Package ‘AnnoProbe’

January 20, 2025

Type Package

Title Annotate the Gene Symbols for Probes in Expression Array

Version 0.1.7

Date 2022-11-12

Maintainer Yonghe Xia <`xiayh17@gmail.com`>

Description We curated 147 of expression array, from 3 species(human,mouse,rat),
3 companies('Affymetrix','Illumina','Agilent'),
by aligning the 'Fasta' sequences of all probes of each platform to their corresponding reference genome,
and then annotate them to genes.

License Apache License (>= 2)

Encoding UTF-8

URL `https://github.com/jmzeng1314/AnnoProbe`

LazyData true

RoxygenNote 7.1.1

Depends R (>= 3.4.0)

Imports ggplot2, DT, ggpubr, pheatmap, utils, methods, Biobase, stats,
xml2, httr, curl

biocViews DataImport, Microarray, OneChannel, SAGE, Software,
TwoChannel

Suggests limma, GEOquery, knitr, rmarkdown

NeedsCompilation no

Author Jianming Zeng [aut],
Yujia Xiang [aut],
Yonghe Xia [ctb, cre]

Repository CRAN

Date/Publication 2022-11-14 08:30:11 UTC

Contents

annoGene 2
checkGPL 3
check_diff_genes 3
deg_heatmap 4
deg_volcano 5
filterEM 5
geoChina 6
getGPLList 7
GSE27533 7
GSE95166 8
idmap 8
printGPLInfo 9

Index **10**

`annoGene` *Annotate gene IDs according to GTF files in gencode*

Description

annoGene will return a data.frame of gene information or write them to a file (csv or html format). The user should set a list of genes to be annotated, with "ENSEMBL" or "SYMBOL" style.

Usage

```
annoGene(IDs, ID_type, species = "human", out_file)
```

Arguments

`IDs`	a list of genes
`ID_type`	the type of input IDs, should be "ENSEMBL" or "SYMBOL"
`species`	choose human or mouse, or rat, default: human
`out_file`	the filename, should be ".csv" or ".html".

Value

a dataframe which columns contain genesymbol, biotypes, ensembl ids and the positions of genes

Examples

```
IDs <- c("DDX11L1", "MIR6859-1", "OR4G4P", "OR4F5")
ID_type = "SYMBOL"
annoGene(IDs, ID_type)

annoGene(IDs, ID_type,out_file = tempfile(fileext = ".html"))
annoGene(IDs, ID_type,out_file = tempfile(fileext = ".csv"))
```

`checkGPL` *Check whether the input gpl in our platform list or not*

Description

Check whether the input gpl in our platform list or not

Usage

```
checkGPL(GPL = NULL)
```

Arguments

`GPL`	GPL(GEO platform) number, eg: GPL570

Value

returns a boolean value

Examples

```
checkGPL('GPL570')
checkGPL('GPL15314')
checkGPL('GPL10558')
```

`check_diff_genes` *Check a list of genes how they show difference.*

Description

How does a gene or a list of genes show difference between two group. The boxplot or heatmap will be drawed. just a wrap function of ggpubr and pheatmap.

Usage

```
check_diff_genes(gene, genes_expr, group_list)
```

Arguments

`gene`	A vector contains all gene ids of interest. Gene ids should be gene symbol.
`genes_expr`	An expression matrix, the rownames should be gene symbol.
`group_list`	A vector contains the group information of each samples in expression matrix

Value

A figure : boxplot or heatmap

Examples

```
attach(GSE95166)
check_diff_genes('LRCH3',genes_expr,group_list )

x=DEG$logFC
names(x)=rownames(DEG)
cg=c(names(head(sort(x),100)),  names(tail(sort(x),100)))
check_diff_genes(cg,genes_expr,group_list )
```

`deg_heatmap` *draw a heatmap for DEG result*

Description

`deg_heatmap` will draw a heatmap for you.

Usage

```
deg_heatmap(deg, genes_expr, group_list, topn = 20)
```

Arguments

`deg`	the result from limma.
`genes_expr`	the expression matrix
`group_list`	a vector
`topn`	the number of genes in heatmap, default:20

Value

a ggplot2 style figure.

Examples

```
attach(GSE27533)
deg_heatmap(DEG,genes_expr,group_list)
```

`deg_volcano` *draw a volcano for DEG result*

Description

`deg_volcano` will draw a volcano for you.

