Package ‘GeoTcgaData’

October 24, 2021

**Type** Package

**Title** Processing various types of data on GEO and TCGA

**Version** 1.0.2

**Description** Gene Expression Omnibus (GEO) and The Cancer Genome Atlas (TCGA) provide us with a wealth of data, such as RNA-seq, DNA Methylation, SNP and Copy number variation data. It's easy to download data from TCGA using the gdc tool, but processing these data into a format suitable for bioinformatics analysis requires more work. This R package was developed to handle these data.

**Depends** R (>= 3.6.0)

**License** Artistic-2.0

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.1.1

**Suggests** knitr, rmarkdown, DESeq2, S4Vectors, ChAMP, impute, tidyr, clusterProfiler, org.Hs.eg.db, edgeR, limma, quantreg

**VignetteBuilder** knitr

**Imports** utils, data.table, magrittr, plyr, cqn

**Language** en-US

**NeedsCompilation** no

**Author** Erqiang Hu [aut, cre]

**Maintainer** Erqiang Hu <13766876214@163.com>

**Repository** CRAN

**Date/Publication** 2021-10-24 09:20:05 UTC

**R topics documented:**

```
cal_mean_module .................................................. 2
classify_sample ................................................... 3
countToFpkm_matrix ............................................... 3
countToTpm_matrix ............................................... 4```
### cal_mean_module

Find the mean value of the gene in each module

#### Description

Find the mean value of the gene in each module

#### Usage

```r
cal_mean_module(geneExpress, module)
```

#### Arguments

- `geneExpress`: a data.frame
- `module`: a data.frame

#### Value

A data.frame, means the mean of gene expression value in the same module
**classify_sample**

*Get the differentially expressed genes using DESeq2 package*

**Description**

Get the differentially expressed genes using DESeq2 package

**Usage**

```r
classify_sample(profile_input)
```

**Arguments**

- `profile_input`: a data.frame

**Value**

a data.frame, an intermediate result of DESeq2

**Examples**

```r
profile2 <- classify_sample(kegg_liver)
```

---

**countToFpkm_matrix**

*Convert count to FPKM*

**Description**

Convert count to FPKM

**Usage**

```r
countToFpkm_matrix(counts_matrix)
```

**Arguments**

- `counts_matrix`: a matrix, colnames of `counts_matrix` are sample name, rownames of `counts_matrix` are gene symbols

**Value**

a matrix
Examples

lung_squ_count2 <- matrix(c(1,2,3,4,5,6,7,8,9),ncol=3)
rownames(lung_squ_count2) <- c("DISC1","TCOF1","SPPL3")
colnames(lung_squ_count2) <- c("sample1","sample2","sample3")
jieguo <- countToFpkm_matrix(lung_squ_count2)


countToTpm_matrix  

Convert count to Tpm

Description

Convert count to Tpm

Usage

countToTpm_matrix(counts_matrix)

Arguments

counts_matrix  a matrix, colnames of counts_matrix are sample name, rownames of counts_matrix are gene symbols

Value

a matrix

Examples

l lung_squ_count2 <- matrix(c(1,2,3,4,5,6,7,8,9),ncol=3)
rownames(lung_squ_count2) <- c("DISC1","TCOF1","SPPL3")
colnames(lung_squ_count2) <- c("sample1","sample2","sample3")
jieguo <- countToTpm_matrix(lung_squ_count2)

differential_cv n  

Do chi-square test to find differential genes

Description

Do chi-square test to find differential genes

Usage

differential_cvn(rt)

Arguments

rt  result of prepare_chi()
**diff_CNV**

**Value**

a matrix

**Examples**

```r
jieguo3 <- matrix(c(-1.09150,-1.47120,-0.87050,-0.50880,
                    -0.50880,2.0,2.0,2.0,2.0,2.0,2.0,2.0,2.0,2.0,2.0,2.0,2.0,2.0,
                    2.621332,2.621332,2.0,2.0,2.0,2.0,2.0,2.0,2.0,2.0,2.0,2.0,2.0,2.0),nrow=5)
rownames(jieguo3) <- c("AJAP1","FHAD1","CLCNKB","CROCCP2","AL137798.3")
rt <- prepare_chi(jieguo3)
chiResult <- differential_cnv(rt)
```

---

**diff_CNV**

*Do difference analysis of gene level copy number variation data*

**Description**

Do difference analysis of gene level copy number variation data

**Usage**

```r
diff_CNV(cnvData, sampleGroup)
```

**Arguments**

- **cnvData**: data.frame of CNV data
- **sampleGroup**: vector of sample group

**Examples**

