Package 'sigPathway'

October 8, 2014

Type Package

Title Pathway Analysis

Version 1.32.1

Date 2008-10-19

Author Weil Lai (optimized R and C code), Lu Tian and Peter Park (algorithm development and initial R code)

Maintainer Weil Lai <wlai@alum.mit.edu>

Depends R (>= 2.10)

Suggests hgu133a.db (>= 1.10.0), XML (>= 1.6-3), AnnotationDbi (>= 1.3.12)

Description Conducts pathway analysis by calculating the NT_k and NE_k statistics as described in Tian et al. (2005)

License GPL-2

URL http://www.pnas.org/cgi/doi/10.1073/pnas.0506577102,http://www.chip.org/~ppark/Supplements/PNAS05.html

biocViews DifferentialExpression, MultipleComparison

R topics documented:

calcTNullFast 2
calcTStatFast 3
calculate.GSEA 4
calculate.NGSk 5
calculatePathwayStatistics 7
estimateNumPerm 10
getPathwayStatistics 11
getPathwayStatistics.NGSk 12
importGeneSets 13
MuscleExample 14

rankPathways 15
rankPathways.NGSk 16
runSigPathway 17
selectGeneSets 19
writeSigPathway 21

Index **23**

`calcTNullFast` *Compute Null T Distribution for Each Gene*

Description

Computes a null t distribution for each gene by permuting the phenotypes.

Usage

```
calcTNullFast(tab, phenotype, nsim, ngroups = 2, allphenotypes = FALSE)
```

Arguments

`tab`	a numeric matrix of expression values, with the rows and columns representing probe sets and sample arrays, respectively
`phenotype`	a numeric (or character if `ngroups` >= 2) vector indicating the phenotype
`nsim`	an integer indicating the number of permutations to use
`ngroups`	an integer indicating the number of groups in the expression matrix
`allphenotypes`	a boolean indicating whether the function should consider all possible permutations of the phenotype, including the original, non-permuted phenotype

Details

Similar to `calcTStatFast` but calculates t-statistics over permuted phenotypes. If `allphenotypes == FALSE`, then any permutation that has a permuted phenotype equal to the original phenotype will be repermuted. For example, all the possible permutations for `phenotype == c(0,0,1,1)` are `c(0,0,1,1)`, `c(0,1,0,1)`, `c(1,0,1,0)`, `c(1,0,0,1)`, `c(0,1,1,0)`, and `c(1,1,0,0)`. If `allphenotypes == FALSE`, then the results will not include values from the `c(0,0,1,1)` case.

The help file of `calcTStatFast` has more details on the different statistics one can calculate based on the value specified for `ngroups`.

Value

A matrix with `nsim` rows and `nrow(tab)` columns.

Author(s)

Weil Lai

`calcTStatFast` *Compute T-Statistics and Corresponding P-Values*

Description

Computes t-statistics and corresponding p-values.

Usage

```
calcTStatFast(tab, phenotype, ngroups = 2)
```

Arguments

`tab`	a numeric matrix of expression values, with the rows and columns representing probe sets and sample arrays, respectively
`phenotype`	a numeric (or character if `ngroups` >= 2) vector indicating the phenotype
`ngroups`	an integer indicating the number of groups in the expression matrix

Details

If there are two groups in the matrix, it is recommended to use 0 and 1 to denote which sample columns belong to which group. If the phenotype is a character vector, then the phenotype ranked first in the alphabet is considered as 0.

If `ngroups` = 2, the t-test done here is equivalent to a unpaired two-sample t-test, assuming unequal variances. Please note that as of version 1.1.6, the sign of the t-statistic is positive when the mean of group 1 is greater than the mean of group 0.

If there is only one group in the matrix (e.g., Alzheimer's data set as reanalyzed in Tian et al. (2005)), then the phenotype vector should consist of continuous values. In this case, the association between phenotype and expression values is first calculated as Pearson correlation coefficients, transformed to Fisher's z, and then rescaled so that its variance is 1:

z = 0.5*log((1+rho)/(1-rho))*sqrt(n-3), where n is the number of phenotypes.

If `ngroups` > 2, the f-statistics (from 1-way ANOVA) are calculated. The user will need to check that the data have similar variances among the groups.

