Package 'PBNPA'

December 20, 2016

Type Package

Title Permutation Based Non-Parametric Analysis of CRISPR Screen Data

Version 0.0.1

Author Gaoxiang Jia [aut, cre],
Xinlei Wang [aut],
Guanghua Xiao [aut]

Maintainer Gaoxiang Jia <GJia@SMU.edu>

Description Implements permutation based non-parametric analysis of CRISPR (Clustered Regularly Interspaced Short Palindromic Repeats) screen data.

Depends R (>= 2.15.0), metaRNASeq

License GPL (>= 2)

Encoding UTF-8

LazyData true

RoxygenNote 5.0.1

NeedsCompilation no

Repository CRAN

Date/Publication 2016-12-20 08:16:04

R topics documented:

PBNPA *Permutation Based Non-Parametric Analysis of CRISPR Screen Data*

Description

This function uses the raw read count data for CRISPR (Clustered Regularly Interspaced Short Palindromic Repeats) screens and conducts statistical analysis for permutation based non-parametric analysis of CRISPR screen data. This function can also be used to analyze data from other types of functional genomics screens such as siRNA screen or shRNA screen. Drug screens or microarray expression data, if have similar structure as this algorithm is designed for, can also be analyzed with this function as the algorithm has no specific distributional assumptions for the data and p-values are calculated from a permutation based procedure. It can handle data with multiple replicates.

Usage

```
PBNPA(dat, sim.no = 10, alpha.threshold = 0.2, fdr = 0.05)
```

Arguments

`dat`	List type with each element being the raw read count data for one replicate. Each element should be a dataframe with four columns. The first column is named 'sgRNA' which is the sgRNA index; the second column is named 'Gene' which is the gene index; the third column should be the initial read count or control read count and the fourth column should be the final read count or treatment read count.
`sim.no`	Number of permutations used to get the un-adjusted p-value.Set to 10 by default.
`alpha.threshold`	Threshold to remove genes with significant p-values. Set to 0.2 by default.
`fdr`	The FDR threshold to determine the selected genes. Set to 0.05 by default.

Details

PBNPA implements permutation based non-parametric analysis of CRISPR screen data. First, it uses the the median natural log fold change of sgRNAs target the same gene as the R score for that gene. Then it randomly assigns the read count pairs (initial and final) to each gene for T times to get a null distribution of the R score. Then it calculates a p-value for each gene based on the null distribution. To improve the accuracy of the p-value, it will remove the genes with p-value smaller than a threshold to remove the significant genes and permute again to get a better estimation of the null distribution. Then p-values for each gene are calculated from this improved null distribution. Then FDR is controlled by Benjamini-Hochberg procedure. If multiple replicates are included, p-values from each replicate are combined with Fisher's method. Details about this algorithm is in the publication to be published.

Value

A list of 5 elements will be returned. The first element is pos.gene, which is the index of genes identified as hits for positive screen by controlling FDR at the selected level; the second element is pos.number, which is the number of genes identified as hits for positive screen; The third element is neg.gene, which is the index of genes identified as hits for negative screen by controlling FDR at the selected level; the fourth element is neg.number, which is the number of genes identified as hits for negative screen; the fifth element is a dataframe which contains unadjusted p-values and FDR adjusted p-values for all the genes (for both negative selection and positive selection).

Examples

```
dat11 = system.file('extdata','simdata_20per_off50.csv', package='PBNPA')
dat22 = system.file('extdata','simdata_20per_off49.csv', package='PBNPA')
dat33 = system.file('extdata','simdata_20per_off48.csv', package='PBNPA')
dat1 = read.csv(dat11, header = TRUE)
dat2 = read.csv(dat22, header = TRUE)
dat3 = read.csv(dat33, header = TRUE)
datlist = list(dat1, dat2, dat3)
result = PBNPA(datlist)
```

Index