Package 'MoPS'

October 16, 2019

Type Package

Title MoPS - Model-based Periodicity Screening

Version 1.18.0

Date 2014-05-16

Author Philipp Eser, Achim Tresch

Maintainer Philipp Eser <eser@genzentrum.lmu.de>

Description Identification and characterization of periodic fluctuations in time-series data.

Imports Biobase

License GPL-3

biocViews GeneRegulation,Classification,TimeCourse,Regression

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

git_branch RELEASE_3_9

git_last_commit d716467

git_last_commit_date 2019-05-02

Date/Publication 2019-10-15

R topics documented:

MoPS-package *MoPS - Model-based Periodicity Screening*

Description

The MoPS package implements the model-based periodicity screen as used in Eser et al.(Mol Syst Biol, 2014) for the detection and characterization of periodic genes.

Details

Package:	MoPS
Type:	Package
Version:	0.99.0
Date:	2014-04-15
License:	GPL-3

MoPS provides methods for screening numerical time series data for periodicity. See the package vignette for a detailed description of the methods and recommended workflows.

Author(s)

Philipp Eser, Achim Tresch

Maintainer: Philipp Eser <eser@genzentrum.lmu.de>

References

Periodic mRNA synthesis and degradation co-operate during cell cycle gene expression (Eser et al. Mol Sys Biol, 2014)

basic *Dataset containing 10 periodic time series*

Description

Example dataset as used in the MoPS vignette. Data matrix with 10 periodic noisy sine wave time courses (rows) with 41 time points (columns).

Usage

```
data(basic)
```

Format

The format is: num [1:10, 1:41] -0.688 -1.237 0.361 1.004 0.775 ...

ccycle *Dataset containing 500 time series gene expression measurements.*

Description

Example dataset as used in the MoPS Vignette Case Study. Data matrix with 500 time series of gene expression measurements of 41 consecutive measurements seperated by 5 minutes (columns). It is a subset of the dataset published in Eser et al. (Mol Sys Biol, 2014). ArrayExpress accession: E-MTAB-1908.

Usage

```
data(ccycle)
```

Format

The format is: num [1:500, 1:41] - attr(*, "dimnames")=List of 2 ..$: chr [1:500] "YKR077W" "YJL218W" "YGR009C" "YIL104C" ... ..$: chr [1:41] "5" "10" "15" "20" ...

Details

unique gene identifiers are given as row names.

Source

Periodic mRNA synthesis and degradation co-operate during cell cycle gene expression (Eser et al. Mol Sys Biol, 2014)

fit.periodic *Fitting of periodic curves to time series data*

Description

Function that fits periodic curves to each time course in a numeric matrix or ExpressionSet object. It determines the best fitting time courses from an exhaustive set of periodic and non-periodic test functions. These are created either automatically or with a user-specified parameter set (see Arguments). Fitting is done with standard linear regression.

In addition to the best fitting periodic curve, the best-fitting linear time course is estimated. A periodicity score is derived based on the godness-of-fit ratio between periodic and non-periodic fits.

The function returns a list containing the best fitting parameters for each time series. The function result.as.dataframe() converts the result into a data.frame.

Usage

```
fit.periodic(mat,timepoints=NULL,phi=NULL,lambda=NULL,sigma=NULL,psi=NULL,weights=NULL)
```

Arguments

mat	a numeric matrix containing individual measurements (rows) across a time series (columns). mat can also supplied as an ExpressionSet object.
timepoints	optional numeric vector corresponding to measurement timepoints. If NULL, timepoints are initialized as 1:ncol(mat).
phi	optional numeric vector specifying all possible phases of periodic test functions (phase = time where periodic curve is maximal).
lambda	optional numeric vector specifying all possible period lengths of periodic test functions.
sigma	optional numeric vector specifying the magnitude of dampening of the signal along the time course.
psi	optional positive integer defining the level of flexibility to shape the test functions. Recommended values for psi are 3 or 4. psi > 4 results in a tremendous increase in runtime.
weights	optional numeric matrix of weights to be used in the fitting. If non-NULL, weighted least squares is used, otherwise ordinary least squares is used.

Details

The input data needs to be a numeric matrix containing in each row a time series of measurements.

The function can take an optional numeric matrix of weights as input that is used in the fitting process. This matrix needs to have the same dimensions as the input data matrix. If weights are supplied, weighted least squares is used otherwise ordinary least squares is used. This option is useful if the size of the measurement error is not constant for all measurements.

Note that this function uses all possible parameter combinations to create periodic test functions. This can be very time consuming if the user chooses wide parameter ranges as input. If possible, the user should specify meaningful ranges with a moderate spacing between values (see also the MoPS vignette).

