Package 'tximport'

April 16, 2019

Version 1.10.1

Title Import and summarize transcript-level estimates for transcript- and 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 [cre,aut], Charlotte Soneson [aut], Mark Robinson [aut], Rob Patro [ctb], Andrew Parker Morgan [ctb], Ryan C. Thompson [ctb], Matt Shirley [ctb]

Maintainer Michael Love <michaelisaiahlove@gmail.com>

License GPL (>=2)

VignetteBuilder knitr

Imports utils, stats

Suggests knitr, rmarkdown, testthat, tximportData, TxDb.Hsapiens.UCSC.hg19.knownGene, readr (>= 0.2.2), limma, edgeR, DESeq2 (>= 1.11.6), rhdf5, jsonlite, matrixStats

biocViews ImmunoOncology, RNASeq, Transcription, GeneExpression, DataImport

RoxygenNote 6.0.1

NeedsCompilation no

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

git_branch RELEASE_3_8

git_last_commit cd8f81c

git_last_commit_date 2019-01-04

Date/Publication 2019-04-15

R topics documented:

`makeCountsFromAbundance` *Low-level function to make counts from abundance using matrices*

Description

Simple low-level function used within tximport to generate `scaledTPM` or `lengthScaledTPM` counts, taking as input the original counts, abundance and length matrices. NOTE: This is a low-level function exported in case it is needed for some reason, but the recommended way to generate counts-from-abundance is using tximport with the `countsFromAbundance` argument.

Usage

```
makeCountsFromAbundance(countsMat, abundanceMat, lengthMat,
  countsFromAbundance = c("scaledTPM", "lengthScaledTPM"))
```

Arguments

`countsMat`	a matrix of original counts
`abundanceMat`	a matrix of abundances (typically TPM)
`lengthMat`	a matrix of effective lengths
`countsFromAbundance`	the desired type of count-from-abundance output

Value

a matrix of count-scale data generated from abundances. for details on the calculation see tximport.

`tximport` *Import transcript-level abundances and counts for transcript- and 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).

Usage

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

tximport(files, type = c("none", "salmon", "sailfish", "kallisto", "rsem",
  "stringtie"), txIn = TRUE, txOut = FALSE, countsFromAbundance = c("no",
  "scaledTPM", "lengthScaledTPM", "dtuScaledTPM"), tx2gene = NULL,
  varReduce = FALSE, dropInfReps = FALSE, infRepStat = NULL,
  ignoreTxVersion = FALSE, ignoreAfterBar = FALSE, geneIdCol, txIdCol,
  abundanceCol, countsCol, lengthCol, importer = NULL,
  existenceOptional = FALSE, readLength = 75)
```

Arguments

txi	list of matrices of trancript-level abundances, counts, and lengths produced by tximport, only used by summarizeToGene
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)
varReduce	whether to reduce per-sample inferential replicates information into a matrix of sample variances variance (default FALSE)
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)
ignoreAfterBar	logical, whether to split the tx id on the 'l' character (default FALSE)
countsFromAbundance	character, either "no" (default), "scaledTPM", "lengthScaledTPM", or "dtuScaledTPM". Whether to generate estimated counts using abundance estimates: • scaled up to library size (scaledTPM), • scaled using the average transcript length over samples and then the library size (lengthScaledTPM), or • scaled using the median transcript length among isoforms of a gene, and then the library size (dtuScaledTPM). dtuScaledTPM is designed for DTU analysis in combination with txOut=TRUE, and it requires specifing a tx2gene data.frame. dtuScaledTPM works such that within a gene, values from all samples and all transcripts get scaled by the same fixed median transcript length. If using scaledTPM, lengthScaledTPM, or geneLengthScaledTPM, the counts are no longer correlated across samples with transcript length, and so the length offset matrix should not be used.
files	a character vector of filenames for the transcript-level abundances
type	character, the type of software used to generate the abundances. Options are "salmon", "sailfish", "kallisto", "rsem", "stringtie", or "none". 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)
dropInfReps	whether to skip reading in inferential replicates (default FALSE)
infRepStat	a function to re-compute counts and abundances from the inferential replicates, e.g. matrixStats::rowMedians to re-compute counts as the median of the inferential replicates. The order of operations is: first counts are re-computed, then abundances are re-computed. Following this, if countsFromAbundance is not "no", tximport will again re-compute counts from the re-computed abundances. infRepStat should operate on rows of a matrix. (default is NULL)
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
existenceOptional	logical, should tximport not check if files exist before attempting import (default FALSE, meaning files must exist according to file.exists)
readLength	numeric, the read length used to calculate counts from StringTie's output of coverage. Default value (from StringTie) is 75. The formula used to calculate counts is: cov * transcript length / read length

Details

tximport will also load in information about inferential replicates – a list of matrices of the Gibbs samples from the posterior, or bootstrap replicates, per sample – if these data are available in the expected locations relative to the files. The inferential replicates, stored in infReps in the output list, are on estimated counts, and therefore follow counts in the output list. By setting varReduce=TRUE, the inferential replicate matrices will be replaced by a single matrix with the sample variance per transcript/gene and per sample.

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).

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.

Value

a simple list containing matrices: abundance, counts, length. Another list element 'countsFromAbundance' carries through the character argument used in the tximport call. If detected, and txOut=TRUE, inferential replicates for each sample will be imported and stored as a list of matrices, itself an element infReps in the returned list. If varReduce=TRUE the inferential replicates will be summarized according to the sample variance, and stored as a matrix variance. The length matrix contains the average transcript length for each gene which can be used as an offset for gene-level analysis.

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.gz")
names(files) <- paste0("sample",1:6)

# tx2gene links transcript IDs to gene IDs for summarization
tx2gene <- read.csv(file.path(dir, "tx2gene.gencode.v27.csv"))

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

Index