Package ‘earlywarnings’

October 13, 2022

Type Package

Title Early Warning Signals for Critical Transitions in Time Series

Version 1.1.29

Date 2022-10-10

Description The Early-Warning-Signals Toolbox provides methods for estimating statistical changes in time series that can be used for identifying nearby critical transitions.

Depends R (>= 3.0.2), ggplot2, moments, tgp, tseries

Imports fields, nortest, graphics, grDevices, quadprog, Kendall, KernSmooth, methods, lmtest, som, spam, stats, knitr

VignetteBuilder knitr

LazyLoad yes

Encoding UTF-8

URL http://www.early-warning-signals.org, http://www.vasilisdakos.info

License BSD_2_clause + file LICENSE

RoxygenNote 7.2.0

NeedsCompilation no

Author Vasilis Dakos [aut, cre], Leo Lahti [aut] (<https://orcid.org/0000-0001-5537-637X>)

Maintainer Vasilis Dakos <vasilis.dakos@gmail.com>

Repository CRAN

Date/Publication 2022-10-10 18:50:15 UTC

R topics documented:

`bdstest_ews` *BDS test Early Warning Signals*

Description

`bdstest_ews` is used to estimate the BDS statistic to detect nonlinearity in the residuals of a timeseries after first-difference detrending, fitting an ARMA(p,q) model, and fitting a GARCH(0,1) model. The function is making use of `bds.test` from the tseries package.

Usage

```
bdstest_ews(
  timeseries,
  ARMAoptim = TRUE,
  ARMAorder = c(1, 0),
  GARCHorder = c(0, 1),
  embdim = 3,
  epsilon = c(0.5, 0.75, 1),
  boots = 1000,
  logtransform = FALSE,
  interpolate = FALSE
)
```

Arguments

`timeseries` a numeric vector of the observed univariate timeseries values or a numeric matrix where the first column represents the time index and the second the observed timeseries values. Use vectors/matrices with headings.

`ARMAoptim` is the order of the `ARMA(p,q)` model to be fitted on the original timeseries. If TRUE the best ARMA model based on AIC is applied. If FALSE the `ARMAorder` is used.

`ARMAorder` is the order of the `AR(p)` and `MA(q)` process to be fitted on the original timeseries. Default is `p=1 q=0`.

`GARCHorder`	fits a GARCH model on the original timeseries where `GARCHorder[1]` is the GARCH part and `GARCHorder[2]` is the ARCH part.
`embdim`	is the embedding dimension (2, 3,... `embdim`) up to which the BDS test will be estimated (must be numeric). Default value is 3.
`epsilon`	is a numeric vector that is used to scale the standard deviation of the timeseries. The BDS test is computed for each element of epsilon. Default is 0.5, 0.75 and 1.
`boots`	is the number of bootstraps performed to estimate significance p values for the BDS test. Default is 1000.
`logtransform`	logical. If TRUE data are logtransformed prior to analysis as log(X+1). Default is FALSE.
`interpolate`	logical. If TRUE linear interpolation is applied to produce a timeseries of equal length as the original. Default is FALSE (assumes there are no gaps in the timeseries).

Details

The function requires the installation of packages tseries and quadprog that are not available under Linux and need to be manually installed under Windows.

Value

`bdstest_ews` returns output on the R console that summarizes the BDS test statistic for all embedding dimensions and `epsilon` values used, and for first-differenced data, ARMA(p.q) residuals, and GARCH(0,1) residuals). Also the significance p values are returned estimated both by comparing to a standard normal distribution and by bootstrapping.

In addition, `bdstest_ews` returns a plot with the original timeseries, the residuals after first-differencing, and fitting the ARMA(p,q) and GARCH(0,1) models. Also the autocorrelation `acf` and partial autocorrelation `pacf` functions are estimated serving as guides for the choice of lags of the linear models fitted to the data.

Author(s)

S. R. Carpenter, modified by V. Dakos

References

J. B. Cromwell, W. C. Labys and M. Terraza (1994): Univariate Tests for Time Series Models, Sage, Thousand Oaks, CA, pages 32-36.

