Package ‘ICDS’

July 15, 2021

Type Package
Title Identification of Cancer Dysfunctional Subpathway with Omics Data
Version 0.1.2
Maintainer Junwei Han <hanjunwei1981@163.com>
Description Identify Cancer Dysfunctional Sub-pathway by integrating gene expression, DNA methylation and copy number variation, and pathway topological information. 1) We firstly calculate the gene risk scores by integrating three kinds of data: DNA methylation, copy number variation, and gene expression. 2) Secondly, we perform a greedy search algorithm to identify the key dysfunctional sub-pathways within the pathways for which the discriminative scores were locally maximal. 3) Finally, the permutation test was used to calculate statistical significance level for these key dysfunctional sub-pathways.

Depends R (>= 2.10)
biocViews
Imports igraph, graphite, metap, methods, org.Hs.eg.db
Suggests knitr, rmarkdown, prettydoc
License GPL (>= 2)
Encoding UTF-8
LazyData true
RoxygenNote 7.1.1
VignetteBuilder knitr
NeedsCompilation no
Author Junwei Han [cre], Baotong Zheng [aut], Siyao Liu [ctb]
Repository CRAN
Date/Publication 2021-07-15 11:30:10 UTC
### ICDS-package

*Identification of Cancer Dysfunctional Subpathway by integrating DNA methylation, copy number variation, and gene expression data*

#### Description

Identify Cancer Dysfunctional Subpathway by integrating gene expression, DNA methylation and copy number variation, and pathway topological information. 1) We firstly calculate the gene risk scores by integrating three kinds of data: DNA methylation, copy number variation, and gene expression. 2) Secondly, we perform a greedy search algorithm to identify the key dysfunctional subpathways within the pathways for which the discriminative scores were locally maximal. 3) Finally, the permutation test was used to calculate statistical significance level for these key dysfunctional subpathways.

#### combinep_three

The function `combinep_three` combines three kinds of p-values, then calculates z-score for them.

**Usage**

```r
combinep_three(p1, p2, p3)
```
**Arguments**

- **p1**  
  the p-values or corrected p-values

- **p2**  
  the p-values or corrected p-values

- **p3**  
  the p-values or corrected p-values

**Value**

A numeric vector of z_scores

**Examples**

```r
exp.p<-GetExampleData("exp.p")
meth.p<-GetExampleData("meth.p")
cnv.p<-GetExampleData("cnv.p")
combinep_three(exp.p,meth.p,cnv.p)
```

```r
combinep_two
```

---

**Description**

'combinep_two' combine two kinds of p-values, then calculate z-score for them.

**Usage**

```
combinep_two(p1, p2)
```

**Arguments**

- **p1**  
  A numeric vector of p-values or corrected p-values

- **p2**  
  A numeric vector of p-values or corrected p-values

**Value**

A numeric vector of z_scores

**Examples**

```r
exp.p<-GetExampleData("exp.p")
meth.p<-GetExampleData("meth.p")
combinep_two(exp.p,meth.p)
```
coverp2zscore

**Description**

'coverp2zscore' calculate z-scores for p-values

**Usage**

coverp2zscore(pdata)

**Arguments**

pdata A numeric vector of p-values or corrected p-values

**Value**

A numeric vector of z_scores

**Examples**

```r
exp.p<-GetExampleData("exp.p")
meth.p<-GetExampleData("meth.p")
cnv.p<-GetExampleData("cnv.p")
coverp2zscore(exp.p)
coverp2zscore(meth.p)
coverp2zscore(cnv.p)
```

**envData**

The variables in the environment include an example expression profile, an methylation profile, an copy number variation data, amplified genes, deleted genes, A numeric vector of z_scores, p-values, A vector of 0/1s, indicating the class of samples, interested subpathways, Optimized subpathway, and the statistical significance p value and FDR for these optimal subpathways

**Description**

Identify Cancer Dysfunctional Subpathway by integrating gene expression, DNA methylation and copy number variation, and pathway topological information.

1) We firstly calculate the gene risk scores by integrating three kinds of data: DNA methylation, copy number variation, and gene expression.

2) Secondly, we perform a greedy search algorithm to identify the key dysfunctional subpathways within the pathways for which the discriminative scores were locally maximal.

3) Finally, the permutation test was used to calculate statistical significance level for these key dysfunctional subpathways.
FindSubPath

Format

An environment variable

Details

The environment variable includes the variable exp_data, meth_data, cnv_data, amp_gene, del_gene, zzz, exp.p, meth.p, cnv.p.

