Package ‘RTCGAToolbox’

April 23, 2016

Type Package

Title A new tool for exporting TCGA Firehose data

Version 2.0.0

Date 2015-05-28

Author Mehmet Kemal Samur

Maintainer Mehmet Kemal Samur `<samur@jimmy.harvard.edu>`

Description Managing data from large scale projects such as The Cancer Genome Atlas (TCGA) for further analysis is an important and time consuming step for research projects. Several efforts, such as Firehose project, make TCGA pre-processed data publicly available via web services and data portals but it requires managing, downloading and preparing the data for following steps. We developed an open source and extensible R based data client for Firehose pre-processed data and demonstrated its use with sample case studies. Results showed that RTCGAToolbox could improve data management for researchers who are interested with TCGA data. In addition, it can be integrated with other analysis pipelines for following data analysis.

License GPL (>= 2)

biocViews Sequencing, DifferentialExpression, GeneExpression

VignetteBuilder knitr

Suggests BiocStyle, knitr, rmarkdown, Homo.sapiens

Imports methods,XML,limma (>= 3.18),survival,RCircos,data.table (>= 1.9.4),RCurl,RJSONIO

Depends R (>= 3.2.0)

NeedsCompilation no

R topics documented:

`CorResult-class` *An S4 class to store correlations between gene expression level and copy number data*

Description

An S4 class to store correlations between gene expression level and copy number data

Slots

`Dataset` A cohort name

`Correlations` Results data frame

`DGEResult-class` *An S4 class to store differential gene expression results*

Description

An S4 class to store differential gene expression results

Slots

`Dataset` Dataset name

`Toptable` Results data frame

`FirehoseCGHArray-class` *An S4 class to store data from CGA platforms*

Description

An S4 class to store data from CGA platforms

Slots

`Filename` Platform name

`DataMatrix` A data frame that stores the CGH data.

`FirehoseData-class` *An S4 class to store main data object from clinent function.*

Description

An S4 class to store main data object from clinent function.

Slots

`Dataset` A cohort name

`Clinical` Clinical data frame

`RNASeqGene` Gene level expression data matrix from RNAseq

`RNASeq2GeneNorm` Gene level expression data matrix from RNAseq (RSEM)

`miRNASeqGene` miRNA expression data from matrix smallRNAseq

`CNASNP` A data frame to store somatic copy number alterations from SNP array platform

`CNVSNP` A data frame to store germline copy number variants from SNP array platform

`CNAseq` A data frame to store somatic copy number alterations from sequencing platform

`CNACGH` A list that stores `FirehoseCGHArray` object for somatic copy number alterations from CGH platform

`Methylation` A list that stores `FirehoseMethylationArray` object for methylation data

`mRNAArray` A list that stores `FirehosemRNAArray` object for gene expression data from microarray

`miRNAArray` A list that stores `FirehosemRNAArray` object for miRNA expression data from microarray

`RPPAArray` A list that stores `FirehosemRNAArray` object for RPPA data

`Mutations` A data frame for mutation infromation from sequencing data

`GISTIC` A `FirehoseGISTIC` object to store processed copy number data

`BarcodeUUID` A data frame that stores the Barcodes, UUIDs and Short sample identifiers

---

`FirehoseGISTIC-class` *An S4 class to store processed copy number data. (Data processed by using GISTIC2 algorithm)*

Description

An S4 class to store processed copy number data. (Data processed by using GISTIC2 algorithm)

Slots

`Dataset` Cohort name

`AllByGene` A data frame that stores continuous copy number

`ThresholedByGene` A data frame for discrete copy number data

`FirehoseMethylationArray-class` *An S4 class to store data from methylation platforms*

Description

An S4 class to store data from methylation platforms

Slots

`Filename` Platform name

`DataMatrix` A data frame that stores the methylation data.

`FirehosemRNAArray-class` *An S4 class to store data from array (mRNA, miRNA etc.) platforms*

Description

An S4 class to store data from array (mRNA, miRNA etc.) platforms

Slots

`Filename` Platform name

`DataMatrix` A data matrix that stores the expression data.

`getCNGECorrelation` *Perform correlation analysis betwwen gene expression and copy number data*

Description

`getCNGECorrelation` returns a list that stores the results correlation between gene expression and copy number data.