Value

`pval`	A vector of unadjusted p-values
`tstat`	A vector of t-statistics (`ngroups` = 2) or rescaled Fisher's z (`ngroups` = 1)
`rho`	(Also returned when `ngroups` = 1) A vector of Pearson correlation coefficients

Author(s)

Weil Lai

Examples

```
## Load inflammatory myopathy data set
data(MuscleExample)
statList <- calcTStatFast(tab, phenotype, ngroups = 2)

## Generate histogram of p-values
hist(statList$pval, xlab = "Unadjusted p-values", ylab = "Frequency")
```

`calculate.GSEA` *Calculate 2-sided statistics based on the GSEA algorithm*

Description

Calculates the 2-sided statistics based on the GSEA algorithm.

Usage

```
calculate.GSEA(tab, phenotype, gsList, nsim = 1000,
               verbose = FALSE, alwaysUseRandomPerm = FALSE)
```

Arguments

`tab`	a numeric matrix of expression values, with the rows and columns representing probe sets and sample arrays, respectively
`phenotype`	a numeric or character vector indicating the phenotype
`gsList`	a list containing three vectors from the output of the `selectGeneSets` function
`nsim`	an integer indicating the number of permutations to use
`verbose`	a boolean to indicate whether to print debugging messages to the R console
`alwaysUseRandomPerm`	a boolean to indicate whether the algorithm can use complete permutations for cases where `nsim` is greater than the total number of unique permutations possible with the `phenotype` vector

Details

This function assumes 2 distinct types of phenotypes in the data. It calculates a variant of the GSEA statistics (Mootha et al.) with the following modifications: (a) GSEA was changed from a 1-sided to a 2-sided approach. (b) The 2-group t-statistics is used as the difference metric.

The function also normalizes the GSEA statistic and calculates the corresponding q-values for each gene set as described in Tian et al. (2005) The function's output can be used for further analysis in other functions such as `rankPathways.NGSk` or `getPathwayStatistics.NGSk`.

Value

A list containing

ngs	number of gene sets
nsim	number of permutations performed
t.set	a numeric vector of Tk statistics
t.set.new	a numeric vector of NTk statistics
p.null	the proportion of nulls
p.value	a numeric vector of p-values
q.value	a numeric vector of q-values

Author(s)

Lu Tian, Peter Park, and Weil Lai

References

Mootha V.K., Lindgren C.M., Eriksson K.F., Subramanian A., Sihag S., Lehar J., Puigserver P., Carlsson E., Ridderstrale M., Laurila E., Houstis N., Daily M.J., Patterson N., Mesirov J.P., Golud T.R., Tamayo P., Spiegelman B., Lander E.S., Hirshhorn J.N., Altshuler D., Groop L.C. (2003) PGC-1alpha-responsive genes involved in oxidative phosphorylation are coordinately downregulated in human diabetes. *Nature Genetics*, **34**, 267-73.

Tian L., Greenberg S.A., Kong S.W., Altschuler J., Kohane I.S., Park P.J. (2005) Discovering statistically significant pathways in expression profiling studies. *Proceedings of the National Academy of Sciences of the USA*, **102**, 13544-9.

http://www.pnas.org/cgi/doi/10.1073/pnas.0506577102

---

calculate.NGSk *Calculate NGSk (NTk-like) statistics with gene label permutation*

Description

Calculates the NGSk (NTk-like) statistics with gene label permutation and the corresponding p-values and q-values for each selected pathway.

Usage

```
calculate.NGSk(statV, gsList, nsim = 1000, verbose = FALSE,
               alwaysUseRandomPerm = FALSE)
```

Arguments

statV	a numeric vector of test statistic (not p-values) for each individual probe/gene
gsList	a list containing three vectors from the output of the selectGeneSets function
nsim	an integer indicating the number of permutations to use
verbose	a boolean to indicate whether to print debugging messages to the R console
alwaysUseRandomPerm	a boolean to indicate whether the algorithm can use complete permutations for cases where nsim is greater than the total number of unique permutations possible with the phenotype vector

Details

This function is a generalized version of NTk calculations; calculate.NTk calls this function internally. To use this function, the user must specify a vector of test statistics (e.g., t-statistic, Wilcoxon). Pathways from this function can be ranked with rankPathways.NGSk or with rankPathways when combined with results from another pathway analysis algorithm (e.g., calculate.NEk).

