# Package 'BioDataome'

November 15, 2017

**Title** Functions used to preprocess datasets stored in BioDataome

**Version** 0.0.0.9000

**Description** BioDataome package contains all the functions used to download, preprocess and annotate gene expression and methylation microarray data from Gene Expression Omnibus, as well as RNASeq data from recount.

**Depends** R (>= 3.4.2), foreach

**Imports** GEOquery, Biobase, RCurl, Rfast, SCAN.UPC, XML, doParallel, parallel, rentrez

**License** What license is it under?

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 6.0.1

**VignetteBuilder** knitr

**Suggests** BiocStyle, knitr, rmarkdown

## R topics documented:

compareDsetList *Column wise comparison of a dataset to a list of datasets This function finds with which datasets of the list y, dataset x shares common samples. The datasets are in the form variables (probes) x samples. The number of variables (probes) in both datasets should be the same.*

## Description

Column wise comparison of a dataset to a list of datasets This function finds with which datasets of the list y, dataset x shares common samples. The datasets are in the form variables (probes) x samples. The number of variables (probes) in both datasets should be the same.

## Usage

```
compareDsetList(x, y)
```

## Arguments

| | |
|---|---|
| x | the path to a normalized dataset x |
| y | a character vector of all paths to datasets to compare |

## Value

a character vector of all datasets for which dataset x shares at least one sample, separated by ;

## Examples

```
Let us assume we want to compare normalized gene expression dataset GSE86013 with datasets GSE86015, GSE9008,
x and y can be either local paths where .Rda normalized data are stored or links to the csv files in BioDataome

First example runs with datasets stored in .csv in BioDataome.
Since these datasets are large we propose to use fread from package data.table to read datasets faster
install.packages("data.table")
library("data.table")
x<-"http://dataome.mensxmachina.org/data/Homo%20sapiens/GPL570/GSE86013.csv"
y<-c("GSE86015.csv","GSE9008.csv","GSE9119.csv")
y<-paste0("http://dataome.mensxmachina.org/data/Homo%20sapiens/GPL570/",y)
commonGSEs<-compareDsetList(x,y)
```

compareDsets *Column wise comparison of two datasets This function finds how many samples are shared between two datasets. The datasets are in the form variables (probes) x samples. The number of variables (probes) in both datasets should be the same*

## Description

Column wise comparison of two datasets This function finds how many samples are shared between two datasets. The datasets are in the form variables (probes) x samples. The number of variables (probes) in both datasets should be the same

### Usage

```
compareDsets(d1, d2)
```

### Arguments

| | |
|---|---|
| d1 | a numeric matrix of a dataset |
| d2 | a numeric matrix of a dataset |

### Value

the number of equal samples

### Examples

```
Let us assume we want to compare two normalized gene expression datasets from the same platform
d1<-get(load(url("http://dataome.mensxmachina.org/data/Homo%20sapiens/GPL570/GSE86013.Rda")))
d2<-get(load(url("http://dataome.mensxmachina.org/data/Homo%20sapiens/GPL570/GSE86015.Rda")))
```

---

## controlSamples *Discover control samples from phenotype data in GEO*

---

### Description

This function discovers control samples from the series matrix found in GEO It searches for specific keywords that are often used to denote controls, in specific columns of series matrices.

### Usage

```
controlSamples(d)
```

### Arguments

| | |
|---|---|
| d | a data frame with the contents of series matrix |

### Value

a data frame of GEO sample ids (i.e. GSM60555) and their class.

### Examples

```
phenos<-downloadPhenotypePlatform("GSE11761","GPL570")
controls<-controlSamples(phenos)
```

## curateGSE — *Run all steps to download, preprocess and annotate a GEO dataset*

### Description

Given a GSE id this function downloads, preprocesses, annotates a study and also creates the metadata. It saves two files, the data file and the GEO metadata file in the given path.

