Package 'tximport'

April 15, 2017

Version 1.2.0

Title Import and summarize transcript-level estimates for gene-level analysis

Description Imports transcript-level abundance, estimated counts and transcript lengths, and summarizes into matrices for use with downstream gene-level analysis packages. Average transcript length, weighted by sample-specific transcript abundance estimates, is provided as a matrix which can be used as an offset for different expression of gene-level counts.

Author Michael Love, Charlotte Soneson, Mark Robinson

Maintainer Michael Love <michaelisaiahlove@gmail.com>

License GPL (>=2)

VignetteBuilder knitr

Imports utils

Suggests knitr, testthat, tximportData, TxDb.Hsapiens.UCSC.hg19.knownGene, readr (>= 0.2.2), limma, edgeR, DESeq2 (>= 1.11.6)

biocViews RNASeq, Transcription, GeneExpression, DataImport

RoxygenNote 5.0.1

NeedsCompilation no

R topics documented:

`tximport` *Import transcript-level abundances and estimated counts for gene-level analysis packages*

Description

`tximport` imports transcript-level estimates from various external software and optionally summarizes abundances, counts, and transcript lengths to the gene-level (default) or outputs transcript-level matrices (see `txOut` argument). While `tximport` summarizes to the gene-level by default, the user can also perform the import and summarization steps manually, by specifing `txOut=TRUE` and then using the function `summarizeToGene`. Note however that this is equivalent to `tximport` with `txOut=FALSE` (the default).

Usage

```
tximport(files, type = c("none", "kallisto", "salmon", "sailfish", "rsem"),
  txIn = TRUE, txOut = FALSE, countsFromAbundance = c("no", "scaledTPM",
  "lengthScaledTPM"), tx2gene = NULL, reader = read.delim, geneIdCol,
  txIdCol, abundanceCol, countsCol, lengthCol, importer, collatedFiles,
  ignoreTxVersion = FALSE)

summarizeToGene(txi, tx2gene, ignoreTxVersion = FALSE,
  countsFromAbundance = c("no", "scaledTPM", "lengthScaledTPM"))
```

Arguments

`files`	a character vector of filenames for the transcript-level abundances
`type`	character, the type of software used to generate the abundances. Options are "kallisto", "salmon", "sailfish", "rsem". This argument is used to autofill the arguments below (geneIdCol, etc.) "none" means that the user will specify these columns.
`txIn`	logical, whether the incoming files are transcript level (default TRUE)
`txOut`	logical, whether the function should just output transcript-level (default FALSE)
`countsFromAbundance`	character, either "no" (default), "scaledTPM", or "lengthScaledTPM", for whether to generate estimated counts using abundance estimates scaled up to library size (scaledTPM) or additionally scaled using the average transcript length over samples and the library size (lengthScaledTPM). if using scaledTPM or lengthScaledTPM, then the counts are no longer correlated with average transcript length, and so the length offset matrix should not be used.
`tx2gene`	a two-column data.frame linking transcript id (column 1) to gene id (column 2). the column names are not relevant, but this column order must be used. this argument is required for gene-level summarization for methods that provides transcript-level estimates only (kallisto, Salmon, Sailfish)
`reader`	a function to replace read.delim in the pre-set importer functions, for example substituting read_tsv from the readr package will substantially speed up tximport
`geneIdCol`	name of column with gene id. if missing, the gene2tx argument can be used
`txIdCol`	name of column with tx id
`abundanceCol`	name of column with abundances (e.g. TPM or FPKM)
`countsCol`	name of column with estimated counts
`lengthCol`	name of column with feature length information
`importer`	a function used to read in the files

`collatedFiles` a character vector of filenames for software which provides abundances and counts in matrix form (e.g. Cufflinks). The files should be, in order, abundances, counts, and a third file with length information

`ignoreTxVersion` logical, whether to split the tx id on the '.' character to remove version information, for easier matching with the tx id in gene2tx (default FALSE)

`txi` list of matrices of trancript-level abundances, counts, and lengths produced by `tximport`, only used by `summarizeToGene`

Details

Solutions to the error "tximport failed at summarizing to the gene-level":

1. provide a `tx2gene` data.frame linking transcripts to genes (more below)
2. avoid gene-level summarization by specifying `txOut=TRUE`
3. set `geneIdCol` to an appropriate column in the files

See `vignette('tximport')` for example code for generating a `tx2gene` data.frame from a `TxDb` object. Note that the `keys` and `select` functions used to create the `tx2gene` object are documented in the man page for AnnotationDb-class objects in the AnnotationDbi package (TxDb inherits from AnnotationDb). For further details on generating TxDb objects from various inputs see `vignette('GenomicFeatures')` from the GenomicFeatures package.

Version support: The last known supported versions of the external quantifiers are: kallisto 0.42.4, Salmon 0.6.0, Sailfish 0.9.0, RSEM 1.2.11.

Value

a simple list with matrices: abundance, counts, length. A final element 'countsFromAbundance' carries through the character argument used in the tximport call. The length matrix contains the average transcript length for each gene which can be used as an offset for gene-level analysis. Note: tximport does not import bootstrap estimates from kallisto, Salmon, or Sailfish.

Functions

- `tximport`: Import tx-level quantifications and summarize abundances, counts and lengths to gene-level (default) or simply output tx-level matrices
- `summarizeToGene`: Summarize tx-level matrices to gene-level

References

Charlotte Soneson, Michael I. Love, Mark D. Robinson (2015): Differential analyses for RNA-seq: transcript-level estimates improve gene-level inferences. F1000Research. http://dx.doi.org/10.12688/f1000research.7563.1

Examples

```
# load data for demonstrating tximport
# note that the vignette shows more examples
# including how to read in files quickly using the readr package

library(tximportData)
dir <- system.file("extdata", package="tximportData")
```

```
samples <- read.table(file.path(dir,"samples.txt"), header=TRUE)
files <- file.path(dir,"salmon", samples$run, "quant.sf")
names(files) <- paste0("sample",1:6)

# tx2gene links transcript IDs to gene IDs for summarization
tx2gene <- read.csv(file.path(dir, "tx2gene.csv"))

txi <- tximport(files, type="salmon", tx2gene=tx2gene)
```

Index