Value

A list containing

ngs	number of gene sets
nsim	number of permutations performed
t.set	a numeric vector of Tk/Ek statistics
t.set.new	a numeric vector of NTk/NEk statistics
p.null	the proportion of nulls
p.value	a numeric vector of p-values
q.value	a numeric vector of q-values

Author(s)

Lu Tian, Peter Park, and Weil Lai

References

Tian L., Greenberg S.A., Kong S.W., Altschuler J., Kohane I.S., Park P.J. (2005) Discovering statistically significant pathways in expression profiling studies. *Proceedings of the National Academy of Sciences of the USA*, **102**, 13544-9.

http://www.pnas.org/cgi/doi/10.1073/pnas.0506577102

Examples

```
## Load in filtered, expression data
data(MuscleExample)

## Prepare the pathways to analyze
probeID <- rownames(tab)
gsList <- selectGeneSets(G, probeID, 20, 500)

nsim <- 1000
ngroups <- 2
verbose <- TRUE
weightType <- "constant"
methodName <- "NGSk"
npath <- 25
allpathways <- FALSE
annotpkg <- "hgu133a.db"

statV <- calcTStatFast(tab, phenotype, ngroups)$tstat
res.NGSk <- calculate.NGSk(statV, gsList, nsim, verbose)

## Summarize top pathways from NGSk
res.pathways.NGSk <-
  rankPathways.NGSk(res.NGSk, G, gsList, methodName, npath)
print(res.pathways.NGSk)

## Get more information about the probe sets means and other statistics
## for the top pathway in res.pathways.NGSk
gpsList <-
  getPathwayStatistics.NGSk(statV, probeID, G, res.pathways.NGSk$IndexG,
                            FALSE, annotpkg)
print(gpsList[[1]])

## Write table of top-ranked pathways and their associated probe sets to
## HTML files
parameterList <-
  list(nprobes = nrow(tab), nsamples = ncol(tab),
       phenotype = phenotype, ngroups = ngroups,
       minNPS = 20, maxNPS = 500, ngs = res.NGSk$ngs,
       nsim.NGSk = res.NGSk$nsim,
       annotpkg = annotpkg, npath = npath, allpathways = allpathways)

writeSP(res.pathways.NGSk, gpsList, parameterList, tempdir(),
        "sigPathway_cNGSk", "TopPathwaysTable.html")
```

`calculatePathwayStatistics`

Calculate the NTk and NEk statistics

Description

Calculates the NTk and NEk statistics and the corresponding p-values and q-values for each selected pathway.

Usage

```
calculate.NTk(tab, phenotype, gsList, nsim = 1000,
              ngroups = 2, verbose = FALSE, alwaysUseRandomPerm = FALSE)
calculate.NEk(tab, phenotype, gsList, nsim = 1000,
              weightType = c("constant", "variable"),
              ngroups = 2, verbose = FALSE, alwaysUseRandomPerm = FALSE)
```

Arguments

tab	a numeric matrix of expression values, with the rows and columns representing probe sets and sample arrays, respectively
phenotype	a numeric (or character if ngroups >= 2) vector indicating the phenotype
gsList	a list containing three vectors from the output of the selectGeneSets function
nsim	an integer indicating the number of permutations to use
weightType	a character string specifying the type of weight to use when calculating NEk statistics
ngroups	an integer indicating the number of groups in the matrix
verbose	a boolean to indicate whether to print debugging messages to the R console
alwaysUseRandomPerm	a boolean to indicate whether the algorithm can use complete permutations for cases where nsim is greater than the total number of unique permutations possible with the phenotype vector

Details

These functions calculate the NTk and NEk statistics and the corresponding p-values and q-values for each selected pathway. The output of both functions should be together to rank top pathways with the rankPathways function.

Value

A list containing

ngs	number of gene sets
nsim	number of permutations performed
t.set	a numeric vector of Tk/Ek statistics
t.set.new	a numeric vector of NTk/NEk statistics
p.null	the proportion of nulls
p.value	a numeric vector of p-values
q.value	a numeric vector of q-values

Author(s)

Lu Tian, Peter Park, and Weil Lai

References

Tian L., Greenberg S.A., Kong S.W., Altschuler J., Kohane I.S., Park P.J. (2005) Discovering statistically significant pathways in expression profiling studies. *Proceedings of the National Academy of Sciences of the USA*, **102**, 13544-9.

