Package ‘TEKRABber’

May 30, 2024

Type  Package

Title  An R package estimates the correlations of orthologs and transposable elements between two species

Version  1.8.0

Description  TEKRABber is made to provide a user-friendly pipeline for comparing orthologs and transposable elements (TEs) between two species. It considers the orthology confidence between two species from BioMart to normalize expression counts and detect differentially expressed orthologs/TEs. Then it provides one to one correlation analysis for desired orthologs and TEs. There is also an app function to have a first insight on the result. Users can prepare orthologs/TEs RNA-seq expression data by their own preference to run TEKRABber following the data structure mentioned in the vignettes.

URL  https://github.com/ferygood/TEKRABber

BugReports  https://github.com/ferygood/TEKRABber/issues

Encoding  UTF-8

License  LGPL (>=3)

Imports  apeglm, biomaRt, dplyr, doParallel, DESeq2, foreach, GenomeInfoDb, magrittr, Rcpp (>= 1.0.7), rtracklayer, SCBN, stats, utils

LinkingTo  Rcpp

Depends  R (>= 4.3)

LazyData  false

Suggests  BiocStyle, bslib, ggplot2, ggpubr, plotly, rmarkdown, shiny, knitr, testthat (>= 3.0.0)

VignetteBuilder  knitr

VignetteEngine  knitr

RoxygenNote  7.3.1

biocViews  DifferentialExpression, Normalization, Transcription, GeneExpression

git_url  https://git.bioconductor.org/packages/TEKRABber
Description

Provide a shiny UI for presenting the results from DE analysis and correlation analysis.

Usage

appTEKRABber(corrRef, corrCompare, DEobject)

Arguments

corrRef correlation results for reference using corrOrthologScale()
corrCompare correlation results for comparison using corrOrthologScale()
DEobject DE object using DEgeneTE()

Value

provide an interactive shinyapp
Examples

data(speciesCounts)
hmGene <- speciesCounts$hmGene
hmTE <- speciesCounts$hmTE
chimpGene <- speciesCounts$chimpGene
chimpTE <- speciesCounts$chimpTE
data(fetchDataHmChimp)
fetchData <- fetchDataHmChimp
inputBundle <- DECorrInputs(fetchData)

meta <- data.frame(
species = c(rep("human", ncol(hmGene) - 1),
            rep("chimpanzee", ncol(chimpGene) - 1)))

meta$species <- factor(meta$species, levels = c("human", "chimpanzee"))
rownames(meta) <- colnames(inputBundle$geneInputDESeq2)
hmchimpDE <- DEgeneTE(
    geneTable = inputBundle$geneInputDESeq2,
    teTable = inputBundle$teInputDESeq2,
    metadata = meta,
    expDesign = TRUE)

hmCorrResult <- corrOrthologTE(
    geneInput = hmchimpDE$geneCorrInputRef[c(1:10),],
    teInput = hmchimpDE$teCorrInputRef[c(1:10),],
    corrMethod = "pearson",
    padjMethod = "fdr")

chimpCorrResult <- corrOrthologTE(
    geneInput = hmchimpDE$geneCorrInputCompare[c(1:10),],
    teInput = hmchimpDE$teCorrInputCompare[c(1:10),],
    corrMethod = "pearson",
    padjMethod = "fdr")

#library(plotly)
#appTEKRABber(
#    corrRef = hmCorrResult,
#    corrCompare = chimpCorrResult,
#    DEobject = hmchimpDE)

corrOrthologTE

Estimate correlation comparing orthologs and TEs

Description

To estimate correlation comparing orthologs and TEs one-by-one from inputs. You can specify the correlation and adjusted p-value methods (see details in parameters). If you want to save your
outputs instead of just returning them, please specify the fileDir and fileName with the extension .csv. The default fileName is TEKRABber_geneTECorrReusult.csv.