Dakos, V., et al (2012).'Methods for Detecting Early Warnings of Critical Transitions in Time Series Illustrated Using Simulated Ecological Data.' *PLoS ONE* 7(7): e41010. doi:10.1371/journal.pone.0041010

See Also

`generic_ews`; `ddjnonparam_ews`; `bdstest_ews`; `sensitivity_ews`; `surrogates_ews`; `ch_ews`; `movpotential_ews`; `livpotential_ews`;

Examples

```
data(foldbif)
bdstest_ews(foldbif, ARMAoptim=FALSE, ARMAorder=c(1,0),
         embdim=3, epsilon=0.5, boots=200,
      logtransform=FALSE, interpolate=FALSE)
```

---

ch_ews — *Conditional Heteroskedasticity*

Description

`ch_ews` is used to estimate changes in conditional heteroskedasticity within rolling windows along a timeseries

Usage

```
ch_ews(
  timeseries,
  winsize = 10,
  alpha = 0.1,
  optim = TRUE,
  lags = 4,
  logtransform = FALSE,
  interpolate = FALSE
)
```

Arguments

`timeseries`	a numeric vector of the observed timeseries values or a numeric matrix where the first column represents the time index and the second the observed timeseries values. Use vectors/matrices with headings.
`winsize`	is length of the rolling window expressed as percentage of the timeseries length (must be numeric between 0 and 100). Default is 10%.
`alpha`	is the significance threshold (must be numeric). Default is 0.1.
`optim`	logical. If TRUE an autoregressive model is fit to the data within the rolling window using AIC optimization. Otherwise an autoregressive model of specific order `lags` is selected.
`lags`	is a parameter that determines the specific order of an autoregressive model to fit the data. Default is 4.
`logtransform`	logical. If TRUE data are logtransformed prior to analysis as log(X+1). Default is FALSE.
`interpolate`	logical. If TRUE linear interpolation is applied to produce a timeseries of equal length as the original. Default is FALSE (assumes there are no gaps in the timeseries).

Value

ch_ews returns a matrix that contains: time the time index. r.squared the R2 values of the regressed residuals. critical.value the chi-square critical value based on the desired alpha level for 1 degree of freedom divided by the number of residuals used in the regression. test.result logical. It indicates whether conditional heteroskedasticity was significant. ar.fit.order the order of the specified autoregressive model- only informative if optim FALSE was selected.

In addition, ch_ews plots the original timeseries and the R2 where the level of significance is also indicated.

Author(s)

T. Cline, modified by V. Dakos

References

Seekell, D. A., et al (2011). 'Conditional heteroscedasticity as a leading indicator of ecological regime shifts.' *American Naturalist* 178(4): 442-451

Dakos, V., et al (2012).'Methods for Detecting Early Warnings of Critical Transitions in Time Series Illustrated Using Simulated Ecological Data.' *PLoS ONE* 7(7): e41010. doi:10.1371/journal.pone.0041010

See Also

generic_ews; ddjnonparam_ews; bdstest_ews; sensitivity_ews; surrogates_ews; ch_ews; movpotential_ews; livpotential_ews

Examples

```
data(foldbif)
out=ch_ews(foldbif, winsize=50, alpha=0.05, optim=TRUE, lags)
```

---

circulation *circulation data set*

Description

circulation data set

Format

TBA

Source

TBA

References

See citation('earlywarnings')

Examples

```
#
```

---

ddjnonparam_ews *Drift Diffusion Jump Nonparametrics Early Warning Signals*

Description

ddjnonparam_ews is used to compute nonparametrically conditional variance, drift, diffusion and jump intensity in a timeseries and it also interpolates to obtain the evolution of the nonparametric statistics in time.

