Package 'GeoTcgaData'

August 12, 2022

Type Package

Title Processing various types of data on GEO and TCGA

Version 1.1.1

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 7.2.1

Suggests knitr, rmarkdown, DESeq2, S4Vectors, ChAMP, impute, tidyr, clusterProfiler, org.Hs.eg.db, edgeR, limma, quantreg, minfi, IlluminaHumanMethylation450kanno.ilmn12.hg19, dearseq, NOISeq

VignetteBuilder knitr

Imports utils, data.table, magrittr, plyr, cqn, dplyr, topconfects

Language en-US

NeedsCompilation no

Author Erqiang Hu [aut, cre]

Maintainer Erqiang Hu <13766876214@163.com>

Repository CRAN

Date/Publication 2022-08-12 08:20:08 UTC

R topics documented:

`arrayDiff` *arrayDiff*

Description

arrayDiff

Usage

```
arrayDiff(df, group, method = "limma")
```

Arguments

`df`	data.frame of the omic data
`group`	a vector, group of samples.
`method`	one of "limma", "ttest", "wilcox",

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 data.frame, 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_CNV *Do difference analysis of gene level copy number variation data*

Description

Do difference analysis of gene level copy number variation data

Usage

```
diff_CNV(cnvData, sampleGroup, ...)
```

Arguments

cnvData	data.frame of CNV data
sampleGroup	vector of sample group
...	parameters for fisher.test

Examples

```
## 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)
diffCnv <- 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

```
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)
```

Diff_limma *Diff_limma*

Description

Diff_limma

Usage

```
Diff_limma(df, group, adjust.method = "BH")
```

Arguments

df	data.frame of the omic data
group	a vector, group of samples.
adjust.method	adjust.method.

diff_RNA *Do difference analysis of RNA-seq data*

Description

Do difference analysis of RNA-seq data

Usage

```
diff_RNA(
  counts,
  group,
  method = "limma",
  geneLength = NULL,
  gccontent = NULL,
  filter = TRUE,
  edgeRNorm = TRUE,
  adjust.method = "BH",
  useTopconfects = TRUE
)
```

Arguments

counts	a dataframe or numeric matrix of raw counts data
group	sample groups
method	one of "DESeq2", "edgeR" , "limma", "dearseq" and "Wilcoxon".
geneLength	a vector of gene length.
gccontent	a vector of gene GC content.
filter	if TRUE, use filterByExpr to filter genes.
edgeRNorm	if TRUE, use edgeR to do normalization for dearseq method.
adjust.method	character string specifying the method used to adjust p-values for multiple testing. See p.adjust for possible values.
useTopconfects	if TRUE, use topconfects to provide a more biologically useful ranked gene list.

Examples

```
## Not run:
library(TCGAbiolinks)

query <- GDCquery(project = "TCGA-ACC",
                  data.category = "Transcriptome Profiling",
                  data.type = "Gene Expression Quantification",
                  workflow.type = "STAR - 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 = "STAR - 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[genes_bitr$ENTREZID, ]
rownames(gene_cov2) <- genes_bitr$ENSEMBL
genes <- intersect(rownames(dataPrep), rownames(gene_cov2))
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_RNA_ucsc` *Do difference analysis of RNA-seq data downloaded from ucsc*

Description

Do difference analysis of RNA-seq data downloaded from ucsc

Usage

```
diff_RNA_ucsc(ucscfile, ...)
```

Arguments

ucscfile	a dataframe or numeric matrix of ucsc RNA-seq data
...	additional parameters

Examples

```
## Not run:
ucscfile <- data.table::fread("TCGA-BRCA.htseq_counts.tsv.gz")
group <- sample(c("grp1", "grp2"), ncol(ucscfile) - 1, replace = TRUE)
result <- diff_RNA_ucsc(ucscfile, group = group)

## 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

```
## Not run:
library(TCGAbiolinks)
query <- GDCquery(
    project = "TCGA-CHOL",
    data.category = "Simple Nucleotide Variation",
    access = "open",
    legacy = FALSE,
    data.type = "Masked Somatic Mutation",
    workflow.type = "Aliquot Ensemble Somatic Variant Merging and Masking"
)
GDCdownload(query)
data_snp <- GDCprepare(query)
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)
```

---

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

geneExpress *a data.frame of gene expression data*

Description

the rowname is gene symbols

Usage

```
geneExpress
```

Format

A data.frame with 10779 rows and 2 column

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 *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_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(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

```
## 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

```
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

```
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 = NULL)
```

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"))
```

methyDiff *Get methylation difference gene*

Description

Get methylation difference gene

Usage

```
methyDiff(
  cpgData,
  sampleGroup,
  combineMethod = RobustRankAggreg::rhoScores,
  missing_value = "knn",
  region = "Body",
  model = "cpg",
  adjust.method = "BH"
)
```

Arguments

cpgData	data.frame of cpg beta value
sampleGroup	vector of sample group
combineMethod	method to combine the cpg pvalues
missing_value	Method to impute missing expression data, one of "zero" and "knn".
region	region of genes, one of "Body", "TSS1500", "TSS200", "3'UTR", "1stExon", "5'UTR", and "IGR".
model	if "cpg", step1: calculate difference cpgs; step2: calculate difference genes. if "gene", step1: calculate the methylation level of genes; step2: calculate difference genes.
adjust.method	character string specifying the method used to adjust p-values for multiple testing. See p.adjust for possible values.

---

methyDiff_ucsc *Title*

Description

Title

Usage

```
methyDiff_ucsc(
  methy,
  sampleGroup = NULL,
  missing_value = "knn",
  model = c("cpg", "gene"),
  combineMethod = RobustRankAggreg::rhoScores,
  region = "Body"
)
```

Arguments

methy	data.frame of the methylation data, which can be downloaded from UCSC Xena.
sampleGroup	a vector of "0" and "1" for group of samples. If null, the samples were divided into two groups: disease and normal.
missing_value	Method to impute missing expression data, one of "zero" and "knn".
model	if "cpg", step1: calculate difference cpgs; step2: calculate difference genes. if "gene", step1: calculate the methylation level of genes; step2: calculate difference genes.
combineMethod	method to combine the cpg pvalues.
region	region of genes, one of "Body", "TSS1500", "TSS200", "3'UTR", "1stExon", "5'UTR", and "IGR".

Examples

```
## Not run:
methy_file <- "TCGA.THCA.sampleMap_HumanMethylation450.gz"
methy <- data.table::fread(methy_file, sep = "\t", header = T)
library(ChAMP)
myImport <- champ.import(directory=system.file("extdata",package="ChAMPdata"))
myfilter <- champ.filter(beta=myImport$beta,pd=myImport$pd,
    detP=myImport$detP,beadcount=myImport$beadcount)
cpg_gene <- hm450.manifest.hg19[, c("probeID", "gene_HGNC")]
result <- methyDiff_ucsc(methy, cpg_gene)

## End(Not run)
```

---

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(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,
    2.0,2.0,2.0,2.0,2.0,2.0,2.0,2.0),nrow=5)
cnv <- as.data.frame(cnv)
rownames(cnv) <- c("AJAP1","FHAD1","CLCNKB","CROCCP2","AL137798.3")
colnames(cnv) <- 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(cnv)
```

---

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_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," /// ")
```

---

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_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

```
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," /// ")
```

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_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