Package 'dynOmics'

June 28, 2018

Type Package

Title Fast Fourier Transform to Identify Associations Between Time Course Omics Data

Version 1.2

Date 2018-06-12

Depends R (>= 3.0.0), ggplot2

Imports methods, parallel, gplots

Author Jasmin Straube [aut, cre], Kim-Anh Le Cao [aut], Emma Huang [aut], Anne Bernard [ctb]

Maintainer Jasmin Straube <jasmin.straube@qimrberghofer.edu.au>

Description Implements a method based on the fast Fourier transform to estimate delays of expression initiation between trajectories to integrate and analyse time course omics data.

License GPL (>= 2) | file LICENSE

VignetteBuilder knitr

Suggests knitr, lmms, nlme, testthat, snow

NeedsCompilation no

LazyData true

RoxygenNote 6.0.1

Repository CRAN

Date/Publication 2018-06-28 10:41:01 UTC

R topics documented:

`dynOmics-package` *Fast Fourier transform to estimate delays in expression initiation to identify associations between time course 'omics' data.*

Description

The package provides functions to identify associations within one or between two time course 'omics' data and visualise the associations : associateData to estimate the delays and identify associations of data sets containing time course 'omics' experiments; plot.associations: to visualise associated profiles.

Details

Package:	dynOmics
Type:	Package
Version:	1.2
Date:	2018-06-12
License:	GPL-2
LazyLoad:	yes

Functions for associating data: `associateData`
Functions for summarization: `summary.associations`
Functions for plots: `plot.associations`

Author(s)

Jasmin Straube with contributions from Kim-Anh Le Cao, Emma Huang and Anne Bernard

Maintainer: Jasmin Straube <j.straube@qimrberghofer.edu.au>

`associateData` *Identify associations of trajectories within a data set or across two data sets*

Description

Function to estimate differences in expression initation of trajectories to identify associations between time course 'omics' data.

Usage

```
associateData(data1,data2,numCores)
```

Arguments

data1	data.frame or matrix containing the time as rows and features as columns
data2	optional an additional data.frame or matrix containing the time as rows and features as columns
numCores	alternative numeric value indicating the number of CPU cores to be used for parallelization. Default value is automatically estimated.

Details

associateData() takes as input two data sets of interest and performs a pairwise associations comparison between features using a fast Fourier transform approach to detect delays (also called 'associations') between the different features. Note that the argument 'numCores' indicates the number of CPUs and is detected by default in the function to perform parallelization. The final result is a table with a row for each pairwise comparison. The output presents the dynOmics estimated delay between two features, the p-value ('p') and correlation coefficient ('cor') from a Pearson's test, before and after the time profiles have been realigned according to the dynOmics estimated delay.

Value

associateData returns an object of class associations containing the following components:

- Feature1 character the colnames or the index of data1.
- Feature2 character the colnames or the index of data2.
- delay numeric estimated delay between feature1 and feature2.
- pBefore numeric p-value of the test for association before applying the predicted time shift.
- pAfter numeric p-value of the test for association after applying the predicted time shift.
- corBefore numeric Pearson correlation before applying the predicted time shift.
- corAfter numeric Pearson correlation after applying the predicted time shift.

References

Straube J., Bernard A., Huang B.E., Le Cao K.-A.(2017). *DynOmics to identify delays and co-expression patterns across time course experiments* Scientific Reports

See Also

summary.associations, plot.associations

Examples

```
## Not run:
data(Metabolites)
data(Transcripts)
associations <- associateData(Metabolites[,1],Transcripts[,c(1:50)])
#summary(associations)
#plot(associations,Metabolites,Transcripts,feature1=1)

## End(Not run)
```

Metabolites *Metabolite and Transcript Simulation Data*

Description

Simulated data were received from Redestig et al., 2011. Metabolite and transcript levels were obtained using an impulse model (Chechik and Koller, 2009). Functions were used to model five different metabolite patterns and for each metabolite 50 associated transcript levels. Time lags were introduced in the range from -2 to 2 with the probability 0.1, 0.2, 0.4, 0.2, 0.1. Simulated profiles have seven time points and normal distributed noise was introduced with mean zero and standard deviation 0.1.