```r
## Not run:
library(TCGAbiolinks)
query <- GDCquery(project = "TCGA-LGG",
data.category = "Copy Number Variation",
data.type = "Gene Level Copy Number Scores")
GDCdownload(query, method = "api", files.per.chunk = 5, directory = Your_Path)
data <- GDCprepare(query = query,save = TRUE,directory = "Your_Path")
class(data) <- "data.frame"
cnvData <- data[, -c(1,2,3)]
rownames(cnvData) <- data[, 1]
sampleGroup = sample(c("A","B"), ncol(cnvData), replace = TRUE)
```
diff_Cnv <- diff_CNV(cnvData, sampleGroup)

## End(Not run)

---

### diff_gene

*Get the differentially expressioned genes using DESeq2 package*

**Description**

Get the differentially expressioned genes using DESeq2 package

**Usage**

```r
diff_gene(profile2_input)
```

**Arguments**

- `profile2_input` - a result of classify_sample

**Value**

a matrix, information of differential expression genes

**Examples**

```r
profile2 <- classify_sample(kegg_liver)
jieguo <- diff_gene(profile2)
```

---

### diff_RNA

*Do difference analysis of RNA-seq data*

**Description**

Do difference analysis of RNA-seq data

**Usage**

```r
diff_RNA(counts, group, method = "limma", geneLength = NULL, gccontent = NULL)
```

**Arguments**

- `counts` - a dataframe or numeric matrix of raw counts data
- `group` - sample groups
- `method` - one of "DESeq2", "edgeR" and "limma".
- `geneLength` - a vector of gene length.
- `gccontent` - a vector of gene GC content.
Examples

```r
## Not run:
library(TCGAbiolinks)
query <- GDCquery(project = "TCGA-ACC",
data.category = "Transcriptome Profiling",
data.type = "Gene Expression Quantification",
workflow.type = "HTSeq - Counts")
GDCdownload(query, method = "api", files.per.chunk = 3,
directory = Your_Path)
dataRNA <- GDCprepare(query = query, directory = Your_Path,
save = TRUE, save.filename = "dataRNA.RData")
## get raw count matrix
dataPrep <- TCGAanalyze_Preprocessing(object = dataRNA,
cor.cut = 0.6,
datatype = "HTSeq - Counts")

# Use `diff_RNA` to do difference analysis.
# We provide the data of human gene length and GC content in `gene_cov`.
group <- sample(c("grp1", "grp2"), ncol(dataPrep), replace = TRUE)
library(cqn) # To avoid reporting errors: there is no function "rq"
## get gene length and GC content
library(org.Hs.eg.db)
genes_bitr <- bitr(rownames(gene_cov), fromType = "ENTREZID", toType = "ENSEMBL",
OrgDb = org.Hs.eg.db, drop = TRUE)
genes_bitr <- genes_bitr[!duplicated(genes_bitr[,2]), ]
gene_cov2 <- gene_cov[gene_bitr$ENTREZID, ]
rownames(gene_cov2) <- genes_bitr$ENSEMBL
genes <- intersect(rownames(dataPrep), rownames(gene_cov))
dataPrep <- dataPrep[genes, ]
geneLength <- gene_cov2(genes, "length")
gccontent <- gene_cov2(genes, "GC")
names(geneLength) <- names(gccontent) <- genes
## Difference analysis
DEGAll <- diff_RNA(counts = dataPrep, group = group,
geneLength = geneLength, gccontent = gccontent)
# Use `clusterProfiler` to do enrichment analytics:
diffGenes <- DEGAll$logFC
names(diffGenes) <- rownames(DEGAll)
diffGenes <- sort(diffGenes, decreasing = TRUE)
library(clusterProfiler)
library(enrichplot)
library(org.Hs.eg.db)
gsego <- gseGO(gene = diffGenes, OrgDb = org.Hs.eg.db, keyType = "ENSEMBL")
dotplot(gsego)
## End(Not run)
```
diff_SNP  

Do difference analysis of SNP data

**Description**
Do difference analysis of SNP data

**Usage**
```
diff_SNP(snpDf, sampleGroup, method = min)
```

**Arguments**
- `snpDf`  
data.frame of SNP data.
- `sampleGroup`  
vector of sample group.
- `method`  
Method of combining the pvalue of multiple snp in a gene.

---

diff_SNP_tcga  

Do difference analysis of SNP data downloaded from TCGAbiolinks

**Description**
Do difference analysis of SNP data downloaded from TCGAbiolinks

**Usage**
```
diff_SNP_tcga(snpData, sampleType)
```

**Arguments**
- `snpData`  
data.frame of SNP data downloaded from TCGAbiolinks
- `sampleType`  
vector of sample group

**Examples**
```r
## Not run:
library(TCGAbiolinks)
query <- GDCquery(project = "TCGA-ACC",
data.category = "Simple Nucleotide Variation",
data.type = "Masked Somatic Mutation",
workflow.type = "MuSE Variant Aggregation and Masking")