http://www.pnas.org/cgi/doi/10.1073/pnas.0506577102

Examples

```
## Load in filtered, expression data
data(MuscleExample)

## Prepare the pathways to analyze
probeID <- rownames(tab)
gsList <- selectGeneSets(G, probeID, 20, 500)

## Calculate NTk and weighted NEk for each gene set
## * Use a higher nsim (e.g., 2500) value for more reproducible results
nsim <- 1000
ngroups <- 2
verbose <- TRUE
weightType <- "constant"
methodNames <- c("NTk", "NEk")
npath <- 25
allpathways <- FALSE
annotpkg <- "hgu133a.db"

res.NTk <- calculate.NTk(tab, phenotype, gsList, nsim, ngroups, verbose)
res.NEk <- calculate.NEk(tab, phenotype, gsList, nsim, weightType,
                         ngroups, verbose)

## Summarize results
res.pathways <- rankPathways(res.NTk, res.NEk, G, tab, phenotype,
                             gsList, ngroups, methodNames, npath, allpathways)
print(res.pathways)

## Get more information about the probe sets means and other statistics
## for the top pathway in res.pathways
statList <- calcTStatFast(tab, phenotype, ngroups)
gpsList <-
  getPathwayStatistics(tab, phenotype, G, res.pathways$IndexG,
                       ngroups, statList, FALSE, annotpkg)
print(gpsList[[1]])

## Write table of top-ranked pathways and their associated probe sets to
## HTML files
parameterList <-
  list(nprobes = nrow(tab), nsamples = ncol(tab),
```

```
        phenotype = phenotype, ngroups = ngroups,
        minNPS = 20, maxNPS = 500, ngs = res.NTk$ngs,
        nsim.NTk = res.NTk$nsim, nsim.NEk = res.NEk$nsim,
        weightType = weightType,
        annotpkg = annotpkg, npath = npath, allpathways = allpathways)

writeSP(res.pathways, gpsList, parameterList, tempdir(), "sigPathway_cPS",
        "TopPathwaysTable.html")
```

`estimateNumPerm` *Compute the Number of Unique Permutations for a Phenotype Vector*

Description

Computes the number of unique permutations based on a vector of phenotypes and the number of groups.

Usage

```
estimateNumPerm(phenotype, ngroups)
```

Arguments

`phenotype`	a numeric (or character if `ngroups` >= 2) vector indicating the phenotype
`ngroups`	an integer indicating the number of groups in the phenotype

Details

This function calculates the number of unique permutations based on the given phenotype and the number of groups present in the phenotype. This function is used internally in `sigPathway` and attempts to avoid numeric overflow associated with multiplying out large factorials.

Value

A numeric with length 1.

Author(s)

Weil Lai

Examples

```
## One group: continuous observations
ptype1 <- c(24,25,17,26,25,16,14,17,12,15,19,20)
print(estimateNumPerm(ptype1, 1))

## Two groups
ptype2 <- c(0,1,1,0,1,0,1)
```

```
print(estimateNumPerm(ptype2, 2))

## Three groups
ptype3a <- c(2,0,1,2,0,1,2,0,0,1,1,2)
print(estimateNumPerm(ptype3a, 3))

ptype3b <- c("Banana","Apple","Lemon","Lemon","Lemon",
             "Apple","Lemon","Banana","Banana")
print(estimateNumPerm(ptype3b, 3))
```

`getPathwayStatistics` *Give the statistics for the probe sets in a pathway*

Description

Gives the statistics for the probe sets associated with a pathway.

Usage

```
getPathwayStatistics(tab, phenotype, G, index, ngroups = 2,
                     statList = NULL, keepUnknownProbes = FALSE,
                     annotpkg = NULL)
```

Arguments

`tab`	a numeric matrix of expression values, with the rows and columns representing probe sets and sample arrays, respectively
`phenotype`	a numeric (or character if `ngroups` >= 2) vector indicating the phenotype
`G`	a list containing the source, title, and probe sets associated with each curated pathway
`index`	an integer vector specifying the pathway(s) to summarize in `G`
`ngroups`	an integer indicating the number of groups in the expression matrix
`statList`	a list containing results from `calcTStatFast`
`keepUnknownProbes`	a boolean indicating whether to keep the names of probe sets not represented in tab in the summary data frame
`annotpkg`	a character vector specifying the name of the BioConductor annotation package to use to fetch accession numbers, Entrez Gene IDs, gene name, and gene symbols

Details

This function gives the mean, standard deviation, and test statistic for each probe in the pathway as indicated in `G[[index]]`.

Value

A list containing data frames (1 per pathway) with the probes' name, mean, standard deviation, the test statistic (e.g., t-test), and the corresponding unadjusted p-value.

If `ngroups = 1`, the Pearson correlation coefficient is also returned.