Usage

```
data(Metabolites)
```

Format

This data set contains the simulated expression of 5 metabolites for 7 time points.

Details

- Metabolites. data matrix with 7 rows and 5 columns. Each row represents an experimental time sample, and each column a single metabolite.

Source

The Metabolite Simulation Data is based on the the paper of Redestig *et al.* (2011).

References

Redestig,H. and Costa,I.G. Detection and interpretation of metabolite-transcript coresponses using combined profiling data. *Bioinformatics* **27**(13) (2011), pp. i357 65.

plot.associations *Plot of* associations *objects*

Description

Plot showing the associated trajectories with or without estimated time shift.

Usage

```
## S3 method for class 'associations'
plot(x, data1, data2, time, feature1, feature2, cutoff,
  fdr = T, absCor = T, withShift = F, ...)
```

Arguments

x	an object of class associations
data1	an object of class matrix or data.frame.
data2	an object of class matrix or data.frame.
time	a vector of class numeric presenting the measured time points.
feature1	the reference feature to visualise, either the index or the name.
feature2	the associated feature to visualise, either the index or the name.
cutoff	for the associated feature. If fdr=TRUE the false discovery rate (fdr) corrected p-value (default cutoff=0.05). If fdr=FALSE the absolute Pearson Correlation cutoff (default cutoff=0.9).
fdr	(default TRUE) indicating if the false discovery rate of the corrected p-values from the associations object should be used as cutoff to visualize associated profiles. If FALSE the absolute Peason correlation is used as cutoff.
absCor	(default FALSE) if fdr=FALSE you can choose to visualise associations invariant for positive or negative correlation.
withShift	(default FALSE) indicating if the associated feature should be plotted with the time shift.
...	ignored

Details

The function allows to visualise features with and without realignement (or shift) of the time profiles according to the estimated delays using associateData() function from the dynOmics package. Features to be visualised can be filtered either using FDR corrected p-values or a correlation threshold.

Value

plot showing the associated data as calculated by associateData()

See Also

associateData, summary.associations

Examples

```
## Not run:
data(Metabolites)
data(Transcripts)
associations <- associateData(Metabolites[,1:2],Transcripts[,c(1:100)])
#if you only define feature1 or feature2 if will plot all associations
plot(associations,Metabolites,Transcripts,feature1=1,withShift = TRUE)
#if you define feature1 and feature2 it will only plot these two profiles
plot(associations,Metabolites,Transcripts,feature1="Metabolite 1",feature2="Transcript 2")

## End(Not run)
```

summary.associations *Summary of a* associations *Object*

Description

Summarises the associations object returned by the associateData method.

Usage

```
## S3 method for class 'associations'
summary(object, ...)
```

Arguments

object	An object of class associations .
...	Additional arguments which are passed to summary.

Value

summary of the associations object.

Examples

```
## Not run:
data(Metabolites)
data(Transcripts)
associations <- associateData(Metabolites[,1],Transcripts[,c(1:50)])
summary(associations)

## End(Not run)
```

Transcripts *Transcript Simulation Data*

Description

Simulated data were received from Redestig et al., 2011. Metabolite and transcript levels were obtained using an impulse model (Chechik and Koller, 2009). Functions were used to model five different metabolite patterns and for each metabolite 50 associated transcript levels. Time lags were introduced in the range from -2 to 2 with the probability 0.1, 0.2, 0.4, 0.2, 0.1. Simulated profiles have seven time points and normal distributed noise was introduced with mean zero and standard deviation 0.1.

Usage

```
data(Transcripts)
```

Format

This data set contains the simulated expression 250 transcripts for 7 time points.

Details

- Transcripts. data matrix with 7 rows and 250 columns. Each row represents an experimental time sample, and each column a single transcript.

Source

The Transcript Simulation Data is based on the the paper of Redestig *et al.* (2011).

References

Redestig,H. and Costa,I.G. Detection and interpretation of metabolite-transcript coresponses using combined profiling data. *Bioinformatics* **27**(13) (2011), pp. i357 65.

Index