Usage

```
deg_volcano(need_deg, style = 1, p_thred = 0.05, logFC_thred = 1)
```

Arguments

`need_deg`	should be 3 columns : gene, logFC, p.value(or p.adjust
`style`	you can try 1 or 2, default: 1
`p_thred`	default:0.05
`logFC_thred`	default:1

Value

a ggplot2 style figure.

Examples

```
deg=GSE27533$DEG
need_deg=data.frame(symbols=rownames(deg), logFC=deg$logFC, p=deg$P.Value)
deg_volcano(need_deg,2)

deg_volcano(need_deg,1)
```

`filterEM` *Filter expression matrix based on annotation*

Description

`filterEM` will annotate the probes in expression matrix and remove the duplicated gene symbols. because there will be many probes mapped to same genes, we will only keep the max value one.

Usage

```
filterEM(probes_expr, probe2gene)
```

Arguments

`probes_expr`	is an expression matrix which rownames are probes of probe2gene and each column is a sample
`probe2gene`	the first column is probes and the second column is corresponding gene symbols

Value

a expression matrix which has been filtered duplicated gene symbols

Examples

```
attach(GSE95166)
# head(probes_expr)
# head(probe2gene)
genes_expr <- filterEM(probes_expr,probe2gene)
# head(genes_expr)
```

`geoChina` *Download expression dataset by GSE id*

Description

`geoChina` will download the expression matrix and phenotype data as ExpressionSet format from cloud in mainland China, it's a alternative method for getGEO function from GEOquery package. geoChina('gse1009') is the same as eSet=getGEO('gse1009', getGPL = F)

Usage

```
geoChina(gse = "GSE2546", mirror = "tencent", destdir = getwd())
```

Arguments

`gse`	input GSE id, such as GSE1009, GSE2546, gse1009.
`mirror`	"tencent" only for now.
`destdir`	The destination directory for data downloads.

Value

a list of ExpressionSet, which contains the expression matrix and phenotype data

Examples

```
## Not run:
geoChina('GSE1009',destdir=tempdir())

## End(Not run)
```

`getGPLList` *Get all GPL list in our package* `getGPLList` *returns all the GPL number checklist stored in package*

Description

Get all GPL list in our package `getGPLList` returns all the GPL number checklist stored in package

Usage

```
getGPLList()
```

Value

a data.frame which contains the gpl and name of array.

`GSE27533` *An example dataset*

Description

A dataset containing genes_expr, group_list, DEG

Usage

```
GSE27533
```

Format

A list with 6 elements:

genes_expr genes_expr, genes_expr

DEG DEG, DEG ...

`GSE95166` *An example dataset*

Description

A dataset containing eSet, probes_expr, probe2gene, genes_expr, group_list, DEG

Usage

```
GSE95166
```

Format

A list with 6 elements:

probes_expr probes_expr, probes_expr

probe2gene probe2gene, probe2gene

genes_expr genes_expr, genes_expr

group_list group_list, group_list

DEG DEG, DEG ...

`idmap` *Get Probe Annotation*

Description

`idmap` returns probe annotations for input gpl

Usage

```
idmap(gpl = "GPL570", type = "bioc", mirror = "tencent", destdir = getwd())
```

Arguments

`gpl`	GPL(GEO platform) number, eg: GPL570
`type`	source of probe anntation stored, one of "pipe", "bioc", "soft", default:"pipe"
`mirror`	"tencent" only for now
`destdir`	The destination directory for data downloads.

Value

probe annotaions

Examples

```
ids=idmap('GPL570',destdir=tempdir())

ids=idmap('GPL570',type='soft',destdir=tempdir())
ids=idmap('GPL18084',type='pipe',destdir=tempdir())
```

`printGPLInfo` *Print GPL information*

Description

Print GPL information

Usage

```
printGPLInfo(GPL = NULL)
```

Arguments

`GPL`	GPL(GEO platform) number, eg: GPL570

Value

print detail information of the input GEO platform

Examples

```
printGPLInfo('GPL93')
```

Index

∗ **datasets**
    GSE27533, 7
    GSE95166, 8

annoGene, 2

check_diff_genes, 3
checkGPL, 3

deg_heatmap, 4
deg_volcano, 5

filterEM, 5

geoChina, 6
getGPLList, 7
GSE27533, 7
GSE95166, 8

idmap, 8

printGPLInfo, 9