Usage

```
getCNGECorrelation(dataObject, adj.method = "BH", adj.pval = 0.05,
  raw.pval = 0.05)
```

Arguments

`dataObject`	This must be `FirehoseData` object.
`adj.method`	Raw p value adjustment methods (Default "BH")
`adj.pval`	Adjusted p value cut off for results table (Default 0.05)
`raw.pval`	raw p value cut off for results table (Default 0.05)

Value

Returns a list that stores results for each dataset

Examples

```
data(RTCGASample)
corRes = getCNGECorrelation(RTCGASample)
corRes
showResults(corRes[[1]])
## Not run:
```

`getData` *Export data from FirehoseData object*

Description

Export data from FirehoseData object

Usage

```
getData(object, type = "", platform = NULL, CN = "All")
```

Arguments

`object`	A `FirehoseData` object
`type`	A data type to be exported (Data types can be seen by typing show(objectname))
`platform`	A list id for data types that may come from multiple platform (such as mRNAArray)
`CN`	A copy number data type (Default: 'All') (Possible values 'All' or 'Thresholed')

Value

Returns matrix or data frame depends on data type

Examples

```
data(RTCGASample)
sampleClinical = getData(RTCGASample,"Clinical")
sampleClinical = getData(RTCGASample,"RNASeqGene")
```

---

getData,FirehoseData-method

Export data from FirehoseData object

Description

Export data from FirehoseData object

Usage

```
## S4 method for signature 'FirehoseData'
getData(object, type = "", platform = NULL,
  CN = "All")
```

Arguments

`object`	A `FirehoseData` object
`type`	A data type to be exported (Data types can be seen by typing show(objectname))
`platform`	A list id for data types that may come from multiple platform (such as mRNAArray)
`CN`	A copy number data type (Default: 'All') (Possible values 'All' or 'Thresholed')

Value

Returns matrix or data frame depends on data type

Examples

```
data(RTCGASample)
sampleClinical = getData(RTCGASample,"Clinical")
sampleClinical = getData(RTCGASample,"RNASeqGene")
```

`getDiffExpressedGenes` *Perform differential gene expression analysis for mRNA expression data.*

Description

`getDiffExpressedGenes` returns a list that stores the results for each dataset.

Usage

```
getDiffExpressedGenes(dataObject, DrawPlots = TRUE, adj.method = "BH",
  adj.pval = 0.05, raw.pval = 0.05, logFC = 2, hmTopUpN = 100,
  hmTopDownN = 100, meanFilter = 10)
```

Arguments

`dataObject`	This must be `FirehoseData` object.
`DrawPlots`	A logical parameter to draw heatmaps and volcano plots.
`adj.method`	Raw p value adjustment methods (Default "BH")
`adj.pval`	Adjusted p value cut off for results table (Default 0.05)
`raw.pval`	raw p value cut off for results table (Default 0.05)
`logFC`	log fold change cut off for results table (Default 2)
`hmTopUpN`	Max number of up regulated genes in heatmap (Default 100)
`hmTopDownN`	Max number of down regulated genes in heatmap (Default 100)
`meanFilter`	Mean read counts for each gene to filter not expressed genes (Default 10)

Value

Returns a list that stores results for each dataset

Examples

```
data(RTCGASample)
dgegenes = getDiffExpressedGenes(RTCGASample)
dgegenes
showResults(dgegenes[[1]])
dgegenes = showResults(dgegenes[[1]])
head(dgegenes)
## Not run:
```

`getFirehoseAnalyzeDates` *Get data analyze dates.*

Description

`getFirehoseAnalyzeDates` returns the character vector for analyze release dates.

Usage

```
getFirehoseAnalyzeDates(last = NULL)
```

Arguments

`last`	To list last n dates. (Default NULL)

Value

A character vector for dates.

Examples

```
getFirehoseAnalyzeDates()
getFirehoseAnalyzeDates(last=2)
```

`getFirehoseData` *Get data from Firehose portal.*

Description

`getFirehoseData` returns `FirehoseData` object that stores TCGA data.

Usage

```
getFirehoseData(dataset, runDate = NULL, gistic2_Date = NULL,
  RNAseq_Gene = FALSE, Clinic = TRUE, miRNASeq_Gene = FALSE,
  RNAseq2_Gene_Norm = FALSE, CNA_SNP = FALSE, CNV_SNP = FALSE,
  CNA_Seq = FALSE, CNA_CGH = FALSE, Methylation = FALSE,
  Mutation = FALSE, mRNA_Array = FALSE, miRNA_Array = FALSE,
  RPPA = FALSE, RNAseqNorm = "raw_counts",
  RNAseq2Norm = "normalized_count", forceDownload = FALSE, destdir = ".",
  fileSizeLimit = 500, getUUIDs = FALSE)
```

