

Package ‘tximeta’

April 16, 2019

Version 1.0.3

Title Transcript Quantification Import with Automatic Metadata

Description Transcript quantification import from Salmon with automatic population of metadata and transcript ranges. Filtered, combined, or de novo transcriptomes can be linked to the appropriate sources with linkedTxomes and shared for reproducible analyses.

Maintainer Michael Love <michaelisaiahlove@gmail.com>

License GPL-2

VignetteBuilder knitr

Imports SummarizedExperiment, tximport, jsonlite, S4Vectors, AnnotationDbi, GenomicFeatures, ensemblDb, BiocFileCache, tibble, GenomeInfoDb, rappdirs, utils, methods

Suggests knitr, rmarkdown, testthat, tximportData, org.Dm.eg.db, DESeq2, devtools

biocViews ImmunoOncology, Annotation, DataImport, Preprocessing, RNASeq, Transcriptomics, Transcription, GeneExpression

RoxygenNote 6.1.0

git_url <https://git.bioconductor.org/packages/tximeta>

git_branch RELEASE_3_8

git_last_commit dd036ee

git_last_commit_date 2019-01-11

Date/Publication 2019-04-15

Author Michael Love [aut, cre],
Rob Patro [aut, ctb],
Peter Hickey [aut, ctb],
Charlotte Sonesson [aut, ctb]

R topics documented:

addIds	2
getTximetaBFC	2
linkedTxome	3
summarizeToGene	4
tximeta	5

Index**7**

addIds	<i>Add IDs to rowRanges of a SummarizedExperiment</i>
--------	---

Description

For now this just works with SummarizedExperiments with Ensembl gene or transcript IDs. See example of usage in tximeta vignette.

Usage

```
addIds(se, column, gene = FALSE)
```

Arguments

se	the SummarizedExperiment
column	the name of the new ID to add (a column of the org database)
gene	logical, whether the rows are genes or transcripts (default is FALSE)

Value

a SummarizedExperiment

Examples

```
example(tximeta)
library(org.Dm.eg.db)
se <- addIds(se, "REFSEQ", gene=FALSE)
```

getTximetaBFC	<i>Get or set the directory of the BiocFileCache used by tximeta</i>
---------------	--

Description

Running getTximetaBFC will report the saved directory, if it has been determined, or will return NULL. Running setTximetaBFC will ask the user to specify a BiocFileCache directory for accessing and saving TxDb sqlite files.

Usage

```
getTximetaBFC()

setTximetaBFC(dir)
```

Arguments

dir	the location for tximeta's BiocFileCache. can be missing in which case the function will call file.choose for choosing location interactively
-----	---

Value

the directory of the BiocFileCache used by tximeta (or nothing, in the case of setTximetaBFC)

Examples

```
# getting the BiocFileCache used by tximeta
# (may not be set, which uses BiocFileCache default or temp directory)
getTximetaBFC()

# don't want to actually change user settings so this is not run:
# setTximetaBFC()
```

linkedTxome	<i>Make and load linked transcriptomes ("linkedTxome")</i>
-------------	--

Description

For now, for details please see the vignette `inst/script/linked.Rmd`

Usage

```
makeLinkedTxome(indexDir, source, organism, release, genome, fasta, gtf,
  write = TRUE, jsonFile)

loadLinkedTxome(jsonFile)
```

Arguments

<code>indexDir</code>	the path to the Salmon or Sailfish index
<code>source</code>	the source of transcriptome (e.g. "Gencode" or "Ensembl")
<code>organism</code>	organism (e.g. "Homo sapiens")
<code>release</code>	release number (e.g. "27")
<code>genome</code>	genome (e.g. "GRCh38")
<code>fasta</code>	FTP location for the FASTA sequence (of which the index is a subset)
<code>gtf</code>	FTP location for the GTF file (of which the index is a subset)
<code>write</code>	should a JSON file be written out which documents the transcriptome signature and metadata? (default is TRUE)
<code>jsonFile</code>	the path to the json file for the linkedTxome

Value

nothing, the function is run for its side effects

Examples

```
# point to a Salmon quantification file which combined two Ensembl FASTA files:
dir <- system.file("extdata/salmon_dm/SRR1197474", package="tximportData")
file <- file.path(dir, "quant.sf.gz")
coldata <- data.frame(files=file, names="SRR1197474", sample="1",
                      stringsAsFactors=FALSE)

# now point to the Salmon index itself to create a linkedTxome
# as the index will not match a known txome
dir <- system.file("extdata", package="tximeta")
indexDir <- file.path(dir, "Drosophila_melanogaster.BDGP6.v92_salmon_0.10.2")

# point to the source FASTA and GTF:
fastaFTP <- c("ftp://ftp.ensembl.org/pub/release-92/fasta/drosophila_melanogaster/cdna/Drosophila_melanogaster",
             "ftp://ftp.ensembl.org/pub/release-92/fasta/drosophila_melanogaster/ncrna/Drosophila_melanogaster")

# we comment this out for the example, and instead point to a local version
# usually one would point to the FTP source for the GTF file here
# gtfFTP <- "ftp://ftp.ensembl.org/pub/release-92/gtf/drosophila_melanogaster/Drosophila_melanogaster.BDGP6"

dir2 <- system.file("extdata/salmon_dm", package="tximportData")
gtfPath <- file.path(dir2, "Drosophila_melanogaster.BDGP6.92.gtf.gz")

# now create a linkedTxome, linking the Salmon index to its FASTA and GTF sources
makeLinkedTxome(indexDir=indexDir, source="Ensembl", organism="Drosophila melanogaster",
                release="92", genome="BDGP6", fasta=fastaFTP, gtf=gtfPath, write=FALSE)

# to clear the entire linkedTxome table
# (don't run unless you want to clear this table!)
# bfcloc <- getTximetaBFC()
# bfc <- BiocFileCache(bfcloc)
# bfcremove(bfc, bfcquery(bfc, "linkedTxomeTbl")$rid)
```

summarizeToGene

Summarize transcript-level quantifications to gene-level

Description

This function uses the tximport package to summarize counts, abundances and effective lengths from transcript-level to gene-level. It automatically will access the relevant TxDb (by either finding it in the BiocFileCache or by building it from an ftp location).