Usage

```
ddjnonparam_ews(
  timeseries,
  bandwidth = 0.6,
  na = 500,
  logtransform = TRUE,
  interpolate = FALSE
)
```

Arguments

timeseries	a numeric vector of the observed univariate timeseries values or a numeric matrix where the first column represents the time index and the second the observed timeseries values. Use vectors/matrices with headings.
bandwidth	is the bandwidht of the kernel regressor (must be numeric). Default is 0.6.
na	is the number of points for computing the kernel (must be numeric). Default is 500.
logtransform	logical. If TRUE data are logtransformed prior to analysis as log(X+1). Default is FALSE.
interpolate	logical. If TRUE linear interpolation is applied to produce a timeseries of equal length as the original. Default is FALSE (assumes there are no gaps in the timeseries).

Details

The approach is based on estimating terms of a drift-diffusion-jump model as a surrogate for the unknown true data generating process: $dx = f(x, \theta)dt + g(x, \theta)dW + dJ$. Here x is the state variable, f() and g() are nonlinear functions, dW is a Wiener process and dJ is a jump process. Jumps are large, one-step, positive or negative shocks that are uncorrelated in time. In addition, ddjnonparam_ews returns a first plot with the original timeseries and the residuals after first-differencing. A second plot shows the nonparametric conditional variance, total variance, diffusion and jump intensity over the data, and a third plot the same nonparametric statistics over time.

Value

ddjnonparam_ews returns an object with elements: avec is the mesh for which values of the non-parametric statistics are estimated. S2.vec is the conditional variance of the timeseries x over avec. TotVar.dx.vec is the total variance of dx over avec. Diff2.vec is the diffusion estimated as total variance - jumping intensity vs avec. LamdaZ.vec is the jump intensity over avec. Tvec1 is the timeindex. S2.t is the conditional variance of the timeseries x data over Tvec1. TotVar.t is the total variance of dx over Tvec1. Diff2.t is the diffusion over Tvec1. Lamda.t is the jump intensity over Tvec1.

Author(s)

S. R. Carpenter, modified by V. Dakos and L. Lahti

References

Carpenter, S. R. and W. A. Brock (2011). 'Early warnings of unknown nonlinear shifts: a nonparametric approach.' *Ecology* 92(12): 2196-2201

Dakos, V., et al (2012).'Methods for Detecting Early Warnings of Critical Transitions in Time Series Illustrated Using Simulated Ecological Data.' *PLoS ONE* 7(7): e41010. doi:10.1371/journal.pone.0041010

See Also

generic_ews; ddjnonparam_ews; bdstest_ews; sensitivity_ews;surrogates_ews; ch_ews; movpotential_ews; livpotential_ews

Examples

```
data(foldbif)
output<-ddjnonparam_ews(foldbif,bandwidth=0.6,na=500,
logtransform=TRUE,interpolate=FALSE)
```

---

find.optima *find.optima*

Description

Detect optima, excluding very local optima below detection.threshold.

Usage

```
find.optima(f, detection.threshold = 0, bw, detection.limit = 1)
```

Arguments

`f` density

`detection.threshold` detection threshold for peaks

`bw` bandwidth

`detection.limit` Minimun accepted density for a maximum; as a multiple of kernel height

Value

A list with the following elements: min minima max maxima detection.density Minimum detection density

Author(s)

Leo Lahti <leo.lahti@iki.fi>

---

foldbif *foldbif data set*

Description

foldbif data set

Format

TBA

Source

TBA

References

See citation('earlywarnings')

Examples

```
#
```

generic_ews *Generic Early Warning Signals*

Description

generic_ews is used to estimate statistical moments within rolling windows along a timeseries.

Usage

```
generic_ews(
  timeseries,
  winsize = 50,
  detrending = c("no", "gaussian", "loess", "linear", "first-diff"),
  bandwidth = NULL,
  span = NULL,
  degree = NULL,
  logtransform = FALSE,
  interpolate = FALSE,
  AR_n = FALSE,
  powerspectrum = FALSE
)
```