Arguments

`dataset`	A cohort name. All dataset names can be accessible via `getFirehoseDatasets`
`runDate`	Standard data run dates. Date list can be accessible via `getFirehoseRunningDates`
`gistic2_Date`	Analyze running dates for GISTIC processed copy number data. Date list can be accessible via `getFirehoseAnalyzeDates`
`RNAseq_Gene`	Logical (default FALSE) parameter for RNAseq data.
`Clinic`	Logical (default TRUE) parameter for clinical data.
`miRNASeq_Gene`	Logical (default FALSE) parameter for smallRNAseq data.
`RNAseq2_Gene_Norm`	Logical (default FALSE) parameter for RNAseq v2 (RSEM processed) data.
`CNA_SNP`	Logical (default FALSE) parameter for somatic copy number alterations data from SNP array.
`CNV_SNP`	Logical (default FALSE) parameter for germline copy number variants data from SNP array.
`CNA_Seq`	Logical (default FALSE) parameter for somatic copy number alterations data from sequencing.
`CNA_CGH`	Logical (default FALSE) parameter for somatic copy number alterations data from CGH.
`Methylation`	Logical (default FALSE) parameter for methylation data.
`Mutation`	Logical (default FALSE) parameter for mutation data from sequencing.
`mRNA_Array`	Logical (default FALSE) parameter for mRNA expression data from microarray.
`miRNA_Array`	Logical (default FALSE) parameter for miRNA expression data from microarray.
`RPPA`	Logical (default FALSE) parameter for RPPA data
`RNAseqNorm`	RNAseq data normalization method. (Default raw_counts)
`RNAseq2Norm`	RNAseq v2 data normalization method. (Default normalized_count)
`forceDownload`	A logic (Default FALSE) key to force download RTCGAToolbox every time. By default if you download files into your working directory once than RTCGAToolbox using local files next time.
`destdir`	Directory in which to store the resulting downloaded file. Defaults to current working directory.
`fileSizeLimit`	Files that are larger than set value (megabyte) won't be downloaded (Default: 500)
`getUUIDs`	Logical key to get UUIDs from barcode (Default: FALSE)

Details

This is a main client frution to download data from Firehose TCGA portal.

Value

A `FirehoseData` data object that stores data for selected data types.

Examples

```
#Sample Dataset
data(RTCGASample)
RTCGASample
## Not run:
BRCAdata <- getFirehoseData(dataset="BRCA",
runDate="20140416",gistic2_Date="20140115",
RNAseq_Gene=TRUE,Clinic=TRUE,mRNA_Array=TRUE,Mutation=TRUE)

## End(Not run)
```

`getFirehoseDatasets` *Get list of TCGA cohorts.*

Description

`getFirehoseDatasets` returns a character array for cohorts.

Usage

```
getFirehoseDatasets()
```

Value

A character string

Examples

```
getFirehoseDatasets()
```

`getFirehoseRunningDates` *Get standard data running dates.*

Description

`getFirehoseRunningDates` returns the character vector for standard data release dates.

Usage

```
getFirehoseRunningDates(last = NULL)
```

Arguments

`last`	To list last n dates. (Default NULL)

Value

A character vector for dates.

Examples

```
getFirehoseRunningDates()
getFirehoseRunningDates(last=2)
```

`getMutationRate` *Make a table for mutation rate of each gene in the cohort*

Description

Make a table for mutation rate of each gene in the cohort

Usage

```
getMutationRate(dataObject)
```

Arguments

`dataObject`	This must be `FirehoseData` object.

Value

Returns a data table

Examples

```
data(RTCGASample)
mutRate = getMutationRate(dataObject=RTCGASample)
mutRate <- mutRate[order(mutRate[,2],decreasing = TRUE),]
head(mutRate)
## Not run:
```

`getReport` *Draws a circle plot into working directory*

Description

`getReport` draws a circle plot into your workin director to show log fold changes for differentially expressed genes, copy number alterations and mutations.

Usage

```
getReport(dataObject, DGEResult1 = NULL, DGEResult2 = NULL, geneLocations)
```

Arguments

`dataObject`	This must be `FirehoseData` object.
`DGEResult1`	Differential gene expression results object (Optional)
`DGEResult2`	Differential gene expression results object (Optional)
`geneLocations`	Gene coordinates.