Usage

corrOrthologTE(geneInput, teInput, corrMethod = "pearson", padjMethod = "fdr", numCore=1, fileDir=NULL, fileName="TEKRABber_geneTECorrReusult.csv")

Arguments

geneInput gene count input for correlation from using DECorrInputs()
teInput te count input for correlation from using DECorrInputs()
corrMethod correlation method, including pearson, kendall, spearman. Default is pearson.
padjMethod method to return adjusted p-value, and default is fdr. See ?p.adjust
numCore number of cores to run parallel. Default is 1. You can use detectCores() to get how many cores you can use.
fileDir the name of directory for saving output files. Default is NULL.
fileName the name for saving output files. Default is "TEKRABber_geneTECorrReusult.csv"

Value

a dataframe includes correlation coefficient, pvalue, padj

Examples

data(ctInputDE)
geneInputDE <- ctInputDE$gene
teInputDE <- ctInputDE$te

metaExp <- data.frame(experiment = c(rep("control", 5), rep("treatment", 5)))
rownames(metaExp) <- colnames(geneInputDE)
metaExp$experiment <- factor(
    metaExp$experiment,
    levels = c("control", "treatment")
)

resultDE <- DEgeneTE(
    geneTable = geneInputDE,
    teTable = teInputDE,
    metadata = metaExp,
    expDesign = FALSE
)

controlCorr <- corrOrthologTE(
    geneInput = resultDE$geneCorrInputRef[c(1:10),],
    teInput = resultDE$teCorrInputRef[c(1:10),],
    numCore = 1,
    corrMethod = "pearson",
    padjMethod = "fdr"
**ctInputDE**

Input expression data of gene/TE for differentially expressed analysis within same species

**Description**

TEKRABber can also be used comparing orthologs and transposable elements within same species, i.e., control and treatment. Here we provide an example data for demonstration. This data was based on syn8466812 RNA-seq (Allen M et al., 2016). However, the expression data was modified due to confidential agreement. Therefore, it cannot represent the original data.

**Usage**

```r
data(ctInputDE)
```

**Format**

An object contains 2 expression data:

- **gene**: input gene data for DE analysis comparing control and treatment
- **te**: input TE data for DE analysis comparing control and treatment

**Examples**

```r
data(ctInputDE)
geneInputDE <- ctInputDE$gene
teInputDE <- ctInputDE$te
```

**DECorrInputs**

Generate all the input files for TEKRABber downstream analysis

**Description**

Generate all the inputs files for differentially expressed orthologous genes/TEs analysis, and for correlation analysis. The output is a list containing 6 dataframes.

**Usage**

```r
DECorrInputs(fetchData)
```

**Arguments**

- `fetchData`: output list from TEKRABber::orthologScale()
Value

create inputs for DE analysis and correlations: (1) geneInputDESeq2 (2) teInputDESeq2 (3) geneCorrInputRef (4) geneCorrInputCompare (5) TECorrInputRef (6) TECorrInputCompare

Examples

data(speciesCounts)
data(hg38_panTro6_rmsk)
hmGene <- speciesCounts$hmGene
chimpGene <- speciesCounts$chimpGene
hmTE <- speciesCounts$hmTE
chimpTE <- speciesCounts$chimpTE

## For demonstration, here we only select 1000 rows to save time
set.seed(1234)
hmGeneSample <- hmGene[sample(nrow(hmGene), 1000),]
chimpGeneSample <- chimpGene[sample(nrow(chimpGene), 1000),]

fetchData <- orthologScale(
  speciesRef = "hsapiens",
  speciesCompare = "ptroglodytes",
  geneCountRef = hmGeneSample,
  geneCountCompare = chimpGeneSample,
  teCountRef = hmTE,
  teCountCompare = chimpTE,
  rmsk = hg38_panTro6_rmsk,
  version = 105
)

inputBundle <- DECorrInputs(fetchData)

DEgeneTE

Estimate differentially expressed genes and TEs

Description

To estimate differentially expressed genes and TEs, DEgeneTE() takes gene inputs and TE inputs from the results using the DECorrInputs function. You need to specify your metadata and expDesign based on your design. If you also want to save the output, please specify the fileDir parameter.

