Package ‘GeoTcgaData’

June 9, 2020

Type Package

Title Processing various types of data on GEO and TCGA

Version 0.2.4

Description Gene Expression Omnibus(GEO) and The Cancer Genome Atlas (TCGA) provide us with a wealth of data, such as RNA-seq, DNA Methylation, 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 7.1.0

Suggests knitr, rmarkdown, DESeq2, S4Vectors

VignetteBuilder knitr

Imports utils, data.table

Language en-US

NeedsCompilation no

Author Erqiang Hu [aut, cre]

Maintainer Erqiang Hu <13766876214@163.com>

Repository CRAN

Date/Publication 2020-06-09 12:20:06 UTC

R topics documented:

`ann_merge` *Merge the copy number variation data downloaded from TCGA using gdc*

Description

Merge the copy number variation data downloaded from TCGA using gdc

Usage

```
ann_merge(dirr, metadatafile)
```

Arguments

`dirr`	a string of direction, catalogue of copy number variation data
`metadatafile`	a metadata file download from TCGA

Value

a matrix,each column is a sample, each row is a gene

Examples

```
metadatafile_name <- "metadata.cart.2018-11-09.json"
## Not run: jieguo2 <- ann_merge(dirr = system.file(file.path("extdata","cnv"),
package="GeoTcgaData"),metadatafile=metadatafile_name)
## End(Not run)
```

`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

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

Arguments

`geneExpress`	a data.frame
`module`	a data.frame

Value

a matrix, means the mean of gene expression value in the same module

Examples

```
result <- cal_mean_module(geneExpress,module)
```

`classify_sample` *Get the differentially expressioned genes using DESeq2 package*

Description

Get the differentially expressioned genes using DESeq2 package

Usage

```
classify_sample(profile_input)
```

Arguments

`profile_input`	a data.frame

Value

a data.frame, a intermediate results of DESeq2

Examples

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

countToFpkm_matrix *Convert count to FPKM*

Description

Convert count to FPKM

Usage

```
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

```
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_cnv` *Do chi-square test to find differential genes*

Description

Do chi-square test to find differential genes

Usage

```
differential_cnv(rt)
```

Arguments

`rt` result of prepare_chi()

Value

a matrix

Examples

```
jieguo3 <- 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,
                    2.0,2.0,2.0,2.0,2.0,2.0,2.0),nrow=5)
rownames(jieguo3) <- c("AJAP1","FHAD1","CLCNKB","CROCCP2","AL137798.3")
colnames(jieguo3) <- c("TCGA-DD-A4NS-10A-01D-A30U-01","TCGA-ED-A82E-01A-11D-A34Y-01",
"TCGA-WQ-A9G7-01A-11D-A36W-01","TCGA-DD-AADN-01A-11D-A40Q-01",
"TCGA-ZS-A9CD-10A-01D-A36Z-01","TCGA-DD-A1EB-11A-11D-A12Y-01")
rt <- prepare_chi(jieguo3)
chiResult <- differential_cnv(rt)
```

diff_gene *Get the differentially expressioned genes using DESeq2 package*

Description

Get the differentially expressioned genes using DESeq2 package

Usage

```
diff_gene(profile2_input)
```

Arguments

`profile2_input`	a result of classify_sample

Value

a matrix, information of differential expression genes

Examples

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

fpkmToTpm_matrix *Convert fpkm to Tpm*

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")
jieguo <- fpkmToTpm_matrix(lung_squ_count2)
```

`geneExpress` *a data.frame of gene expression data*

Description

the first column is a vector of gene symbols

Usage

```
geneExpress
```

Format

A data.frame with 10779 rows and 3 column

Details

the other 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("Gene Symbol","MARCH1","MARC1","MARCH1","MARCH1","MARCH1")
bb <- c("GSM1629982","2.969058399","4.722410064","8.165514853","8.24243893","8.60815086")
cc <- c("GSM1629982","3.969058399","5.722410064","7.165514853","6.24243893","7.60815086")
file3 <- data.frame(aa=aa,bb=bb,cc=cc)
result <- gene_ave(file3)
```

`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

`hgnc` *a matrix for Converting gene symbol to entrez_id or ensembl_gene_id*

Description

the columns represent "symbol", "locus_group", "locus_type", "entrez_id" and "ensembl_gene_id"

Usage

```
hgnc
```

Format

A matrix with 37647 rows and 5 column

`hgnc_file` *a matrix for Converting gene symbol.*

Description

a matrix for Converting gene symbol.

Usage

```
hgnc_file
```

Format

A matrix with 43547 rows and 52 column

`id_ava` *Gene id conversion types*

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(profile)
```

Arguments

`profile`	a data.frame

Value

a data.frame, gene symbols and their expression value

Examples

```
result <- id_conversion(profile)
```

`id_conversion_vector` *Gene id conversion*

Description

Gene id conversion

Usage

```
id_conversion_vector(from, to, IDs)
```

Arguments

`from`	one of "id_ava()"
`to`	one of "id_ava()"
`IDs`	the gene id which needed to convert

Value

a vector of genes

Examples

```
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

```
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)
```

Arguments

dirr	a string for the directory of methylation data download from tcga useing 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"))
```

`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 column

`prepare_chi` *Preparer file for chi-square test*

Description

Preparer file for chi-square test

Usage

```
prepare_chi(jieguo2)
```

Arguments

`jieguo2`	result of ann_merge()

Value

a matrix

Examples

```
jieguo3 <- 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,
2.0,2.0,2.0,2.0,2.0,2.0,2.0),nrow=5)
rownames(jieguo3) <- c("AJAP1","FHAD1","CLCNKB","CROCCP2","AL137798.3")
colnames(jieguo3) <- c("TCGA-DD-A4NS-10A-01D-A30U-01","TCGA-ED-A82E-01A-11D-A34Y-01",
"TCGA-WQ-A9G7-01A-11D-A36W-01","TCGA-DD-AADN-01A-11D-A40Q-01",
"TCGA-ZS-A9CD-10A-01D-A36Z-01","TCGA-DD-A1EB-11A-11D-A12Y-01")
cnv_chi_file <- prepare_chi(jieguo3)
```

`profile` *a matrix of gene expression data in TCGA*

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_file1, string)
```

Arguments

`input_file1`	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_fil <- data.frame(aa=aa,bb=bb,cc=cc)
rep1_result <- rep1(input_fil," /// ")
```

rep2 *Handle the case where one id corresponds to multiple genes*

Description

Handle the case where one id corresponds to multiple genes

Usage

```
rep2(input_file1, string)
```

Arguments

input_file1	input file, a data.frame or a matrix
string	a string,sep of the gene

Value

a matrix,when an id corresponds to multiple genes, the expression value is deleted

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_fil <- data.frame(aa=aa,bb=bb,cc=cc)
rep2_result <- rep2(input_fil," /// ")
```

tcga_cli_deal *Combine clinical information obtained from TCGA and extract survival data*

Description

Combine clinical information obtained from TCGA and extract survival data

Usage

```
tcga_cli_deal(Files_dir1)
```

Arguments

Files_dir1	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