### Usage

```
curateGSE(x, y, z)
```

### Arguments

| | |
|---|---|
| x | a GSE series ID |
| y | a GEO platform id (GPL) |
| z | the path to write the output |

### Value

writes in the given path two data frames, the preprocessed data and the metadata file with phenotype information

### Examples

```
curateGSE("GSE11761","GPL570",getwd())
```

## curateRecountRNASeq — *Run all steps to download, preprocess and annotate an RNASeq dataset from Recount*

### Description

Run all steps to download, preprocess and annotate an RNASeq dataset from Recount

### Usage

```
curateRecountRNASeq(x, y)
```

### Arguments

| | |
|---|---|
| x | a recount dataset ID |
| y | the path to write the output |

### Value

writes in the given path two data frames, the preprocessed data and the metadata file with phenotype information

### Examples

```
curateRecountRNASeq("SRP032775",getwd())
```

## diseasetoChildrenNodes *Map a Disease Ontology (D-O) term to the first children nodes in D-O*

### Description

This function uses internal look up data to map a disease to its first children node.

### Usage

```
diseasetoChildrenNodes(x)
```

### Arguments

| | |
|---|---|
| x | a disease in D-O terms |

### Value

the first children node of x disease

### Examples

```
DOChild<-diseasetoChildrenNodes("vesiculitis")
```

## diseasetoParentNodes *Map a Disease Ontology (D-O) term to the parent nodes in D-O*

### Description

This function uses internal look up data to map a disease to its parent node.

### Usage

```
diseasetoParentNodes(x)
```

### Arguments

| | |
|---|---|
| x | a disease in D-O terms |

### Value

the parent node of x disease

### Examples

```
DOParent<-diseasetoParentNodes("vesiculitis")
```

`downloadPhenotype` *Download series matrices from GEO for a given study*

## Description

This function downlads all series matrices related to a given GEO Series (GSE) and saves them in a list. The same GSE study may be related to more than one platforms (i.e GPL570 and GPL1261). The length of the output list is the number of the related platforms.

## Usage

```
downloadPhenotype(x)
```

## Arguments

| | |
|---|---|
| x | a GEO Series id (GSE) |

## Value

a list of series matrices related to the given study

## Examples

```
downloadPhenotype("GSE11761")
```

`downloadPhenotypePlatform`
*Download series matrices from GEO for a given study and platform*

## Description

This function downloads the series matrices related to a given GEO Series (GSE) for the given platform

## Usage

```
downloadPhenotypePlatform(x, y)
```

## Arguments

| | |
|---|---|
| x | a GEO Series id (GSE) |
| y | a GEO platform id (GPL) |

## Value

a data frame with the contents of the series matrix

## Examples

```
downloadPhenotypePlatform("GSE11761","GPL570")
```

## downloadRaw *Download raw CEL files from GEO for a given study*

### Description

Download raw CEL files from GEO for a given study

### Usage

```
downloadRaw(x, y)
```

### Arguments

| | |
|---|---|
| x | a GEO Series id (GSE) |
| y | the path to save the downloaded files |

### Value

a directory with the GSE with the compressed RAW files

### Examples

```
downloadRaw("GSE11761")
```

## downloadRecount *Download gene-level RangedSummarizedExperiment data from Recount*

### Description

This function downloads the RangedSummarizedExperiment object with the data summarized at the gene level from Recount (https://jhubiostatistics.shinyapps.io/recount/)

### Usage

```
downloadRecount(x)
```

### Arguments

| | |
|---|---|
| x | a recount dataset ID |
| y | the destination path for the downloaded RangedSummarizedExperiment object |

### Value

RangedSummarizedExperiment object for the given study

### Examples

```
downloadRecount("SRP032775",getwd())
```

## entrezIDtoGSE *Find all GSE ids for a given Entrez query*

### Description

Find all GSE ids for a given Entrez query

### Usage

```
entrezIDtoGSE(x)
```

### Arguments

| | |
|---|---|
| x | an esearch object as a result of an entrez_search query |

### Value

a matrix the first column of which is the GSE id and the second the entrezID

### Examples

```
query GEO for all Homo sapiens studies with sample size between 200-300, measured with GPL570 and provide CEL
r_search <- entrez_search(db="gds", term="Homo sapiens[ORGN] AND CEL[SFIL] AND gpl570[ACCN] AND 200:300[Numbe
entrezIDtoGSE(r_search)
```