GDCdownload(query, method = "api", files.per.chunk = 5, directory = Your_Path)
data_snp <- GDCprepare(query = query,
    save = TRUE,
```
directory = "Your_Path")
samples <- unique(data_snp$Tumor_Sample_Barcode)
sampleType <- sample(c("A","B"), length(samples), replace = TRUE)
names(sampleType) <- samples
pvalue <- diff_SNP_tcga(snpData = data_snp, sampleType = sampleType)

## End(Not run)

description  
Convert fpkm to Tpm

Usage
fpkmToTpm_matrix(fpkm_matrix)

Arguments

fpkm_matrix  
a matrix, colnames of fpkm_matrix are sample name, rownames of fpkm_matrix are genes

Value

a matrix

Examples

lung_squ_count2 <- matrix(c(0.11,0.22,0.43,0.14,0.875,0.66,0.77,0.18,0.29),ncol=3)
rownames(lung_squ_count2) <- c("DISC1","TCOF1","SPPL3")
colnames(lung_squ_count2) <- c("sample1","sample2","sample3")
result <- fpkmToTpm_matrix(lung_squ_count2)

description  
a data.frame of gene expression data

Usage
geneExpress

geneExpress
Format

A data.frame with 10779 rows and 2 columns

Details

the columns are gene expression values

gene_ave

Average the values of same genes in gene expression profile

Description

Average the values of same genes in gene expression profile

Usage

gene_ave(file_gene_ave, k = 1)

Arguments

file_gene_ave a data.frame
k a number

Value

a data.frame, the values of same genes in gene expression profile

Examples

aa <- c("MARCH1", "MARC1", "MARCH1", "MARCH1", "MARCH1")
bb <- c(2.969058399, 4.722410064, 8.165514853, 8.24243893, 8.60815086)
cc <- c(3.969058399, 5.722410064, 7.165514853, 6.24243893, 7.60815086)
file_gene_ave <- data.frame(aa=aa, bb=bb, cc=cc)
colnames(file_gene_ave) <- c("Gene", "GSM1629982", "GSM1629983")
result <- gene_ave(file_gene_ave, 1)
gene_cov

---

gene_cov    a data.frame of gene length and GC content

Description

a data.frame of gene length and GC content

Usage

gene_cov

Format

A data.frame with 27341 rows and 2 column

---

GSE66705_sample2    a matrix of gene expression data in GEO

Description

the first column represents the gene symbol

Usage

GSE66705_sample2

Format

A matrix with 999 rows and 3 column

Details

the other columns represent the expression of genes
id_conversion

<table>
<thead>
<tr>
<th>id_ava</th>
<th>Gene id conversion types</th>
</tr>
</thead>
</table>

**Description**

Gene id conversion types

**Usage**

`id_ava()`

**Value**

a vector

**Examples**

`id_ava()`

---

id_conversion

*Convert ENSEMBL gene id to gene Symbol in TCGA*

**Description**

Convert ENSEMBL gene id to gene Symbol in TCGA

**Usage**

`id_conversion(profiles, toType = "SYMBOL")`

**Arguments**

- `profiles`: a data.frame
- `toType`: one of `keytypes(org.Hs.eg.db)`

**Value**

a data.frame, gene symbols and their expression value

**Examples**

```r
## Not run:
library(org.Hs.eg.db)
profile <- GeoTcgaData::profile
result <- id_conversion(profile)

## End(Not run)
```
**id_conversion_vector**  
*Gene id conversion*

---

**Description**

Gene id conversion

**Usage**

```r
id_conversion_vector(from, to, IDs, na.rm = FALSE)
```

**Arguments**

- `from` : one of `id_ava()`  
- `to` : one of `id_ava()`  
- `IDs` : the gene id which needed to convert  
- `na.rm` : Whether to remove lines containing NA

**Value**

a vector of genes

**Examples**

```r
id_conversion_vector("symbol", "Ensembl_ID",  
c("A2ML1", "A2ML1-AS1", "A4GALT", "A12M1", "AAAS")
```

---

**kegg_liver**  
*a matrix of gene expression data in TCGA*

---

**Description**

the first column represents the gene symbol

**Usage**

