Package ‘TFutils’

October 16, 2018

Title TFutils

Description Package to work with TF data.

Version 1.0.0

Depends R (>= 3.5.0)

Imports methods, GenomicRanges, IRanges, S4Vectors, GSEABase, shiny, miniUI, data.table, dplyr, magrittr, AnnotationDbi, Rsamtools, GenomeInfoDb, GenomicFiles, AnnotationFilter, GenomicFeatures, Gviz, utils, stats, Biobase

Suggests knitr, DT, Homo.sapiens, GO.db, org.Hs.eg.db, ensembldb, EnsDb.Hsapiens.v75, testthat, BiocParallel, BiocStyle, SummarizedExperiment, UpSetR, ggplot2

License Artistic-2.0

LazyLoad yes

LazyData true

VignetteBuilder knitr

RoxygenNote 6.0.1

biocViews Transcriptomics

Roxygen list(markdown = TRUE)

git_url https://git.bioconductor.org/packages/TFutils

git_branch RELEASE_3_7

git_last_commit 2f08b9b

git_last_commit_date 2018-04-30

Date/Publication 2018-10-15

Author Vincent Carey [aut],
Shweta Gopaulakrishnan [cre, aut]

Maintainer Shweta Gopaulakrishnan <reshg@channing.harvard.edu>

R topics documented:

cisbpTFcat *cisbpTFcat: data.frame with information on CISBP TFs for human*

Description

cisbpTFcat: data.frame with information on CISBP TFs for human

Usage

```
cisbpTFcat
```

Format

data.frame

Note

Extracted March 2018

Source

http://cisbp.ccbr.utoronto.ca/bulk.php select Homo_sapiens

Examples

```
head(TFutils::cisbpTFcat)
```

`fimo16` *fimo16: GenomicFiles instance to AWS S3-resident FIMO bed for 16 TFs*

Description

fimo16: GenomicFiles instance to AWS S3-resident FIMO bed for 16 TFs

Usage

```
fimo16
```

Format

GenomicFiles for a TabixFileList

Source

K. Glass ran FIMO

Examples

```
TFutils::fimo16
```

`fimoMap` *fimoMap: table with Mnnnn (motif PWM tags) and HGNC symbols for TFs*

Description

fimoMap: table with Mnnnn (motif PWM tags) and HGNC symbols for TFs

Usage

```
fimoMap
```

Format

data.frame

Source

Kimberly Glass (rekrg@channing.harvard.edu)

Examples

```
head(TFutils::fimoMap)
```

genemodelDF *use EnsDb to generate an exon-level model of genes identified by symbol*

Description

use EnsDb to generate an exon-level model of genes identified by symbol

Usage

```
genemodelDF(sym, resource, columnsKept = c("gene_id", "tx_id"), ...)
```

Arguments

sym	a character() vector of gene symbols
resource	should be or inherit from EnsDb, answering exons(), with AnnotationFilter::SymbolFilter as filter parameter
columnsKept	character vector used as columns param in exons()
...	passed to exons()

Value

data.frame instance with exons in rows

Note

There are many approaches available to acquiring 'gene models' in Bioconductor; this one emphasizes the use of the exons method for Ensembl annotation.

Examples

```
if (requireNamespace("EnsDb.Hsapiens.v75")) {
 orm = genemodelDF("ORMDL3", EnsDb.Hsapiens.v75::EnsDb.Hsapiens.v75)
 dim(orm)
}
head(orm)
```

genemodForGviz *create a GeneRegionTrack instance for selected symbols*

Description

create a GeneRegionTrack instance for selected symbols

Usage

```
genemodForGviz(sym = "ORMDL3", id_elem = c("symbol", "tx_id"),
  resource = EnsDb.Hsapiens.v75::EnsDb.Hsapiens.v75, ...)
```

Arguments

`sym`	character vector of gene symbols, should be neighboring genes
`id_elem`	vector of names of columns generated by genemodelDF to be used to label transcripts
`resource`	should be or inherit from EnsDb, answering exons(), with AnnotationFilter::SymbolFilter as filter parameter
`...`	passed to genemodelDF

Value

instance of Gviz GeneRegionTrack

Note

This function helps to display the locations of TF binding sites in the context of complex gene models. A complication is that we have nice visualization of quantitative affinity predictions for TFs in the vignette, based on ggplot2, but it is not clear how to use that specific code to work with Gviz.