Arguments

`timeseries`	a numeric vector of the observed univariate timeseries values or a numeric matrix where the first column represents the time index and the second the observed timeseries values. Use vectors/matrices with headings. If the powerspectrum is to be plotted as well, the timeseries lenght should be even number.
`winsize`	is the size of the rolling window expressed as percentage of the timeseries length (must be numeric between 0 and 100). Default is 50%.
`detrending`	the timeseries can be detrended/filtered prior to analysis. There are four options: gaussian filtering, loess fitting, linear detrending and first-differencing. Default is no detrending.
`bandwidth`	for the Gaussian kernel when gaussian filtering is applied. It is expressed as percentage of the timeseries length (must be numeric between 0 and 100). Alternatively it can be given by the bandwidth selector bw.nrd0 (Default).
`span`	parameter that controls the degree of smoothing (numeric between 0 and 100, Default 25).
`degree`	the degree of polynomial to be used for when loess fitting is applied, normally 1 or 2 (Default).
`logtransform`	logical. If TRUE data are logtransformed prior to analysis as log(X+1). Default is FALSE.
`interpolate`	logical. If TRUE linear interpolation is applied to produce a timeseries of equal length as the original. Default is FALSE (assumes there are no gaps in the timeseries).

AR_n	logical. If TRUE the best fitted AR(n) model is fitted to the data. Default is FALSE.
powerspectrum	logical. If TRUE the power spectrum within each rolling window is plotted. Default is FALSE.

Details

In addition, generic_ews returns three plots. The first plot contains the original data, the detrending/filtering applied and the residuals (if selected), and all the moment statistics. For each statistic trends are estimated by the nonparametric Kendall tau correlation. The second plot, if asked, quantifies resilience indicators fitting AR(n) selected by the Akaike Information Criterion. The third plot, if asked, is the power spectrum estimated by spec.ar for all frequencies within each rolling window.

Value

generic_ews returns a matrix that contains: tim the time index. ar1 the autoregressive coefficient ar(1) of a first order AR model fitted on the data within the rolling window. sd the standard deviation of the data estimated within each rolling window. sk the skewness of the data estimated within each rolling window. kurt the kurtosis of the data estimated within each rolling window. cv the coefficient of variation of the data estimated within each rolling window. returnrate the return rate of the data estimated as 1-ar(1) cofficient within each rolling window. densratio the density ratio of the power spectrum of the data estimated as the ratio of low frequencies over high frequencies within each rolling window; acf1 the autocorrelation at first lag of the data estimated within each rolling window.

Author(s)

Vasilis Dakos <vasilis.dakos@gmail.com>

References

Ives, A. R. (1995). 'Measuring resilience in stochastic systems.' *Ecological Monographs* 65: 217-233

Dakos, V., et al (2008). 'Slowing down as an early warning signal for abrupt climate change.' *Proceedings of the National Academy of Sciences* 105(38): 14308-14312

Dakos, V., et al (2012).'Methods for Detecting Early Warnings of Critical Transitions in Time Series Illustrated Using Simulated Ecological Data.' *PLoS ONE* 7(7): e41010. doi:10.1371/journal.pone.0041010

Examples

```
data(foldbif)
out=generic_ews(foldbif,winsize=50,detrending='gaussian',
bandwidth=5,logtransform=FALSE,interpolate=FALSE)
```

livpotential_ews *Potential Analysis for univariate data*

Description

livpotential_ews performs one-dimensional potential estimation derived from a uni-variate time-series.

Usage

```
livpotential_ews(
  x,
  std = 1,
  bw = "nrd",
  weights = c(),
  grid.size = NULL,
  detection.threshold = 1,
  bw.adjust = 1,
  density.smoothing = 0,
  detection.limit = 1
)
```

Arguments

x	Univariate data (vector) for which the potentials shall be estimated
std	Standard deviation of the noise (defaults to 1; this will set scaled potentials)
bw	kernel bandwidth estimation method
weights	optional weights in ksdensity (used by movpotentials).
grid.size	Grid size for potential estimation.
detection.threshold	maximum detection threshold as fraction of density kernel height dnorm(0, sd = bandwidth)/N
bw.adjust	The real bandwidth will be bw.adjust*bw; defaults to 1
density.smoothing	Add a small constant density across the whole observation range to regularize density estimation (and to avoid zero probabilities within the observation range). This parameter adds uniform density across the observation range, scaled by density.smoothing.
detection.limit	minimum accepted density for a maximum; as a multiple of kernel height return `livpotential` returns a list with the following elements: xi the grid of points on which the potential is estimated pot The estimated potential: -log(f)*std^2/2, where f is the density. density Density estimate corresponding to the potential. min.inds indices of the grid points at which the density has minimum values; (-potentials; neglecting local optima) max.inds indices the grid

points at which the density has maximum values; (-potentials; neglecting local optima) bw bandwidth of kernel used min.points grid point values at which the density has minimum values; (-potentials; neglecting local optima) max.points grid point values at which the density has maximum values; (-potentials; neglecting local optima)