Usage

DEgeneTE(geneTable, teTable, metadata, expDesign=TRUE, fileDir=NULL)

Arguments

geneTable        gene input table from using DECorrInputs()
teTable          TE input table from using DECorrInputs()
metadata an one column dataframe with rownames same as the column name of gene/te count table. Column name must be `species` or `experiment`.

expDesign Logic value for comparing between or within species. `TRUE` for comparing between two species, and `FALSE` for comparing between control and treatment.

fileDir the name and path of directory for saving output files. Default is NULL.

Value

return DESeq2 res and normalized gene counts.

Examples

```r
## comparing between species:
## (1) set expDesign = TRUE
## (2) column name of metadata needs to be "species".

data(fetchDataHmChimp)
fetchData <- fetchDataHmChimp
inputBundle <- DECorrInputs(fetchData)

meta <- data.frame(species=c(rep("human", ncol(fetchData$geneRef) - 1),
                   rep("chimpanzee", ncol(fetchData$geneCompare) - 1)))
rownames(meta) <- colnames(inputBundle$geneInputDESeq2)
meta$species <- factor(meta$species, levels = c("human", "chimpanzee"))

hmchimpDE <- DEgeneTE(
  geneTable = inputBundle$geneInputDESeq2,
  teTable = inputBundle$teInputDESeq2,
  metadata = meta,
  expDesign = TRUE
)
```

fetchDataHmChimp Example output comparing human and chimpanzee data using orthologScale()

Description

An output list of data contains 7 elements after using orthologScale(), including (1) orthology table comparing human and chimpanzee. (2) scaling factor for orthologous genes (3) gene count table from reference species (4) gene count table from species you want to compare (5) scaling factor for TEs (6) TE count table from reference species (7) TE count table from the species you want to compare. The aim to provide this dataset is to save time for user running the vignettes and give a template for demonstration.
Usage

data(fetchDataHmChimp)

Format

An object contains 2 elements:

- **orthologTable**  orthology information from Ensembl
- **scaleFactor**  scaling factor to normalize data

Examples

data(fetchDataHmChimp)
fetchData <- fetchDataHmChimp
fetchData$orthologTable
fetchData$scaleFactor

---

**hg38_panTro6_rmsk**  Repeatmasker track annotations with human and chimpanzee

---

Description

This Repeatmasker track annotations table was first downloaded from UCSC Genome Table Browser and it included the name, class, and average gene length in repeats (transposable elements). This data is used for demonstrate an example for user how to provide a annotation table to normalize their data which in this case comparing human(hg38) to chimpanzee(panTro6).

Usage

data(hg38_panTro6_rmsk)

Format

An object of class grouped_df (inherits from tbl_df, tbl, data.frame) with 12550 rows and 4 columns.

Examples

data(hg38_panTro6_rmsk)
Normalized orthologous genes and TEs between two species

Description

Normalize orthologous genes and TEs between two species with a scaling factor using their expression level and gene lengths.

Usage

orthologScale(speciesRef, speciesCompare, geneCountRef, geneCountCompare, teCountRef, teCountCompare, rmsk, version)

Arguments

speciesRef The scientific name for your reference species. i.e., hsapiens
speciesCompare The scientific name for your species to compare. i.e., ptroglodytes
geneCountRef Gene count from your reference species. First column should be Ensembl gene ID.
geneCountCompare Gene count from the species you want to compare. First column should be Ensembl gene ID.
teCountRef TE count from your reference species. First column should be teName.
teCountCompare TE count from the species you want to compare. First column should be teName.
rmsk a repeatmasker table including 4 columns: (1) the name of TE (2) the class of TE (3) The average length of that TE from your reference species (4) The average length of that TE from the species you want to compare.
version for specify Ensembl version. Default is NULL for getting the latest version

Value

a list of outputs: (1) orthologTable, orthology information (2) c_ortholog, scaling factor for orthologous genes (3) geneRef, gene count table for reference species (4) geneCompare, normalized gene count table for species compared (5) c_te, scaling factor for TEs (6) teRef, TE count table for reference species (7) teCompare, normalized TE count table for species compared.