Examples

```
if (requireNamespace("EnsDb.Hsapiens.v75") &
    requireNamespace("Gviz")) {
 orm = genemodForGviz("ORMDL3", resource= EnsDb.Hsapiens.v75::EnsDb.Hsapiens.v75)
 orm
 Gviz::plotTracks(orm, showId=TRUE) # change id_elem for shorter id string
}
```

---

grabTab *create table of TF targets and related metadata*

Description

create table of TF targets and related metadata

Usage

```
grabTab(tfstub = "STAT1", gscoll = TFutils::tftColl,
  orgdb = org.Hs.eg.db::org.Hs.eg.db, gwrngs = TFutils::gwascat_hg19_chr17)
```

Arguments

`tfstub`	character(1) gene-like symbol for TF; will be grepped in names(gscoll)
`gscoll`	a GSEABase GeneSetCollection
`orgdb`	an instance of OrgDb as defined in AnnotationDbi
`gwrngs`	a GRanges representing EBI gwascat, must have `DISEASE/TRAIT`, `MAPPED_GENE`

Value

data.frame instance

Note

This function will link together information on targets of a given TF to the GWAS catalog.

Examples

```
gt = grabTab("VDR", gscoll=TFutils::tftColl,
   orgdb=org.Hs.eg.db::org.Hs.eg.db, gwrngs=TFutils::gwascat_hg19_chr17)
dim(gt)
head(gt)
```

`gwascat_hg19_chr17` *gwascat_hg19: GRanges of march 21 2018 EBI gwascat, limit to chr17*

Description

gwascat_hg19: GRanges of march 21 2018 EBI gwascat, limit to chr17

Usage

```
gwascat_hg19_chr17
```

Format

GenomicRanges GRanges instance

Source

gwascat::makeCurrentGwascat, with gwascat:::lo38to19 applied

Examples

```
TFutils::gwascat_hg19_chr17[,1:5]
```

`HGNCmap` *simple accessor for HGNCmap component of TFCatalog*

Description

simple accessor for HGNCmap component of TFCatalog

Usage

```
HGNCmap(x)
```

Arguments

x	instance of TFCatalog

Value

dataframe instance

Examples

```
HGNCmap
```

`hocomoco.mono` *hocomoco.mono: data.frame with information on HOCOMOCO TFs for human*

Description

hocomoco.mono: data.frame with information on HOCOMOCO TFs for human

Usage

```
hocomoco.mono
```

Format

data.frame

Note

Extracted March 2018

Source

http://hocomoco11.autosome.ru/human/mono?full=true

Examples

```
head(TFutils::hocomoco.mono)
```

`importFIMO,TabixFile,GRanges-method`
import a FIMO bed-like file

Description

import a FIMO bed-like file

Usage

```
## S4 method for signature 'TabixFile,GRanges'
importFIMO(src, parms, ...)

## S4 method for signature 'character,missing'
importFIMO(src, parms, ...)
```

Arguments

`src`	TabixFile instance
`parms`	a GRanges instance delimiting the import; multiple GRanges can be used
`...`	passed to GenomicRanges::GRanges

Value

instance of GRanges

Examples

```
if (requireNamespace("Rsamtools")) {
 tf = Rsamtools::TabixFile(system.file("M5946_1/chr1.bed.gz", package="TFutils"))
 importFIMO(tf, GenomicRanges::GRanges("chr1", IRanges::IRanges(1e6,11e6)))
 }
```

importFIMO_local_split *utility to read FIMO outputs from local resource(cluster), assuming bed text split by chromosome*

Description

utility to read FIMO outputs from local resource(cluster), assuming bed text split by chromosome

Usage

```
importFIMO_local_split(tf, chr)
```

Arguments

`tf`	character(1) file id
`chr`	character(1) chromosome name

Value

data.table instance

Examples

```
requireNamespace("GenomicRanges")
requireNamespace("IRanges")
importFIMO_local_split("M5946_1", "chr1")
dim(importFIMO_local_split("M5946_1", "chr17"))
```

metadata_tf *metadata_tf: list with metadata (motif_if and hgnc_symbol) about all the CISBP FIMO scan TF bed files*

Description

metadata_tf: list with metadata (motif_if and hgnc_symbol) about all the CISBP FIMO scan TF bed files

Usage

```
metadata_tf
```

Format

list

Source

K. Glass ran FIMO

Examples

```
TFutils::metadata_tf
```

named_tf *named_tf: named list with the names being the hgnc_symbol of the motif_id*