Author(s)

Based on Matlab code from Egbert van Nes modified by Leo Lahti. Implemented in early warnings package by V. Dakos.

References

Livina, VN, F Kwasniok, and TM Lenton, 2010. Potential analysis reveals changing number of climate states during the last 60 kyr . *Climate of the Past*, 6, 77-82.

Dakos, V., et al (2012).'Methods for Detecting Early Warnings of Critical Transitions in Time Series Illustrated Using Simulated Ecological Data.' *PLoS ONE* 7(7): e41010. doi:10.1371/journal.pone.0041010

Examples

```
data(foldbif)
res <- livpotential_ews(foldbif[,1])
```

`movpotential_ews` *Moving Average Potential*

Description

This function reconstructs a potential derived from data along a gradient of a given parameter.

Usage

```
movpotential_ews(
  X,
  param = NULL,
  bw = "nrd",
  bw.adjust = 1,
  detection.threshold = 0.1,
  std = 1,
  grid.size = 50,
  plot.cutoff = 0.5,
  plot.contours = TRUE,
  binwidth = 0.2,
  bins = NULL
)
```

Arguments

`X`	a vector of the X observations of the state variable of interest
`param`	parameter values corresponding to the observations in X
`bw`	Bandwidth for smoothing kernels. Automatically determined by default.
`bw.adjust`	Bandwidth adjustment constant
`detection.threshold`	Threshold for local optima to be discarded.
`std`	Standard deviation.
`grid.size`	number of evaluation points; number of steps between min and max potential; also used as kernel window size
`plot.cutoff`	cuttoff for potential minima and maxima in visualization
`plot.contours`	Plot contours on the landscape visualization
`binwidth`	binwidth for contour plot
`bins`	bins for contour plot. Overrides binwidth if given

Value

A list with the following elements: pars values of the covariate parameter as matrix; xis values of the x as matrix; pots smoothed potentials; mins minima in the densities (-potentials; neglecting local optima); maxs maxima in densities (-potentials; neglecting local optima); plot an object that displays the potential estimated in 2D

Author(s)

L. Lahti, E. van Nes, V. Dakos.

References

Hirota, M., Holmgren, M., van Nes, E.H. & Scheffer, M. (2011). Global resilience of tropical forest and savanna to critical transitions. *Science*, 334, 232-235.

Examples

```
X <- c(rnorm(1000, mean = 0), rnorm(1000, mean = -2), rnorm(1000, mean = 2));
param <- seq(0,5,length=3000);
res <- movpotential_ews(X, param)
```

`PlotPotential` *Plot Potential*

Description

Visualization of the potential function from the movpotential function.

Usage

```
PlotPotential(
  res,
  title = "",
  xlab.text,
  ylab.text,
  cutoff = 0.5,
  plot.contours = TRUE,
  binwidth = 0.2,
  bins = NULL
)
```

Arguments

`res`	output from movpotential function
`title`	title text
`xlab.text`	xlab text
`ylab.text`	ylab text
`cutoff`	parameter determining the upper limit of potential for visualizations
`plot.contours`	Plot contour lines.
`binwidth`	binwidth for contour plot
`bins`	bins for contour plot. Overrides binwidth if given

Value

ggplot2 potential plot

Author(s)

Leo Lahti `<leo.lahti@iki.fi>`

References

Dakos, V., et al (2012).'Methods for Detecting Early Warnings of Critical Transitions in Time Series Illustrated Using Simulated Ecological Data.' *PLoS ONE* 7(7): e41010. doi:10.1371/journal.pone.0041010