Author(s)

Junwei Han<hanjunwei1981@163.com>, Baotong Zheng<btzheng1116@163.com>, Siyao Liu<liusiyao29@163.com>

Description

‘FindSubPath’ uses a greedy search algorithm to search for key subpathways in each entire pathway.

Usage

FindSubPath(
  zz,
  Pathway = "kegg",
  delta = 0.05,
  seed_p = 0.05,
  min.size = 5,
  out.F = FALSE,
  out.file = "Subpath.txt"
)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>zz</td>
<td>A numeric vector of z.scores.</td>
</tr>
<tr>
<td>Pathway</td>
<td>The name of the pathway database.</td>
</tr>
<tr>
<td>delta</td>
<td>Diffusion coefficient in each step of searching subpath.</td>
</tr>
<tr>
<td>seed_p</td>
<td>Define gene whose p-value smaller than seed_p as seed gene.</td>
</tr>
<tr>
<td>min.size</td>
<td>The smallest size of subpathways.</td>
</tr>
<tr>
<td>out.F</td>
<td>Logical, tell if output subpathways.</td>
</tr>
<tr>
<td>out.file</td>
<td>file name of subpathways.</td>
</tr>
</tbody>
</table>

Value

Key dysfunctional subpathways in each pathway, in which the risk score of the genes were significantly higher.
Example

```r
require(graphite)
zz<-GetExampleData("zzz")
k<-FindSubPath(zz)
```

### Description

`getCnvp` perform t-test on copy number variation data

### Usage

```r
getCnvp(
  exp_data,
  cnv_data,
  amp_gene,
  del_gene,
  p.adjust = TRUE,
  method = "fdr"
)
```

### Arguments

- `exp_data`: A data frame
- `cnv_data`: Copy number variation data
- `amp_gene`: A vector of strings, the IDs of amplified genes.
- `del_gene`: A vector of strings, the IDs of deleted genes.
- `p.adjust`: Logical, tell if returns corrected p-values
- `method`: Correction method, which can be one of "holm", "hochberg", "hommel", "bonferroni", "BH", "BY".

### Details

- `cnv_data` is TCGA level4 data. if `p.adjust=TRUE`, return corrected p-values, if `p.adjust=FALSE`, return p-values

### Value

A numeric vector of p-values or corrected p-values
GetExampleData

Examples

```r
exp_data<-GetExampleData("exp_data")
meth_data<-GetExampleData("meth_data")
cnv_data<-GetExampleData("cnv_data")
amp_gene<-GetExampleData("amp_gene")
del_gene<-GetExampleData("del_gene")
getCnvp(exp_data,cnv_data,amp_gene,del_gene,p.adjust=FALSE,method="fdr")
```

---

GetExampleData  Get the example data

Description

Get the example data of test package for little trials.

Usage

`GetExampleData(exampleData)`

Arguments

- `exampleData`: A character, should be one of "exp_data", "meth_data", "cnv_data", "amp_gene", "del_gene", "label1", "label2", "zz", "exp.p", "meth.p", "cnv.p" and "pathdata".

Details

The function `getExampleData(ExampleData = "exp.p")` obtains a vector of IncRNAs confirmed to be related with breast cancer. The function `getExampleData(ExampleData = "Profile")` obtains the expression pr

References

Description

'getExpp' perform t-test on Expression profile data

Usage

getExpp(exp_data, label, p.adjust = TRUE, method = "fdr")

Arguments

exp_data  A data frame, the expression profile to calculate p-value for each gene, the row-names should be the symbol of genes.
label     A vector of 0/1s, indicating the class of samples in the expression profile, 0 represents case, 1 represents control.
p.adjust  Logical, tell if returns corrected p-values
method    Correction method, which can be one of "holm", "hochberg", "hommel", "bonferroni", "BH", "BY".

Details

For a given expression profile of two conditions, ICDS package provide t-test method to calculate p-values or corrected p-values (if p.adjust=TRUE, return corrected p-values, if p.adjust=FALSE, return p-values) for each genes. The row of the expression profile should be gene symbols and the column of the expression profile should be names of samples. Samples should be under two conditions and the label should be given as 0 and 1.

Value

A numeric vector of p-values or corrected p-values

Examples

profile<-GetExampleData("exp_data")
label<-GetExampleData("label1")
getExpp(profile,label,p.adjust=FALSE)
Description

‘getMethp’ perform t-test on Methylation profile data

Usage

getMethp(meth_data, label, p.adjust = TRUE, method = "fdr")

Arguments

- **meth_data**: A data frame, the Methylation profile to calculate p-value for each gene, the rownames should be the symbol of genes.
- **label**: label A vector of 0/1s, indicating the class of samples in the Methylation profile, 0 represents case, 1 represents control.
- **p.adjust**: Logical, tell if returns corrected p-values
- **method**: Correction method, which can be one of "holm", "hochberg", "hommel", "bonferroni", "BH", "BY".