Usage

```
summarizeToGene(se, varReduce = FALSE, ...)
```

Arguments

se	a SummarizedExperiment produced by tximeta
varReduce	whether to reduce per-sample inferential replicates information into a matrix of sample variances variance (default FALSE)
...	arguments passed to tximport

Value

a SummarizedExperiment with summarized quantifications

Examples

```
example(tximeta)
gse <- summarizeToGene(se)
```

tximeta	<i>tximeta: Transcript quantification import with automatic metadata</i>
---------	--

Description

tximeta leverages the hash signature of the Salmon or Sailfish index, in addition to a number of core Bioconductor packages (GenomicFeatures, ensemblDb, GenomeInfoDb, BiocFileCache) to automatically populate metadata for the user, without additional effort from the user. Note that this package is in "beta" / under development.

Usage

```
tximeta(coldata, type = "salmon", skipMeta = FALSE, ...)
```

Arguments

coldata	a data.frame with at least two columns (others will propagate to object): <ul style="list-style-type: none"> files - character, paths of quantification files names - character, sample names if coldata is a vector, it is assumed to be the paths of quantification files and unique sample names are created
type	what quantifier was used (see tximport)
skipMeta	whether to skip metadata generation (e.g. to avoid errors if not connected to internet). This calls tximport directly and so either txOut=TRUE or tx2gene should be specified.
...	arguments passed to tximport

Details

Most of the code in tximeta works to add metadata and transcript ranges when the quantification was performed with Salmon or Sailfish. However, tximeta can be used with any quantification type that is supported by [tximport](#), where it will return an un-ranged SummarizedExperiment.

tximeta checks the hash signature of the index against a database of known transcriptomes (this database under construction) or a locally stored linkedTxome (see [link{makeLinkedTxome}](#)), and then will automatically populate, e.g. the transcript locations, the transcriptome release, the genome with correct chromosome lengths, etc. It allows for automatic and correct summarization of transcript-level quantifications to the gene-level via [summarizeToGene](#) without the need to manually build a tx2gene table.

tximeta on the first run will ask where the BiocFileCache for this package should be kept, either using a default location or a temporary directory. At any point, the user can specify a location using `setTximetaBFC` and this choice will be saved for future sessions. Multiple users can point to the same BiocFileCache, such that transcript databases (TxDb) associated with certain Salmon or Sailfish indices and linkedTxomes can be accessed by different users without additional effort or time spent downloading/building the relevant TxDb.

In order to allow that multiple users can read and write to the same location, one should set the BiocFileCache directory to have group write permissions (g+w).

Value

a SummarizedExperiment with metadata on the rowRanges. (if the hash signature in the Salmon or Sailfish index does not match any known transcriptomes, or any locally saved linkedTxome, tximeta will just return a non-ranged SummarizedExperiment)

Examples

```
# point to a Salmon quantification file:
dir <- system.file("extdata/salmon_dm", package="tximportData")
files <- file.path(dir, "SRR1197474_cdna", "quant.sf.gz")
coldata <- data.frame(files, names="SRR1197474", condition="A", stringsAsFactors=FALSE)

# normally we would just run the following which would download the appropriate metadata
# se <- tximeta(coldata)

# for this example, we instead point to a local path where the GTF can be found
# by making a linkedTxome:
dir <- system.file("extdata", package="tximeta")
indexDir <- file.path(dir, "Drosophila_melanogaster.BDGP6_cdna.v92_salmon_0.10.2")
fastaFTP <- "ftp://ftp.ensembl.org/pub/release-92/fasta/drosophila_melanogaster/cdna/Drosophila_melanogaster"
dir2 <- system.file("extdata/salmon_dm", package="tximportData")
gtfPath <- file.path(dir2, "Drosophila_melanogaster.BDGP6.92.gtf.gz")
makeLinkedTxome(indexDir=indexDir, source="Ensembl", organism="Drosophila melanogaster",
                release="92", genome="BDGP6", fasta=fastaFTP, gtf=gtfPath, write=FALSE)
se <- tximeta(coldata)

# to clear the entire linkedTxome table
# (don't run unless you want to clear this table!)
# bfcloc <- getTximetaBFC()
# bfc <- BiocFileCache(bfcloc)
# bfcremove(bfc, bfcquery(bfc, "linkedTxomeTbl")$rid)
```

Index

`addIds`, 2

`getTximetaBFC`, 2

`linkedTxome`, 3

`loadLinkedTxome (linkedTxome)`, 3

`makeLinkedTxome (linkedTxome)`, 3

`setTximetaBFC`, 6

`setTximetaBFC (getTximetaBFC)`, 2

`summarizeToGene`, 4, 5

`tximeta`, 5

`tximport`, 5