s. Rownames are sample keys and colnames are genes.
`chem_dict`	Expects CHEM_DICT object. Dataframe with one row per sample key and seven columns: sample_key, sample_id, conc, time, casrn, name, dtxsid.
`signature_data`	Named list of gene name vectors. Each element is one signature, defined by the genes it contains.
`mc.cores`	Number of cores to use. Parallelization is performed by gsva itself.

Details

This function is a wrapper for GSVA with Gaussian cumulative distribution function (cdf) kernels.

Value

Dataframe with one row per chemical/conc/signature combination. Columns are: sample_id, dtxsid, casrn, name, time, conc, sigset, signature, size, signature_score

signatureScoreMerge *Merge the up and down halves of the directional pathway data*

Description

Merge the up and down halves of the directional pathway data

Usage

```
signatureScoreMerge(
  sigset,
  sigcatalog = "../inst/extdata/signatureDB_master_catalog_2022-05-16.xlsx",
  method,
  signaturescoremat,
  sigdbgenelist = "../inst/extdata/signatureDB_genelists.RDS"
)
```

Arguments

sigset	Name of the signature set.
sigcatalog	Name of the catalog file
method	Pathway scoring method in c("fc", "gsva", "gsea")
signaturescoremat	dataframe returned by the upstream signatureScore function
sigdbgenelist	full path to signature DB gene list file; default is repo version

Value

signaturescoremat dataframe

TxT *Calculate several statistics on a 2 x 2 matrix*

Description

Calculate several statistics on a 2 x 2 matrix

Usage

```
TxT(tp, fp, fn, tn, do.p = TRUE, rowname = NA)
```

Arguments

`tp`	number of true positives
`fp`	number of false positives
`fn`	number of false negatives
`tn`	number of true negatives
`do.p`	if TRUE, calculate an exact p-value
`rowname`	if not NA, add a column to the output with this rowname

Value

Returns a list of the results with the following columns: TP (true positives), FP (false positives), FN (false negatives), TN (true negatives), Sens (sensitivity), Spec (specificity), BA (Balanced Accuracy), Acrcy (Accuracy), RelRsk Relative Risk, OR Odds Ratio, CI.OR.LWR (lower confidence interval of the Odds Ratio), CI.OR.UPR (upper confidence interval of the Odds Ratio), PPV (Positive Predictive Value), NPV (Negative Predictive Value), F1 (2TP/(2TP+FP+FN)), p.value (Chi-squared p-value)

`WRMSE` *Weighted Root-mean-square-error*

Description

Computes root-mean-square error with weighted average.

Usage

```
WRMSE(x, y, w)
```

Arguments

`x`	First vector of numbers.
`y`	Second vector of numbers.
`w`	Vector of weights.

Details

x,y,w should all be the same length. Order of x and y won't change output.

Value

Weighted RMSE.

Examples

```
WRMSE(1:3, c(1,3,5), 1:3)
```

Index