

GenomicState

October 16, 2024

GenomicStateHub

Access GenomicState objects through AnnotationHub

Description

This function uses AnnotationHub to obtain the objects provided by this package. These are: the TxDb object made with `gencode_txdb()`, the annotated genes object made with `gencode_annotated_genes()` or the GenomicState object made with `gencode_genomic_state()`.

Usage

```
GenomicStateHub(  
  version = "31",  
  genome = c("hg38", "hg19"),  
  filetype = c("TxDb", "AnnotatedGenes", "GenomicState"),  
  ah = AnnotationHub::AnnotationHub()  
)
```

Arguments

| | |
|----------|--|
| version | A character(1) with the Gencode version number. |
| genome | A character(1) with the human genome version number. Valid options are 'hg38' or 'hg19'. |
| filetype | A character() with either TxDb, AnnotatedGenes or GenomicState. |
| ah | An AnnotationHub object AnnotationHub-class . |

Value

The [AnnotationHub-class](#) query for the file you requested.

Author(s)

Leonardo Collado-Torres

See Also

[gencode_txdb\(\)](#) [gencode_annotated_genes\(\)](#) [gencode_genomic_state\(\)](#)

Examples

```
## Query AnnotationHub for the GenomicState object for Gencode v31 on
## hg19 coordinates
hub_query_gs_gencode_v31_hg19 <- GenomicStateHub(
  version = "31",
  genome = "hg19",
  filetype = "GenomicState"
)
hub_query_gs_gencode_v31_hg19

## Check the metadata
mcols(hub_query_gs_gencode_v31_hg19)

## Access the file through AnnotationHub
if (length(hub_query_gs_gencode_v31_hg19) == 1) {
  hub_gs_gencode_v31_hg19 <- hub_query_gs_gencode_v31_hg19[[1]]

  hub_gs_gencode_v31_hg19
}
```

`gencode_annotated_genes`

Annotation the genes for a given Gencode TxDb object

Description

Based on a TxDb object built by [gencode_txdb\(\)](#) this function annotates the genes. This information is then used by packages like `derfinder` and `derfinderPlot`.

Usage

```
gencode_annotated_genes(txdb)
```

Arguments

`txdb` A `GenomicFeatures::TxDb` object built with [gencode_txdb\(\)](#).

Value

The annotated genes resulting from `bumphunter:::annotateTranscripts()`.

Author(s)

Leonardo Collado-Torres.

References

Based on code for the brainflowprobes package at: https://github.com/LieberInstitute/brainflowprobes/blob/master/data-raw/create_sysdata.R

See Also

[gencode_txdb\(\)](#) [gencode_genomic_state\(\)](#)

Examples

```
## Start from scratch if you want:
## Not run:
txdb_v31_hg19_chr21 <- gencode_txdb("31", "hg19", chrs = "chr21")

## End(Not run)

## or read in the txdb object for hg19 chr21 from this package
txdb_v31_hg19_chr21 <- AnnotationDbi::loadDb(
  system.file("extdata", "txdb_v31_hg19_chr21.sqlite",
    package = "GenomicState"
  )
)

## Obtain the annotated genes for the Gencode TxDb object
genes_v31_hg19_chr21 <- gencode_annotated_genes(txdb_v31_hg19_chr21)

## Explore the result
genes_v31_hg19_chr21
```

`gencode_genomic_state` *Build a GenomicState object for Gencode TxDb objects*

Description

Based on a TxDb object built by [gencode_txdb\(\)](#) this function builds a GenomicState object which you can then use with [derfinder::annotateRegions\(\)](#). This information is then used by packages like [derfinderPlot](#).

Usage

```
gencode_genomic_state(txdb)
```

Arguments

`txdb` A [GenomicFeatures::TxDb](#) object built with [gencode_txdb\(\)](#).

Details

Note that not all genes will have symbols as many will be NA.

Value

A GenomicState object with the gene symbols as built using `derfinder::makeGenomicState()`.