```r
kegg_liver
```

**Format**

A matrix with 100 rows and 150 column

**Details**

the other columns represent the expression(count) of genes
**Merge_methy_tcga**  
*Merge methylation data downloaded from TCGA*

**Description**  
Merge methylation data downloaded from TCGA

**Usage**  
`Merge_methy_tcga(dirr = NULL)`

**Arguments**  
- `dirr`  
  a string for the directory of methylation data download from tcga using the tools gdc

**Value**  
a matrix, a combined methylation expression spectrum matrix

**Examples**  
`merge_result <- Merge_methy_tcga(system.file(file.path("extdata","methy"),package="GeoTcgaData"))`

---

**methyDiff**  
*Get methylation difference gene*

**Description**  
Get methylation difference gene

**Usage**  
`methyDiff(cpgData, sampleGroup, combineMethod = RobustRankAggreg::rhoScores)`

**Arguments**  
- `cpgData`  
  data.frame of cpg beta value
- `sampleGroup`  
  vector of sample group
- `combineMethod`  
  method to combine the cpg pvalues
module

| module | a matrix of module name, gene symbols, and the number of gene symbols |

Description

A matrix of module name, gene symbols, and the number of gene symbols

Usage

module

Format

A matrix with 176 rows and 3 columns

prepare_chi

Preparer file for chi-square test

Description

Preparer file for chi-square test

Usage

prepare_chi(cnv)

Arguments

cnv: result of ann_merge()

Value

a matrix

Examples

cnv <- matrix(c(-1.09150,-1.47120,-0.87050,-0.50880,
    -0.50880,2.0,2.0,2.0,2.0,2.0,2.601962,2.621332,2.621332,
    2.621332,2.621332,2.0,2.0,2.0,2.0,2.0,2.0,2.0,2.0,2.0,2.0),nrow=5)
cnv <- as.data.frame(cnv)
rownames(cnv) <- c("AJAP1","FHAD1","CLCNKB","CROCCP2","AL137798.3")
cnv_chi_file <- prepare_chi(cnv)
Description

the first column represents the gene symbol

Usage

profile

Format

A matrix with 10 rows and 10 column

Details

the other columns represent the expression (FPKM) of genes

rep1

Handle the case where one id corresponds to multiple genes

Description

Handle the case where one id corresponds to multiple genes

Usage

rep1(input_file, string)

Arguments

input_file: input file, a data.frame or a matrix
string: a string, sep of the gene

Value

a data.frame, when an id corresponds to multiple genes, the expression value is assigned to each gene

Examples

aa <- c("MARCH1 /// MMA","MARC1","MARCH2 /// MARCH3","MARCH3 /// MARCH4","MARCH1")
bb <- c("2.969058399","4.722410064","8.165514853","8.24243893","8.60815086")
cc <- c("3.969058399","5.722410064","7.165514853","6.24243893","7.60815086")
input_file <- data.frame(aa=aa,bb=bb,cc=cc)
rep1_result <- rep1(input_file," /// ")
**Description**

Handle the case where one id corresponds to multiple genes

**Usage**

`rep2(input_file, string)`

**Arguments**

- `input_file`: input file, a data.frame or a matrix
- `string`: a string, sep of the gene

**Value**

A data.frame, when an id corresponds to multiple genes, the expression value is deleted

**Examples**

```r
aa <- c("MARCH1 /// MMA","MARC1","MARCH2 /// MARCH3","MARCH3 /// MARCH4","MARCH1")
bb <- c("2.969058399","4.722410064","8.165514853","8.24243893","8.60815086")
cc <- c("3.969058399","5.722410064","7.165514853","6.24243893","7.60815086")
input_file <- data.frame(aa=aa,bb=bb,cc=cc)
rep2_result <- rep2(input_file," /// ")
```

**Description**

Combine clinical information obtained from TCGA and extract survival data

**Usage**

`tcga_cli_deal(Files_dir = "your_clinical_directory")`

**Arguments**

- `Files_dir`: a dir data

**Value**

A matrix, survival time and survival state in TCGA
Examples

tcga_cli_deal(system.file(file.path("extdata","tcga_cli"),package="GeoTcgaData"))

| ventricle | a matrix of gene expression data in GEO |

Description

the first column represents the gene symbol

Usage

ventricle

Format

A matrix with 32 rows and 20 column

Details

the other columns represent the expression of genes
Index

* datasets
  * gene_cov, 11
  * geneExpress, 9
  * GSE66705_sample2, 11
  * kegg_liver, 13
  * module, 15
  * profile, 16
  * ventricle, 18

  cal_mean_module, 2
  classify_sample, 3
  countToFpm_matrix, 3
  countToTpm_matrix, 4

  diff_CNV, 5
  diff_gene, 6
  diff_RNA, 6
  diff_SNP, 8
  diff_SNP_tcga, 8
  differential_cnv, 4

  fpkmToTpm_matrix, 9
  gene_ave, 10
  gene_cov, 11
  geneExpress, 9
  GSE66705_sample2, 11

  id_ava, 12
  id_conversion, 12
  id_conversion_vector, 13

  kegg_liver, 13
  Merge_methy_tcga, 14
  methyDiff, 14
  module, 15

  prepare_chi, 15
  profile, 16

  rep1, 16
  rep2, 17
  tcga_cli_deal, 17
  ventricle, 18