Examples

```
X = c(rnorm(1000, mean = 0), rnorm(1000, mean = -2),
              rnorm(1000, mean = 2))
     param = seq(0,5,length=3000);
     res <- movpotential_ews(X, param);
     PlotPotential(res$res, title = '',
                    xlab.text = '', ylab.text = '',
   cutoff = 0.5,
   plot.contours = TRUE, binwidth = 0.2)
```

---

qda_ews *Quick Detection Analysis for Generic Early Warning Signals*

Description

Estimate autocorrelation, variance within rolling windows along a timeseries, test the significance of their trends, and reconstruct the potential landscape of the timeseries.

Usage

```
qda_ews(
  timeseries,
  param = NULL,
  winsize = 50,
  detrending = c("no", "gaussian", "linear", "first-diff"),
  bandwidth = NULL,
  boots = 100,
  s_level = 0.05,
  cutoff = 0.05,
  detection.threshold = 0.002,
  grid.size = 50,
  logtransform = FALSE,
  interpolate = FALSE
)
```

Arguments

`timeseries`	a numeric vector of the observed univariate timeseries values or a numeric matrix where the first column represents the time index and the second the observed timeseries values. Use vectors/matrices with headings.
`param`	values corresponding to observations in timeseries
`winsize`	is the size of the rolling window expressed as percentage of the timeseries length (must be numeric between 0 and 100). Default is 50%.
`detrending`	the timeseries can be detrended/filtered prior to analysis. There are four options: `gaussian` filtering, `linear` detrending and `first-differencing`. Default is no detrending.

bandwidth	is the bandwidth used for the Gaussian kernel when gaussian filtering is applied. It is expressed as percentage of the timeseries length (must be numeric between 0 and 100). Alternatively it can be given by the bandwidth selector bw.nrd0 (Default).
boots	the number of surrogate data to generate from fitting an ARMA(p,q) model. Default is 100.
s_level	significance level. Default is 0.05.
cutoff	the cutoff value to visualize the potential landscape
detection.threshold	detection threshold for potential minima
grid.size	grid size (for potential analysis)
logtransform	logical. If TRUE data are logtransformed prior to analysis as log(X+1). Default is FALSE.
interpolate	logical. If TRUE linear interpolation is applied to produce a timeseries of equal length as the original. Default is FALSE (assumes there are no gaps in the timeseries).

Value

qda_ews produces three plots. The first plot contains the original data, the detrending/filtering applied and the residuals (if selected), autocorrelation and variance. For each statistic trends are estimated by the nonparametric Kendall tau correlation. The second plot, returns a histogram of the distributions of the Kendall trend statistic for autocorrelation and variance estimated on the surrogated data. Vertical lines represent the level of significance, whereas the black dots the actual trend found in the time series. The third plot is the reconstructed potential landscape in 2D. In addition, the function returns a list containing the output from the respective functions generic_RShiny (indicators); surrogates_RShiny (trends); movpotential_ews (potential analysis)

Author(s)

Vasilis Dakos, Leo Lahti, March 1, 2013 <vasilis.dakos@gmail.com>

References

Dakos, V., et al (2012).'Methods for Detecting Early Warnings of Critical Transitions in Time Series Illustrated Using Simulated Ecological Data.' *PLoS ONE* 7(7): e41010. doi:10.1371/journal.pone.0041010

See Also

generic_ews; ddjnonparam_ews; bdstest_ews; sensitivity_ews; surrogates_ews; ch_ews; movpotential_ews; livpotential_ews;

Examples

```
data(foldbif)
out <- qda_ews(foldbif, param = NULL, winsize = 50,
                 detrending='gaussian', bandwidth=NULL,
  boots = 10, s_level = 0.05, cutoff=0.05,
```

```
detection.threshold = 0.002, grid.size = 50,
logtransform=FALSE, interpolate=FALSE)
```

---

sensitivity_ews *Sensitivity Early Warning Signals*

Description

sensitivity_ews is used to estimate trends in statistical moments for different sizes of rolling windows along a timeseries and the trends are estimated by the nonparametric Kendall tau correlation coefficient.