Value

fit.periodic() returns a list object containing information about the fitting results for each input time series and the parameter ranges used in the screening.

The first slot of the result object contains the following values for each time series:

$ID	unique id
$is.wPeriodic	TRUE if $minLossPeriodic < $minLossNonPeriodic
$minLossPeriodic	loss of best periodic fit
$minLossNonPeriodic	loss of best non-periodic fit
$phi	phase
$psi	variable sampling points of best fitting psi transformation
$lambda	period length
$sigma	signal attenuation along the time series
$a.coef	coefficient a from linear model (amplitude)
$b.coef	coefficient b from linear model (mean)

The remaining slots contain the following values:

`$time`	measurement time points
`$cols.mat`	number of columns of the input data matrix
`$phi`	all screened phi values
`$lambda`	all screened lambda values
`$sigma`	all screened sigma values

For convenient sorting or filtering, this list can be converted to a data.frame with the function result.as.dataframe().

Author(s)

Philipp Eser, Achim Tresch

References

Chambers, J. M. (1992) Linear models. Chapter 4 of Statistical Models in S eds J. M. Chambers and T. J. Hastie, Wadsworth & Brooks/Cole.

Examples

```
x = seq(0,40,by=1) # time points

## create 10 periodic time series with added noise
mat.p = matrix(rep(x,10),nrow=10,ncol=length(x),byrow=TRUE)
y = -seq(1:10)
mat.p = apply(mat.p,2,function(x){
y = sin(pi*(x/41*6)+y)+rnorm(length(x),sd=1)
})

## add 10 non-periodic noisy time series
mat.nonP = matrix(rep(x,10),nrow=10,ncol=length(x),byrow=TRUE)
mat.nonP = apply(mat.nonP,2,function(x){
y = rnorm(length(x),sd=1)
})

mat = rbind(mat.p,mat.nonP)

res = fit.periodic(mat,phi=seq(0,20,1),lambda=seq(1,20,1))
time.courses = predictTimecourses(res)

plot(mat[1,],type="l",main="",xlab="",ylab="")
points(time.courses[1,],type="l",col="limegreen",lwd=2)
```

predictTimecourses *Prediction of periodic time courses.*

Description

Function that predicts periodic time courses using parameters identified by fit.periodic().

Usage

```
predictTimecourses(res.fits)
```

Arguments

res.fits	List object returned by fit.periodic().

Details

This function takes as input the result list from MoPS function fit.periodic() and creates a list of best fitting time courses. The input list also contains information about the screening parameters, which is used in the generation of predicted time courses.

Value

a numeric matrix containing the predicted values. The number of rows equals the number of rows of the original data matrix, the number of columns equals the number of screened phases.

Author(s)

Philipp Eser, Achim Tresch

Examples

```
x = seq(0,40,by=1) # time points

## create 10 periodic time series with added noise
mat.p = matrix(rep(x,10),nrow=10,ncol=length(x),byrow=TRUE)
y = -seq(1:10)
mat.p = apply(mat.p,2,function(x){
y = sin(pi*(x/41*6)+y)+rnorm(length(x),sd=1)
})

## add 10 non-periodic noisy time series
mat.nonP = matrix(rep(x,10),nrow=10,ncol=length(x),byrow=TRUE)
mat.nonP = apply(mat.nonP,2,function(x){
y = rnorm(length(x),sd=1)
})

mat = rbind(mat.p,mat.nonP)

res = fit.periodic(mat,phi=seq(0,20,1),lambda=seq(1,20,1))
time.courses = predictTimecourses(res)

plot(mat[1,],type="l",main="",xlab="",ylab="")
```

```
points(time.courses[1,],type="l",col="limegreen",lwd=2)
```

`result.as.dataframe` *Parameters of fitted periodic time courses.*

Description

Converts the result list derived by MoPS function fit.periodic() to a data.frame.

Usage

```
result.as.dataframe(result.list)
```

Arguments

`result.list`	List of best fitting parameters returned by fit.periodic().

Details

This function takes as input the result list from MoPS function fit.periodic() and extracts the time course specific optimal parameters.

Value

data.frame containing the best fitting periodic parameters for each time series (rows):

columns: ID : unique identifier score : log-likelihood for periodic behaviour phi : phase lambda : period length sigma : attenuation of the signal along the complete time series mean : mean amplitude : amplitude

Author(s)

Philipp Eser, Achim Tresch

Examples

```
y = 2*sin(seq(0,6*pi,length.out=50)+rnorm(50))
res = fit.periodic(y)
result.as.dataframe(res)
```

Index