## GSEmetadata *Create metadata of a GEO dataset for BioDataome*

### Description

Given a GSE id this function downloads phenotype data from GEO for a specific study, discovers control samples and creates the metadata file for BioDataome

### Usage

```
GSEmetadata(x, y)
```

### Arguments

| | |
|---|---|
| x | a GSE series ID |
| y | a GEO platform id (GPL) |

### Value

a data frame of metadata with columns: sample IDs, Class and all other GEO phenotype data found in series matrix

### Examples

```
metadata<-GSEmetadata("GSE11761","GPL570")
```

## GSEtoDisease — *Annotate a study (GSE) with a disease term from the Disease Ontology by exploiting both PubTator and GEO*

### Description

Given a GSE id this function annotates the study with a disease term from the Disease Ontology (D-O): http://disease-ontology.org/ It provides the most specific disease term, meaning the term with the highest depth in the D-O.

### Usage

```
GSEtoDisease(GSE)
```

### Arguments

| | |
|---|---|
| GSE | a GSE series ID |

### Value

a character vector of all related diseases, separated by ;

### Examples

```
diseases<-GSEtoDisease("GSE10245")
```

## GSEtoDiseaseGEO — *Annotate a study (GSE) with a disease term from the Disease Ontology by exploiting only GEO*

### Description

Given a GSE id this function annotates the study with a disease term from the Disease Ontology (D-O): http://disease-ontology.org/ It provides the most specific disease term, meaning the term with the highest depth in the D-O.

### Usage

```
GSEtoDiseaseGEO(GSE)
```

### Arguments

| | |
|---|---|
| GSE | a GSE series ID |

### Value

a character vector of all related diseases, separated by ;

### Examples

```
diseases<-GSEtoDiseaseGEO("GSE10245")
```

## preprocessGEO *Preprocess CEL files with SCAN*

### Description

This function calls the SCAN method as described in Piccolo SR, Sun Y, Campbell JD, Lenburg ME, Bild AH and Johnson WE (2012). A single-sample microarray normalization method to facilitate personalized-medicine workflows. Genomics, 100(6), pp. 337-344.

### Usage

```
preprocessGEO(x, y)
```

### Arguments

| | |
|---|---|
| x | the path where the CEL files are stored |
| y | the number of cores to run in parallel |

### Value

a matrix of dimensions: probes x samples with the normalized expression values

### Examples

```
Assuming that CEL files are located in working directory
preprocessGEO(getwd(),3)
```

## preprocessGEOMethylation *Preprocess IDAT files from Illumina HumanMethylation450 BeadChip*

### Description

This function utilizes minfi Package to convert data into methylation measurements.

### Usage

```
preprocessGEOMethylation(x)
```

### Arguments

| | |
|---|---|
| x | a character array with the paths to the idat files |

### Value

a matrix of dimensions: probes x samples with the normalized methylation values

## Examples

```
Assuming there is a directory named GSE78279 in the working directory where idat files are stored and
"GSM2071074_8655685078_R03C02" and "GSM2071074_8655685078_R03C02" are the file names for the idat files,
then x should be:
x<-c("GSM2071074_8655685078_R03C02","GSM2071075_8655685078_R04C02")
x<-file.path(getwd(),"GSE78279",x)
dataNorm<-preprocessGEOMethylation(x)
```

---

# recountIDtoGSE *Map a recount dataset ID to GSE ID*

---

## Description

Map a recount dataset ID to GSE ID

## Usage

```
recountIDtoGSE(x)
```

## Arguments

| | |
|---|---|
| x | a recount dataset ID |

## Value

a GSE series ID

## Examples

```
recountIDtoGSE("SRP032775")
```

# Index