If a valid `annotpkg` is specified, the probes' accession numbers, Entrez Gene IDs, gene name, and gene symbols are also returned. This option only works if the probes in the gene set list `G` are manufacturer IDs corresponding to those used in making the BioConductor annotation package.

Note

See the help page of `calculate.NTk` or `calculate.NEk` for example code that uses `getPathwayStatistics`

Author(s)

Weil Lai

`getPathwayStatistics.NGSk`

Give the statistics for the probe sets in a pathway

Description

Gives the statistics for the probe sets associated with a pathway.

Usage

```
getPathwayStatistics.NGSk(statV, probeID, G, index,
                          keepUnknownProbes = FALSE, annotpkg = NULL)
```

Arguments

`statV`	a numeric vector of test statistic (not p-values) for each individual probe/gene
`probeID`	a character vector containing the names of probe sets associated with a matrix of expression values
`G`	a list containing the source, title, and probe sets associated with each curated pathway
`index`	an integer vector specifying the pathway(s) to summarize in `G`
`keepUnknownProbes`	a boolean indicating whether to keep the names of probe sets not represented in tab in the summary data frame
`annotpkg`	a character vector specifying the name of the BioConductor annotation package to use to fetch accession numbers, Entrez Gene IDs, gene name, and gene symbols

Details

This function gives the test statistic for each probe in the pathway as indicated in `G[[index]]`.

Value

A list containing data frames (1 per pathway) with the probes' name and the corresponding test statistic.

If a valid `annotpkg` is specified, the probes' accession numbers, Entrez Gene IDs, gene name, and gene symbols are also returned. This option only works if the probes in the gene set list `G` are manufacturer IDs corresponding to those used in making the BioConductor annotation package.

Note

See the help page for `calculate.NGSk` for example code that uses `getPathwayStatistics.NGSk`

Author(s)

Weil Lai

`importGeneSets` *Import gene sets stored in GMT, GMX, GRP, and XML file formats*

Description

Imports gene sets stored in GMT, GMX, GRP, and XML file formats and converts them to sigPathway's preferred format.

Usage

```
importGeneSets(fileNames, verbose = TRUE)
gmtToG(fileNames, verbose = TRUE)
gmxToG(fileNames, verbose = TRUE)
grpToG(fileNames, verbose = TRUE)
xmlToG(fileNames, verbose = TRUE)
```

Arguments

`fileNames`	a character vector specifying the file(s) containing the gene sets of interest
`verbose`	a boolean to indicate whether to print debugging messages to the R console

Details

These functions read in gene sets stored in GMT, GMX, GRP, and XML file formats and converts them to a list format that sigPathway can use. Redundant gene IDs in each gene set are removed during conversion. The `importGeneSets` function can read in GMT, GMX, GRP, and XML files in one pass. The `gmtToG`, `gmxToG`, `grpToG`, and `xmlToG` functions are specific to reading in their respective file formats.

Value

A list containing sublists representing each imported gene set. The vignette contains more details about the list structure.

Note

These functions do not check whether the files are in the correct format and will give spurious output when given files in the wrong format. The `xmlToG` function requires the `XML` package, which is available on CRAN. The `xmlToG` function also requires XML files to be formatted based on the MSigDB Document Type Definition.

Author(s)

Weil Lai

References

http://www.broad.mit.edu/cancer/software/gsea/wiki/index.php/Data_formats

---

`MuscleExample` *Subset of Inflammatory Myopathy Dataset to Demonstrate sigPathway*

Description

`MuscleExample` is an R workspace containing the following objects: (1) `tab`: a matrix of 5000 rows and 15 columns (2) `phenotype`: a indicator vector which denotes which columns in `tab` are arrays from normal (NORM) and inclusion body myositis (IBM) (3) `G`: a list containing the source, title, and the probe set IDs associated with 626 pathways

The full inflammatory myopathway dataset (which includes all probe sets and samples, including more NORM, IBM, and dermatomyositis arrays) and a more comprehensive pathway annotation list for the HG-U133A and other selected array platforms are available at http://www.chip.org/~ppark/PNAS05/

Although the objects contained in `MuscleExample` are subsets of the full dataset, the results obtained from running pathway analysis with `MuscleExample` are comparable to those obtained using the full dataset. This example dataset contains 8 IBM and 7 NORM arrays. The 5000 probe sets were selected by considering the variance of the expression values of each probe set among the 15 arrays.