Author(s)

Leonardo Collado-Torres

References

Based on code for the brainflowprobes package at: https://github.com/LieberInstitute/brainflowprobes/blob/master/data-raw/create_sysdata.R

See Also

`gencode_txdb()` `gencode_annotated_genes()`

Examples

```
## Start from scratch if you want:
## Not run:
txdb_v31_hg19_chr21 <- gencode_txdb("31", "hg19", chrs = "chr21")

## End(Not run)

## or read in the txdb object for hg19 chr21 from this package
txdb_v31_hg19_chr21 <- AnnotationDbi::loadDb(
  system.file("extdata", "txdb_v31_hg19_chr21.sqlite",
    package = "GenomicState"
  )
)

## Now build the GenomicState object
gs_v31_hg19_chr21 <- gencode_genomic_state(txdb_v31_hg19_chr21)

## Explore the result
gs_v31_hg19_chr21
```

gencode_txdb

Create a Gencode TxDb object

Description

This function builds a transcript database (TxDb) object which you can then use to build a Gencode GenomicState object. This function will download the data from Gencode, import it into R, process it and build the TxDb object.

Usage

```
gencode_txdb(
  version = "31",
  genome = c("hg38", "hg19"),
  chrs = paste0("chr", c(seq_len(22), "X", "Y", "M"))
)

gencode_source_url(version = "31", genome = c("hg38", "hg19"))
```

Arguments

| | |
|---------|--|
| version | A character(1) with the Gencode version number. |
| genome | A character(1) with the human genome version number. Valid options are 'hg38' or 'hg19'. |
| chrs | A character() vector with the chromosome (contig) names to keep. |

Value

A [GenomicFeatures::TxDb](#) object.
 A character(1) with the URL for the GTF Gencode file of interest.

Author(s)

Leonardo Collado-Torres

References

Based on code for the brainflowprobes package at: https://github.com/LieberInstitute/brainflowprobes/blob/master/data-raw/create_sysdata.R

See Also

[gencode_annotated_genes\(\)](#) [gencode_genomic_state\(\)](#)

Examples

```
## Start from scratch if you want:
## Not run:
txdb_v31_hg19_chr21 <- gencode_txdb("31", "hg19", chrs = "chr21")

## End(Not run)

## or read in the txdb object for hg19 chr21 from this package
txdb_v31_hg19_chr21 <- AnnotationDbi::loadDb(
  system.file("extdata", "txdb_v31_hg19_chr21.sqlite",
    package = "GenomicState"
  )
)
```

```
## Explore the result
txdb_v31_hg19_chr21

## Locate the GTF file for Gencode version 31 for hg19
gencode_source_url(version = "31", genome = "hg19")
```

| | |
|----------------|---|
| local_metadata | <i>Locate local metadata outside of AnnotationHub</i> |
|----------------|---|

Description

Locate local metadata outside of AnnotationHub

Usage

```
local_metadata(local_path = "/dcl01/lieber/ajaffe/lab/GenomicState/data-raw")
```

Arguments

local_path A character(1) pointing to where the data is stored locally

Value

The AnnotationHub metadata `data.frame()` for the data in this package with `RDataPath` updated to point to the `local_path`. It includes an additional column called `loadCode` which you can evaluate with `eval(parse(text = entry))`.

Author(s)

Leonardo Collado-Torres

See Also

[AnnotationHubData::makeAnnotationHubMetadata\(\)](#)

Examples

```
## Get the local metadata
meta <- local_metadata()

## Subset to the data of interest, lets say hg19 TxDb for v31
interest <- subset(meta, RDataClass == "TxDb" & Tags == "Gencode:v31:hg19")

## Inspect the result
interest

## Next you can load the data
if (file.exists(interest$RDataPath)) {
  ## This only works at JHPCE
```

```
eval(parse(text = interest$loadCode))

## Explore the loaded object (would be gencode_v31_hg19_txdb in this case)
gencode_v31_hg19_txdb
}
```

Index

AnnotationHub-class, [1](#)
AnnotationHubData::makeAnnotationHubMetadata(),
[6](#)

bumphunter::annotateTranscripts(), [2](#)

derfinder::annotateRegions(), [3](#)
derfinder::makeGenomicState(), [4](#)

gencode_annotated_genes, [2](#)
gencode_annotated_genes(), [1](#), [2](#), [4](#), [5](#)
gencode_genomic_state, [3](#)
gencode_genomic_state(), [1-3](#), [5](#)
gencode_source_url (gencode_txdb), [4](#)
gencode_txdb, [4](#)
gencode_txdb(), [1-4](#)
GenomicFeatures::TxDb, [2](#), [3](#), [5](#)
GenomicStateHub, [1](#)

local_metadata, [6](#)