Usage

```
sensitivity_ews(
  timeseries,
  indicator = c("ar1", "sd", "acf1", "sk", "kurt", "cv", "returnrate", "densratio"),
  winsizerange = c(25, 75),
  incrwinsize = 25,
  detrending = c("no", "gaussian", "loess", "linear", "first-diff"),
  bandwidthrange = c(5, 100),
  spanrange = c(5, 100),
  degree = NULL,
  incrbandwidth = 20,
  incrspanrange = 10,
  logtransform = FALSE,
  interpolate = FALSE
)
```

Arguments

timeseries	a numeric vector of the observed univariate timeseries values or a numeric matrix where the first column represents the time index and the second the observed timeseries values. Use vectors/matrices with headings.
indicator	is the statistic (leading indicator) selected for which the sensitivity analysis is perfomed. Currently, the indicators supported are: ar1 autoregressive coefficient of a first order AR model, sd, standard deviation, acf1 autocorrelation at first lag, sk skewness, kurt kurtosis, cv coeffcient of variation, returnrate, and densratio density ratio of the power spectrum at low frequencies over high frequencies.
winsizerange	is the range of the rolling window sizes expressed as percentage of the timeseries length (must be numeric between 0 and 100). Default is 25% - 75%.
incrwinsize	increments the rolling window size (must be numeric between 0 and 100). Default is 25.
detrending	the timeseries can be detrended/filtered. There are three options: gaussian filtering, loess fitting, linear detrending and first-differencing. Default is no detrending.

`bandwidthrange`	is the range of the bandwidth used for the Gaussian kernel when gaussian filtering is selected. It is expressed as percentage of the timeseries length (must be numeric between 0 and 100). Default is 5% - 100%.
`spanrange`	parameter that controls the degree of smoothing (numeric between 0 and 100). Default is 5% - 100%.
`degree`	the degree of polynomial to be used for when loess fitting is applied, normally 1 or 2 (Default).
`incrbandwidth`	is the size to increment the bandwidth used for the Gaussian kernel when gaussian filtering is applied. It is expressed as percentage of the timeseries length (must be numeric between 0 and 100). Default is 20.
`incrspanrange`	Span range
`logtransform`	logical. If TRUE data are logtransformed prior to analysis as log(X+1). Default is FALSE.
`interpolate`	logical. If TRUE linear interpolation is applied to produce a timeseries of equal length as the original. Default is FALSE (assumes there are no gaps in the timeseries).

Details

In addition, `sensitivity_ews` returns a plot with the Kendall tau estimates and their p-values for the range of rolling window sizes used, together with a histogram of the distributions of the statistic and its significance. When gaussian filtering is chosen, a contour plot is produced for the Kendall tau estimates and their p-values for the range of both rolling window sizes and bandwidth used. A reverse triangle indicates the combination of the two parameters for which the Kendall tau was the highest

Value

`sensitivity_ews` returns a matrix that contains the Kendall tau rank correlation estimates for the rolling window sizes (rows) and bandwidths (columns), if gaussian filtering is selected.

Author(s)

Vasilis Dakos <vasilis.dakos@gmail.com>

References

Dakos, V., et al (2008). 'Slowing down as an early warning signal for abrupt climate change.' *Proceedings of the National Academy of Sciences* 105(38): 14308-14312

Dakos, V., et al (2012).'Methods for Detecting Early Warnings of Critical Transitions in Time Series Illustrated Using Simulated Ecological Data.' *PLoS ONE* 7(7): e41010. doi:10.1371/journal.pone.0041010

Examples

```
data(foldbif)
output=sensitivity_ews(foldbif,indicator='sd',detrending='gaussian',
incrwinsize=25,incrbandwidth=20)
```

surrogates_ews *Surrogates Early Warning Signals*

Description

surrogates_ews is used to estimate distributions of trends in statistical moments from different surrogate timeseries generated after fitting an ARMA(p,q) model on the data. The trends are estimated by the nonparametric Kendall tau correlation coefficient and can be compared to the trends estimated in the original timeseries to produce probabilities of false positives.