Usage

```
data(MuscleExample)
```

Format

1 integer matrix, 1 numeric vector, and 1 list

Source

http://www.chip.org/~ppark/PNAS05/

References

Tian L., Greenberg S.A., Kong S.W., Altschuler J., Kohane I.S., Park P.J. (2005) Discovering statistically significant pathways in expression profiling studies. *Proceedings of the National Academy of Sciences of the USA*, **102**, 13544-9.

http://www.pnas.org/cgi/doi/10.1073/pnas.0506577102

`rankPathways` *Summarizes Top Pathways from Pathway Analyses*

Description

Summarizes top pathways from pathway analyses.

Usage

```
rankPathways(res.A, res.B, G, tab, phenotype, gsList, ngroups,
             methodNames = NULL, npath = 25, allpathways = FALSE)
```

Arguments

`res.A`	a list from the output of `calculate.NTk` or `calculate.NEk`
`res.B`	a list from the output of `calculate.NTk` or `calculate.NEk`
`G`	a list containing the source, title, and probe sets associated with each curated pathway
`tab`	a numeric matrix of expression values, with the rows and columns representing probe sets and sample arrays, respectively
`phenotype`	a numeric (or character if `ngroups` >= 2) vector indicating the phenotype
`gsList`	a list containing three vectors from the output of the `selectGeneSets` function
`ngroups`	an integer indicating the number of groups in the matrix
`methodNames`	a character vector of length 2 giving the names for `res.A` and `res.B`
`npath`	an integer indicating the number of top gene sets to consider from each statistic when ranking the top pathways
`allpathways`	a boolean to indicate whether to include the top `npath` pathways from each statistic or just consider the top `npath` pathways (sorted by the sum of ranks of both statistics) when generating the summary table

Details

This function ranks together the statistics given in `res.A` and `res.B` and summarizes the top gene sets in a tabular format similar to Table 2 in Tian et al. (2005)

Value

A data frame showing the pathways' indices in G, gene set category, pathway title, set size, res.A's statistics, res.B's statistics, the corresponding q-values, and the ranks for the top gene sets.

Note

See the help page for calculate.NTk or calculate.NEk for example code that uses rankPathways

Author(s)

Lu Tian, Peter Park, and Weil Lai

References

Tian L., Greenberg S.A., Kong S.W., Altschuler J., Kohane I.S., Park P.J. (2005) Discovering statistically significant pathways in expression profiling studies. *Proceedings of the National Academy of Sciences of the USA*, **102**, 13544-9.

http://www.pnas.org/cgi/doi/10.1073/pnas.0506577102

---

rankPathways.NGSk *Summarizes Top Pathways from One of the Pathway Analyses*

Description

Summarizes top pathways from one of the pathway analyses (i.e., calculate.NTk, calculate.NEk, calculate.NGSk, or calculate.GSEA)

Usage

```
rankPathways.NGSk(res.NGSk, G, gsList, methodName = "NGSk",
                  npath = 25)
```

Arguments

res.NGSk	a list from the output of calculate.NGSk, calculate.NTk, calculate.NEk, or calculate.GSEA
G	a list containing the source, title, and probe sets associated with each curated pathway
gsList	a list containing three vectors from the output of the selectGeneSets function
methodName	a character vector of length 1 giving the name of the pathway analysis used in making res.NGSk
npath	an integer indicating the number of top gene sets to consider when ranking the top pathways

Details

This function ranks the statistics given in `res.NGSk` and summarizes the top gene sets in a tabular format similar to Table 2 in Tian et al. (2005)

Value

A data frame showing the pathways' indices in `G`, gene set category, pathway title, set size, `res.NGSk`'s statistics, the corresponding q-values, and the numerical ranks for the top gene sets.

Note

See the help page for `calculate.NGSk` for example code that uses `rankPathways.NGSk`

Author(s)

Lu Tian, Peter Park, and Weil Lai

References

Tian L., Greenberg S.A., Kong S.W., Altschuler J., Kohane I.S., Park P.J. (2005) Discovering statistically significant pathways in expression profiling studies. *Proceedings of the National Academy of Sciences of the USA*, **102**, 13544-9.

http://www.pnas.org/cgi/doi/10.1073/pnas.0506577102

`runSigPathway` *Perform pathway analysis*

Description

Performs pathway analysis

Usage

```
runSigPathway(G, minNPS = 20, maxNPS = 500,
              tab, phenotype, nsim = 1000,
              weightType = c("constant", "variable"), ngroups = 2,
              npath = 25, verbose = FALSE, allpathways = FALSE,
              annotpkg = NULL, alwaysUseRandomPerm = FALSE)
```