Description

named_tf: named list with the names being the hgnc_symbol of the motif_id

Usage

```
named_tf
```

Format

list

Source

K. Glass ran FIMO

Examples

```
TFutils::named_tf
named_tf[["VDR"]]
```

show,TFCatalog-method *produce a concise report on TFCatalog instance*

Description

produce a concise report on TFCatalog instance

Usage

```
## S4 method for signature 'TFCatalog'
show(object)
```

Arguments

object	instance of TFCatalog

Value

side effect

TFCatalog *Constructor for TFCatalog*

Description

Constructor for TFCatalog

Usage

```
TFCatalog(name, nativeIds, HGNCmap, metadata)
```

Arguments

name	informative character(1) for collection
nativeIds	character() vector of identifiers used by collection creators
HGNCmap	data.frame with column 1 nativeIds, column 2 HGNC or hgnc.heur for MSigDb and any other columns of use
metadata	a list of metadata elements

Value

instance of TFCatalog

Examples

```
TFs_MSIG = TFCatalog(name="MsigDb.TFT",nativeIds=names(tftColl),
HGNCmap=data.frame(tftCollMap,stringAsFactors=FALSE))
TFs_MSIG
```

TFCatalog-class *define a structure to hold information about TFs from diverse reference sources*

Description

define a structure to hold information about TFs from diverse reference sources

Slots

`name` character

`nativeIds` character tokens used by the provider to enumerate transcription factors

`HGNCmap` data.frame with atleast two columns, native id as first column and HGNC symbol as second column

`metadata` ANY

Note

This class respects the notions that 1) a source of information about transcription factors should have a name, 2) each source has its own 'native' nomenclature for the factors themselves, 3) it is common to use the gene symbol to refer to the transctiption factor, and 4) additional metadata will frequently be required to establish information about provenance of assertions about transcription factors.

tfhash *tfhash: data.frame with MSigDb TFs, TF targets as symbol or ENTREZ*

Description

tfhash: data.frame with MSigDb TFs, TF targets as symbol or ENTREZ

Usage

```
tfhash
```

Format

list

Source

MSigDb "c3" (motif gene sets) has been harvested for simple annotation of TFs and targets.

Examples

```
TFutils::tfhash
tfhash[1:3,]
```

TFtargs *gadget to help sort through tags naming TFs*

Description

gadget to help sort through tags naming TFs

Usage

```
TFtargs(gscoll = TFutils::tftColl, initTF = "VDR_Q3",
  gadtitle = "Search for a TF; its targets will be checked for mapped status in GWAS catalog")
```

Arguments

`gscoll`	a GSEABase GeneSetCollection
`initTF`	character(1) initial TF string for app
`gadtitle`	character(1) a title for the gadget panel

Value

on app conclusion a data.frame is returned

Note

Will use TFutils::gwascat_hg19_chr17 to look for 'MAPPED_GENE' field entries matching targets, also hardcoded to use org.Hs.eg.db to map symbols

Examples

```
if (interactive()) TFtargs()
```

tftColl *tftColl: GSEABase GeneSetCollection for transcription factor targets*

Description

tftColl: GSEABase GeneSetCollection for transcription factor targets

Usage

```
tftColl
```

Format

GSEABase GeneSetCollection instance

Note

run GSEABase::getGMT() on c3/TFT geneset collection from MSigDb

Source

broad institute

Examples

```
TFutils::tftColl
```

`tftCollMap` *tftCollMap: data.frame with information on MSigDb TFs for human*

Description

tftCollMap: data.frame with information on MSigDb TFs for human

Usage

```
tftCollMap
```

Format

data.frame

Note

Annotation of TFs is ad-hoc. GeneSet names were tokenized, splitting by underscore, and then fragments were matched to SYMBOL and ALIAS elements of org.Hs.eg.db. Extracted March 2018

Source

http://software.broadinstitute.org/gsea/msigdb/genesets.jsp?collection=TFT

Examples

```
head(TFutils::tftCollMap)
```

`URL_s3_tf` *utility to generate link to biocfound bucket for FIMO TFBS scores*

Description

utility to generate link to biocfound bucket for FIMO TFBS scores

Usage

```
URL_s3_tf(tag = "M3433")
```

Arguments

`tag`	character(1) token identifying TF, can be an HGNC gene name or Mnnnn PWM tag. It must be findable in TFutils::fimoMap table.

Value

character(1) URL

Examples

```
URL_s3_tf
```

Index