Usage

```
surrogates_ews(
  timeseries,
 indicator = c("ar1", "sd", "acf1", "sk", "kurt", "cv", "returnrate", "densratio"),
  winsize = 50,
  detrending = c("no", "gaussian", "loess", "linear", "first-diff"),
  bandwidth = NULL,
  span = NULL,
  degree = NULL,
  boots = 100,
  logtransform = FALSE,
  interpolate = FALSE
)
```

Arguments

timeseries	a numeric vector of the observed univariate timeseries values or a numeric matrix where the first column represents the time index and the second the observed timeseries values. Use vectors/matrices with headings.
indicator	is the statistic (leading indicator) selected for which the surrogate timeseries are produced. Currently, the indicators supported are: `ar1` autoregressive coefficient of a first order AR model, `sd` standard deviation, `acf1` autocorrelation at first lag, `sk` skewness, `kurt` kurtosis, `cv` coeffcient of variation, `returnrate`, and `densratio` density ratio of the power spectrum at low frequencies over high frequencies.
winsize	is the size of the rolling window expressed as percentage of the timeseries length (must be numeric between 0 and 100). Default valuise 50%.
detrending	the timeseries can be detrended/filtered prior to analysis. There are three options: `gaussian` filtering, `loess` fitting, `linear` detrending and `first-differencing`. Default is no detrending.
bandwidth	is the bandwidth used for the Gaussian kernel when gaussian filtering is selected. It is expressed as percentage of the timeseries length (must be numeric between 0 and 100). Alternatively it can be given by the bandwidth selector bw.nrd0 (Default).

span	parameter that controls the degree of smoothing (numeric between 0 and 100, Default 25). see more on loessstats
degree	the degree of polynomial to be used for when loess fitting is applied, normally 1 or 2 (Default). see more on loessstats
boots	the number of surrogate data. Default is 100.
logtransform	logical. If TRUE data are logtransformed prior to analysis as log(X+1). Default is FALSE.
interpolate	logical. If TRUE linear interpolation is applied to produce a timeseries of equal length as the original. Default is FALSE (assumes there are no gaps in the timeseries).

Details

In addition, `surrogates_ews` returns a plot with the distribution of the surrogate Kendall tau estimates and the Kendall tau estimate of the original series. Vertical lines indicate the 5% and 95% significance levels.

Value

`surrogates_ews` returns a matrix that contains: Kendall tau estimate original the trends of the original timeseries; Kendall tau p-value original the p-values of the trends of the original timeseries; Kendall tau estimate surrogates the trends of the surrogate timeseries; Kendall tau p-value surrogates the associated p-values of the trends of the surrogate timeseries; significance p the p-value for the original Kendall tau rank correlation estimate compared to the surrogates;

Author(s)

Vasilis Dakos `<vasilis.dakos@gmail.com>`

References

Dakos, V., et al (2008). 'Slowing down as an early warning signal for abrupt climate change.' *Proceedings of the National Academy of Sciences* 105(38): 14308-14312

Dakos, V., et al (2012).'Methods for Detecting Early Warnings of Critical Transitions in Time Series Illustrated Using Simulated Ecological Data.' *PLoS ONE* 7(7): e41010. doi:10.1371/journal.pone.0041010

Examples

```
data(foldbif)
output <- surrogates_ews(foldbif,indicator='sd',winsize=50,detrending='gaussian', bandwidth=10,
            boots=200, logtransform=FALSE,interpolate=FALSE)
```

`UnivariateGrouping` *Get group assigment indices for univariate data points, given cluster break points*

Description

Get group assigment indices for univariate data points, given cluster break points

Usage

```
UnivariateGrouping(x, breakpoints)
```

Arguments

`x`	Univariate data vector
`breakpoints`	Cluster breakpoints

Value

A vector of cluster indices

Author(s)

Leo Lahti <leo.lahti@iki.fi>

`YD2PB_grayscale` *YD2PB_grayscale data set*

Description

YD2PB_grayscale data set

Format

TBA

Source

TBA

References

See citation('earlywarnings')

Examples

```
#
```

Index