Arguments

`G`	a list containing the source, title, and probe sets associated with each curated pathway
`minNPS`	an integer specifying the minimum number of probe sets in `tab` that should be in a gene set
`maxNPS`	an integer specifying the maximum number of probe sets in `tab` that should be in a gene set

tab	a numeric matrix of expression values, with the rows and columns representing probe sets and sample arrays, respectively
phenotype	a numeric (or character if ngroups >= 2) vector indicating the phenotype
nsim	an integer indicating the number of permutations to use
weightType	a character string specifying the type of weight to use when calculating NEk statistics
ngroups	an integer indicating the number of groups in the matrix
npath	an integer indicating the number of top gene sets to consider from each statistic when ranking the top pathways
verbose	a boolean to indicate whether to print debugging messages to the R console
allpathways	a boolean to indicate whether to include the top npath pathways from each statistic or just consider the top npath pathways (sorted by the sum of ranks of both statistics) when generating the summary table
annotpkg	a character vector specifying the name of the BioConductor annotation package to use to fetch accession numbers, Entrez Gene IDs, gene name, and gene symbols
alwaysUseRandomPerm	a boolean to indicate whether the algorithm can use complete permutations for cases where nsim is greater than the total number of unique permutations possible with the phenotype vector

Details

runSigPathway is a wrapper function that

(1) Selects the gene sets to analyze using selectGeneSets

(2) Calculates NTk and NEk statistics using calculate.NTk and calculate.NEK

(3) Ranks the top npath pathways from each statistic using rankPathways

(4) Summarizes the means, standard deviation, and individual statistics of each probe set in each of the above pathways using getPathwayStatistics

Value

A list containing

gsList	a list containing three vectors from the output of the selectGeneSets function
list.NTk	a list from the output of calculate.NTk
list.NEk	a list from the output of calculate.NEk
df.pathways	a data frame from rankPathways which contains the top pathways' indices in G, gene set category, pathway title, set size, NTk statistics, NEk statistics, the corresponding q-values, and the ranks.
list.gPS	a list from getPathwayStatistics containing nrow(df.pathways) data frames corresponding to the pathways listed in df.pathways. Each data frame contains the name, mean, standard deviation, the test statistic (e.g., t-test), and the corresponding unadjusted p-value. If ngroups = 1, the Pearson correlation coefficient is also returned. If a valid annotpkg is specified, the probes' accession numbers, Entrez Gene IDs, gene name, and gene symbols are also returned.

parameters a list of parameters (e.g., nsim) used in the analysis

Author(s)

Lu Tian, Peter Park, and Weil Lai

References

Tian L., Greenberg S.A., Kong S.W., Altschuler J., Kohane I.S., Park P.J. (2005) Discovering statistically significant pathways in expression profiling studies. *Proceedings of the National Academy of Sciences of the USA*, **102**, 13544-9.

http://www.pnas.org/cgi/doi/10.1073/pnas.0506577102

Examples

```
## Load in filtered, expression data
data(MuscleExample)

## Prepare the pathways to analyze and run analysis with 1 wrapper function

nsim <- 1000
ngroups <- 2
verbose <- TRUE
weightType <- "constant"
npath <- 25
allpathways <- FALSE
annotpkg <- "hgu133a.db"

res.muscle <- runSigPathway(G, 20, 500, tab, phenotype, nsim,
                            weightType, ngroups, npath, verbose,
                            allpathways, annotpkg)

## Summarize results
print(res.muscle$df.pathways)

## Get more information about the probe sets means and other statistics
## for the top pathway in res.pathways
print(res.muscle$list.gPS[[1]])

## Write table of top-ranked pathways and their associated probe sets to
## HTML files
writeSigPathway(res.muscle, tempdir(), "sigPathway_rSP",
                "TopPathwaysTable.html")
```

---

selectGeneSets *Select gene sets to be analyzed in pathway analysis*

Description

Selects gene sets to be analyzed in pathway analysis based on minimum and maximum number of probe sets to consider per pathway.

Usage

```
selectGeneSets(G, probeID, minNPS = 20, maxNPS = 500)
```

Arguments

G	a list containing the source, title, and probe sets associated with each curated pathway
probeID	a character vector containing the names of probe sets associated with a matrix of expression values
minNPS	an integer specifying the minimum number of probe sets in probeID that should be in a gene set
maxNPS	an integer specifying the maximum number of probe sets in probeID that should be in a gene set

Details

This function selects the appropriate pathways from a large, curated list based on the minimum and maximum number of probe sets that should be considered in a gene set. It creates three vectors: nprobesV and indexV representing a sparse indicator matrix and indGused indicating which gene sets were selected from G.