Examples

data(speciesCounts)
data(hg38_panTro6_rmsk)
hmGene <- speciesCounts$hmGene
chimpGene <- speciesCounts$chimpGene
hmTE <- speciesCounts$hmTE
chimpTE <- speciesCounts$chimpTE

## For demonstration, here we only select 1000 rows to save time
prepareRMSK

Prepare a table from two species RepeatMakser track from UCSC genome Table

Description

create a table to the rmsk argument in orthologScale(). Before version 1.8, TEKRABber requires user to prepare this table by themselves and this function can help user automatically get the RepeatMasker table from UCSC. The arguments required are the abbreviation of the version of reference (case-sensitive). For example, "hg38" for human. Note: currently only 91 genomes provided. Check if the reference exists with GenomeInfoDb::registered_UCSC_genomes().

Usage

prepareRMSK(refSpecies, compareSpecies)

Arguments

refSpecies the version of reference species, i.e. hg38
compareSpecies the version of compared species, i.e. panTro6

Value

Dataframe with four columns: repName, repClass, rLen and cLen

Examples

df_rmsk <- prepareRMSK(refSpecies = "hg38", compareSpecies = "panTro6")
rcpp_corr

Estimate the correlation between genes and transposable elements

Description

Estimate the correlation between genes and transposable elements

Usage

rcpp_corr(df1, df2, Method)

Arguments

df1  First dataframe
df2  Second dataframe
Method  correlation method

Value

a dataframe containing correlation results

speciesCounts

Gene/TE expression data from human/chimpanzee brain RNA-seq

Description

Dataset contains 4 expression data from human and chimpanzee brain RNA-seq. We select raw fastq data from 10 humans and 10 chimpanzees from (Khrameeva E et al., 2020). Gene expression is generated using HISAT2 and featureCounts (Kim D et al., 2019; Liao Y et al., 2014). Transposable elements (TEs) expression is generated with multi-mapping option using STAR and TEtranscripts (Dobin A et al., 2013; Jin Y et al., 2015).

Usage

data(speciesCounts)

Format

An object contains 4 expression counts:

hmGene  human gene expression data
hmTE  human TE expression
cchimpGene  chimpanzee gene expression data
cchimpTE  chimpanzee TE expression data
Examples

data(speciesCounts)
hmGene <- speciesCounts$hmmGene
hmTE <- speciesCounts$hmmTE
chimpGene <- speciesCounts$chimpGene
chimpTE <- speciesCounts$chimpTE

TEKRABber

An R package estimates the correlations of orthologs and transposable elements between two species

Description

TEKRABber is made to provide an user-friendly pipeline for comparing orthologs and transposable elements (TEs) between two species. It considers the orthology confidence between two species from BioMart to normalize expression counts and detect differentially expressed ortholog/TEs. Then it provides one to one correlation analysis for desired orthologs and TEs. There is also an app function to have a first insight on the result. Users can prepare orthologs/TEs RNA-seq expression data by their own preference to run TEKRABber following the data structure mentioned in the vignettes.

Details

TEKRABber analysis pipeline includes 5 main functions:
1. orthologScale(): obtain orthology information and calculate scaling factor. 2. DECorrInputs(): create the input files for running DE/correlation analysis. 3. DEgeneTE(): run DE analysis on orthologs and transposable elements. 4. corrOrthologTE(): estimate correlation between selected orthologs and transposable elements. 5. appTEKRABber(): (optional) find first insight from data using an local webapp. Find more details in vignette or on the helping page, i.e. ?orthologScale

Author(s)

Yao-Chung Chen, Katja Nowick.
Maintainer: Yao-Chung Chen <yao-chung.chen@fu-berlin.de>

TEKRABber GitHub Repo
Index

* datasets
  ctInputDE, 5
  fetchDataHmChimp, 7
  hg38_panTro6_rmsk, 8
  speciesCounts, 11

  appTEKRABber, 2
  corrOrthologTE, 3
  ctInputDE, 5
  DECorrInputs, 5
  DEgeneTE, 6

  fetchDataHmChimp, 7
  hg38_panTro6_rmsk, 8
  orthologScale, 9
  prepareRMSK, 10
  rcpp_corr, 11
  speciesCounts, 11
  TEKRABber, 12