Details

For a given Methylation profile of two conditions, ICDS package provide t-test method to calculate p-values or corrected p-values (if p.adjust=TRUE, return corrected p-values, if p.adjust=FALSE, return p-values.) for each genes. The row of the Methylation profile should be gene symbols and the column of the Methylation profile should be names of samples. Samples should be under two conditions and the label should be given as 0 and 1.

Value

A numeric vector of p-values or corrected p-values

Examples

```r
profile<-GetExampleData("meth_data")
label<-GetExampleData("label2")
getMethp(profile, label, p.adjust=FALSE)
```
Description

'opt_subpath' Optimize interested subpathways. If the number of genes shared by the two pathways accounted for more than the Overlap ratio of each pathway genes, then combine two pathways.

Usage

opt_subpath(subpathdata, zz, overlap = 0.6)

Arguments

subpathdata interested subpathways
zz a vector of z-scores
overlap Overlap ratio of each two pathway genes

Value

Optimized subpathway: the number of genes shared by any two pathways accounted for less than the Overlap ratio of each pathway genes.

Examples

zz<-GetExampleData("zzz")
subpathdata<-GetExampleData("subpathdata")
optsubpath<-opt_subpath(subpathdata,zz,overlap=0.6)

Description

The permutation test method 1 and method 2 were used to calculate the statistical significance level for these optimal subpathways.

Usage

Permutation(
  subpathwayz,
  zz,
  nperm1 = 1000,
  method1 = TRUE,
  nperm2 = 1000,
  method2 = FALSE
)
**PlotSubpathway**

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>subpathwayz</td>
<td>Optimize interested subpathways</td>
</tr>
<tr>
<td>zz</td>
<td>a vector of z-scores</td>
</tr>
<tr>
<td>nperm1</td>
<td>times of permutation to perform use method1</td>
</tr>
<tr>
<td>method1</td>
<td>permutation analysis method1</td>
</tr>
<tr>
<td>nperm2</td>
<td>times of permutation to perform use method2</td>
</tr>
<tr>
<td>method2</td>
<td>permutation analysis method2</td>
</tr>
</tbody>
</table>

**Value**

the statistical significance p value and FDR for these optimal subpathways

**Examples**

```r
require(graphite)
keysubpathways<-GetExampleData("keysubpathways")
zzz<-GetExampleData("zzz")
Permutation(keysubpathways,zzz,nperm1=10,method1=TRUE,nperm2=10,method2=FALSE)
```

**Description**

PlotSubpathway: plot a network graph when user input a list of gene

**Usage**

```r
PlotSubpathway(
  subpID,
  pathway.name,
  zz,
  Pathway = "kegg",
  layout = layout.fruchterman.reingold
)
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>subpID</td>
<td>gene list of a interested subpathway</td>
</tr>
<tr>
<td>pathway.name</td>
<td>name of the interested subpathway</td>
</tr>
<tr>
<td>zz</td>
<td>z-score of each gene</td>
</tr>
<tr>
<td>Pathway</td>
<td>the name of the pathway database</td>
</tr>
<tr>
<td>layout</td>
<td>The layout specification(\texttt{layout}). It must be a call to a layout specification function.</td>
</tr>
</tbody>
</table>
Value
Network graph

Examples

```r
require(graphite)

subpID<-unlist(strsplit("ACSS1/ALDH3B2/ADH1B/ADH1A/ALDH2/DLAT/ACSS2","/"))
pathway.name="Glycolysis / Gluconeogenesis"
zzz<- GetExampleData("zzz")
PlotSubpathway(subpID=subpID,pathway.name=pathway.name,zz=zzz)
```
## Index

* data
  - envData, 4
  - amp_gene (envData), 4
  - cnv.p (envData), 4
  - cnv_data (envData), 4
  - combinep_three, 2
  - combinep_two, 3
  - coverp2zscore, 4
  - del_gene (envData), 4
  - envData, 4
  - exp.p (envData), 4
  - exp_data (envData), 4
  - FindSubPath, 5
  - getCnvp, 6
  - GetExampleData, 7
  - getExpp, 8
  - getMethp, 9
  - ICDS (ICDS-package), 2
  - ICDS-package, 2
  - label1 (envData), 4
  - label2 (envData), 4
  - layout_, 11
  - meth.p (envData), 4
  - meth_data (envData), 4
  - opt_subpath, 10
  - opt_subpathways (envData), 4
  - Permutation, 10
  - PlotSubpathway, 11
  - subpathdata (envData), 4
  - zzz (envData), 4