Value

A list containing

nprobesV	an integer vector indicating the number of probe sets in probeID that is in each selected gene set
indexV	an integer vector containing positions for each 1s in the sparse indicator matrix
indGused	an integer vector indicating which pathways in G were chosen

Note

See the help page for calculate.NTk or calculate.NEk for example code that uses getPathwayStatistics

Author(s)

Lu Tian, Peter Park, and Weil Lai

References

Tian L., Greenberg S.A., Kong S.W., Altschuler J., Kohane I.S., Park P.J. (2005) Discovering statistically significant pathways in expression profiling studies. *Proceedings of the National Academy of Sciences of the USA*, **102**, 13544-9.

http://www.pnas.org/cgi/doi/10.1073/pnas.0506577102

writeSigPathway *Write results of pathway analysis to HTML format*

Description

Writes the table of top-ranked pathways and their associated probe set to HTML files.

Usage

```
writeSigPathway(spList, resDir = getwd(),
                outputDirName = "sigPathway_results",
                topIndexFileName = "TopPathwaysTable.html")
writeSP(rpDF, gpsList, parameterList = NULL, resDir = getwd(),
        outputDirName = "sigPathway_results",
        topIndexFileName = "TopPathwaysTable.html")
```

Arguments

spList	a list containing the output from the runSigPathway function
rpDF	a data frame of top-ranked pathways from rankPathways or rankPathways.NGSk
gpsList	a list containing data frames of probes represented in gene sets from getPathwayStatistics or getPathwayStatistics.NGSk
parameterList	a list containing the values of parameters used in the analysis
resDir	a character string specifying the file directory to write the results
outputDirName	a character string specifying the folder to write the results within resDir
topIndexFileName	a character string specifying the name for the HTML file containing the table of top-ranked pathways

Details

These functions export the results of the pathway analysis (e.g., runSigPathway) to several HTML files. The user can then quickly browse through the files for genes of interest within the top-ranked genes.

Value

None returned

Note

This function only uses the output of runSigPathway to generate the HTML files. Please see the help page of runSigPathway for example usage. The writeSP function should be used for those who have taken calculated the pathway statistics separately as shown in the help file of calculate.NTk, calculate.NEk, and calculate.NGSk

Author(s)

Weil Lai

Index

∗Topic **array**
calcTNullFast, 2
calcTStatFast, 3
calculate.GSEA, 4
calculate.NGSk, 5
calculatePathwayStatistics, 7
estimateNumPerm, 10
getPathwayStatistics, 11
getPathwayStatistics.NGSk, 12
rankPathways, 15
rankPathways.NGSk, 16
runSigPathway, 17
selectGeneSets, 19
writeSigPathway, 21
∗Topic **datagen**
importGeneSets, 13
∗Topic **datasets**
MuscleExample, 14
∗Topic **file**
importGeneSets, 13
∗Topic **htest**
calcTNullFast, 2
calcTStatFast, 3
calculate.GSEA, 4
calculate.NGSk, 5
calculatePathwayStatistics, 7
estimateNumPerm, 10
getPathwayStatistics, 11
getPathwayStatistics.NGSk, 12
rankPathways, 15
rankPathways.NGSk, 16
runSigPathway, 17
selectGeneSets, 19
writeSigPathway, 21

calcTNullFast, 2
calcTStatFast, 3
calculate.GSEA, 4
calculate.NEk
*(*calculatePathwayStatistics*)*, 7
calculate.NGSk, 5
calculate.NTk
*(*calculatePathwayStatistics*)*, 7
calculatePathwayStatistics, 7

estimateNumPerm, 10

G *(*MuscleExample*)*, 14
getPathwayStatistics, 11
getPathwayStatistics.NGSk, 12
gmtToG *(*importGeneSets*)*, 13
gmxToG *(*importGeneSets*)*, 13
grpToG *(*importGeneSets*)*, 13

importGeneSets, 13

MuscleExample, 14

phenotype *(*MuscleExample*)*, 14

rankPathways, 15
rankPathways.NGSk, 16
runSigPathway, 17

selectGeneSets, 19

tab *(*MuscleExample*)*, 14

writeSigPathway, 21
writeSP *(*writeSigPathway*)*, 21

xmlToG *(*importGeneSets*)*, 13