Value

Draws a circle plot

Examples

```
data(RTCGASample)
require("Homo.sapiens")
locations = genes(Homo.sapiens,columns="SYMBOL")
locations = as.data.frame(locations)
locations <- locations[,c(6,1,5,2:3)]
locations <- locations[!is.na(locations[,1]),]
rownames(locations) <- locations[,1]
t1=getDiffExpressedGenes(RTCGASample)
## Not run:
getReport(dataObject=RTCGASample,DGEResult1=t1[[1]],geneLocations=locations)

## End(Not run)
```

`getSurvival` *Perform survival analysis based on gene expression data*

Description

`getSurvival` draws a KM plot and show survival analysis results between groups that are defined by gene expression data

Usage

```
getSurvival(dataObject, numberofGroups = 2, geneSymbols, sampleTimeCensor)
```

Arguments

`dataObject`	This must be `FirehoseData` object.
`numberofGroups`	Can be set as 2 or 3. (Default 2) Order and divide samples into n groups by using gene expression data.
`geneSymbols`	Gene symbol that is going to be tested
`sampleTimeCensor`	a data frame that stores clinical data. First column should store sample IDs, second column should have time and third column should have event information. For more information please see vignette.

Value

Draws a KM plot

Examples

```
## get data with  getFirehoseData() function and call survival analysis
## Always check clinical data file for structural changes
data(RTCGASample)
clinicData <- getData(RTCGASample,"Clinical")
clinicData = clinicData[,3:5]
clinicData[is.na(clinicData[,3]),3] = clinicData[is.na(clinicData[,3]),2]
survData <- data.frame(Samples=rownames(clinicData),Time=as.numeric(clinicData[,3]),
Censor=as.numeric(clinicData[,1]))
getSurvival(dataObject=RTCGASample,geneSymbols=c("FCGBP"),sampleTimeCensor=survData)
```

`hg19.ucsc.gene.locations` *Gene coordinates for circle plot.*

Description

A dataset containing the gene coordinates The variables are as follows:

Format

A data frame with 28454 rows and 5 variables

Details

- GeneSymbol. Gene symbols
- Chromosome. Chromosome name
- Strand. Gene strand on chromosome
- Start. Gene location on chromosome
- End. Gene location on chromosome

`RTCGASample` *A sample data object for sample codes.*

Description

A `FirehoseData` object for running sample codes. The variables are as follows:

Format

A `FirehoseData` data object

Details

- a2. A sample data object

`RTCGAToolbox` *RTCGAToolbox: A New Tool for Exporting TCGA Firehose Data*

Description

Managing data from large-scale projects (such as The Cancer Genome Atlas (TCGA) for further analysis is an important and time consuming step for research projects. Several efforts, such as the Firehose project, make TCGA pre-processed data publicly available via web services and data portals, but this information must be managed, downloaded and prepared for subsequent steps. We have developed an open source and extensible R based data client for pre-processed data from the Firehose, and demonstrate its use with sample case studies. Results show that our RTCGAToolbox can facilitate data management for researchers interested in working with TCGA data. The RTCGAToolbox can also be integrated with other analysis pipelines for further data processing.

Details

The main function you're likely to need from **RTCGAToolbox** is `getFirehoseData`. Otherwise refer to the vignettes to see how to use the **RTCGAToolbox**

Author(s)

Mehmet Kemal Samur

`showResults` *Export toptable or correlation data frame*

Description

Export toptable or correlation data frame

Usage

```
showResults(object)
```

Arguments

`object`	A `DGEResult` or `CorResult` object

Value

Returns toptable or correlation data frame

Examples

```
data(RTCGASample)
dgeRes = getDiffExpressedGenes(RTCGASample)
dgeRes
showResults(dgeRes[[1]])
```

`showResults,CorResult-method`

Export toptable or correlation data frame

Description

Export toptable or correlation data frame

Usage

```
## S4 method for signature 'CorResult'
showResults(object)
```

Arguments

`object`	A `DGEResult` or `CorResult` object

Value

Returns correlation results data frame

Examples

```
data(RTCGASample)
corRes = getCNGECorrelation(RTCGASample,adj.pval = 1,raw.pval = 1)
corRes
showResults(corRes[[1]])
```

`showResults,DGEResult-method`

Export toptable or correlation data frame

Description

Export toptable or correlation data frame

Usage

```
## S4 method for signature 'DGEResult'
showResults(object)
```

Arguments

`object`	A `DGEResult` or `CorResult` object

Value

Returns toptable for DGE results

Examples

```
data(RTCGASample)
dgeRes = getDiffExpressedGenes(RTCGASample)
dgeRes
showResults(dgeRes[[1]])
```

Index