Package ‘nlts’

October 12, 2018

Title  Nonlinear Time Series Analysis
Date   2018-10-11
Author Ottar N. Bjornstad [aut, cre]
Maintainer Ottar N. Bjornstad <onb1@psu.edu>
Description R functions for (non)linear time series analysis with an emphasis on nonparametric autoregression and order estimation, and tests for linearity / additivity.
Version 1.0-2
Depends R (>= 2.2.0), locfit (>= 1.5.3), acepack
NeedsCompilation no
URL http://ento.psu.edu/directory/onb1
License GPL-3
RoxygenNote 6.0.1
Repository CRAN
Date/Publication 2018-10-12 20:00:02 UTC

R topics documented:

add.test ................................................. 2
contingency.periodogram ............................... 3
lin.order.cls ........................................ 4
lin.test ............................................... 5
ll.edm .................................................. 6
ll.order .............................................. 7
lpx ...................................................... 9
mkx ..................................................... 10
plodia .................................................. 10
plot.contingency.periodogram ....................... 11
plot.lin.order ....................................... 11
plot.ll.order ....................................... 12
plot.lomb .......................................... 13
plot.ppll ......................................... 13
add.test

Description

add.test is a function to test the permissibility of the additive autoregressive model:

Usage

add.test(x, order, n.cond = FALSE)

Arguments

x  A time series (vector without missing values).
order a scalar representing the order to be considered.
n.cond The number of observation to condition on. The default is order (must be >= order)

Details

\[ N(t) = f1(N(t-1)) + f2(N(t-2)) + \ldots + fd(N(t-d)) + e(t) \]

against the alternative:

\[ N(t) = F(N(t-1), N(t-2), \ldots, N(t-d)) + e(t) \]

This is the Lagrange multiplier test for additivity developed by Chen et al. (1995: test II).

Value

a vector is returned consisting of the asymptotic chi-square value, the associated d.f. and asymptotic
p.val for the test of additivity.
References


Examples

```r
data(plodia)
add.test(sqrt(plodia), order = 3)
```

```r
contingency.periodogram

The contingency periodogram for periodicity in categorical time series

Description

A function to estimate the contingency periodogram to test for periodicity in categorical time series.

Usage

```r
contingency.periodogram(x, maxper = 6, exact = FALSE)
```

Arguments

- `x` A vector representing the categorical time series.
- `maxper` the maximum lag (period) considered.
- `exact` If TRUE the FISHER exact test is calculated

Details

This is the contingency periodogram of Pierre Legedre and Pierre Dutiel to test for periodicity in categorical time series. I have coded the function so as to provide both the Fisher exact test and the asymptotic chi-square test.

Value

An object of class "contingency.periodogram" is returned consisting of a matrix with a row for each period considered. The columns are:

- `exact.p` the Fisher exact test at each lag (if exact=TRUE).
- `chi2` the asymptotic chi-square value.
- `df` the chi-square degrees-of-freedom.
- `asympt.p` the chi-squared asymptotic p-value.
lin.order.cls

The order of a time series using cross-validation of the linear autoregressive model (conditional least-squares).

Description
A function to estimate the order of a time series using cross-validation of the linear autoregressive model. Coefficients are estimated using conditional least-squares. I coded this functions to estimate the order of ecological time series. Bjornstad et al. (1998, 2001)

Usage
lin.order.cls(x, order = 1:5, n.cond = 5, echo = TRUE)

Arguments
- `x` A time series without missing values
- `order` The candidate orders. The default is 1:5
- `n.cond` The number of observation to condition on. The default is 5 (must be \( \geq \max(order) \))
- `echo` if TRUE a counter for the data points and the orders is produced to monitor progress.

Details
The time series is normalized prior to cross-validation.
Note that if the dynamics is highly nonlinear, the nonparametric order-estimators (ll.order) may be more appropriate. (I coded this function to use for comparison with the nonparametric methods, because these also uses (nonlinear) conditional least-squares.)

Value
An object of class "lin.order" is returned consisting of the following components:
- `order` the grid of orders considered.
- `CVd` the cross-validation errors across the grid of orders.
lin.test

References


See Also

ll.order

Examples

data(plodia)
fit <- lin.order.cls(sqrt(plodia), order=1:5)
## Not run: plot(fit)
summary(fit)

lin.test A Tukey one-degree-of-freedom test for linearity in time series.

Description

a function to test the permissibility of the linear autoregressive model:

Usage

lin.test(x, order)

Arguments

x A time series (vector without missing values).
order a scalar representing the order to be considered.

Details

N(t) = a0 + a1N(t-1) + a2N(t-2) + ... + adN(t-d) + e(t )
against the alternative:
Nt = F(N(t-1), N(t-2), ..., N(t-d)) + e(t)
This is the Tukey one-degree-of-freedom test of linearity developed by Tsay (1986). Orders up to 5 is permissible. [although the code is easily extended].
Value
A vector is returned consisting of the asymptotic F-value, the associated numerator and denominator
d.f.’s and asymptotic p.val for the test of linearity

References

Examples
```
data(plodia)
lin.test(sqrt(plodia), order = 3)
```

```
ll. edm Nonlinear forecasting of local polynomial 'empirical dynamic model'.
```

Description
A function to forecast a local polynomial 'empirical dynamic model'.

Usage
```
ll. edm(x, order, bandwidth, len = NA, deg = 2)
```

Arguments
- `x`: A time series without missing values.
- `order`: The order for the nonparametric (local polynomial) autoregression.
- `bandwidth`: The bandwidth for the nonparametric (local polynomial) autoregression.
- `len`: The length of the predicted time-series. If NA the length of the training time series will be used.
- `deg`: The degree of the local polynomial.

Details
The function produces a nonlinear (nonparametric) forecast using the conditional mean method of Fan et al (1996). A Gaussian kernel is used for the local polynomial autoregression.
The bandwidth and order is best estimated with the `ll.order`-function.
Missing values are NOT permitted.
If deg is set to 0, the forecast uses the Nadaraya-Watson (locally constant) estimator of the conditional expectation against lagged-abundances.

Value
A time series with the nonlinear (nonparametric) forecast is returned
References


See Also

ll.order

Examples

```r
data(plodia)
sim1 <- ll.edm(sqrt(plodia), order=2, bandwidth = 1.5)
```

ll.order `Consistent nonlinear estimate of the order using local polynomial regression.`

Description

A function to estimate the order of a time series using the nonparametric order selection method of Cheng and Tong (1992, 1994) as modified by Yao & Tong (1994; see also Fan, Yao & Tong 1996). The method uses leave-one-out cross-validation of the locally linear regression against lagged-abundances.

Usage

```r
ll.order(x, order = 1:5, step = 1, deg = 2, bandwidth = c(seq(0.3, 1.5, by = 0.1), 2:10), cv = TRUE, echo = TRUE)
```

Arguments

- `x`: A time series without missing values.
- `order`: The candidate orders. The default is 1:5.
- `step`: The time step for prediction.
- `deg`: The degree of the local polynomial.
- `bandwidth`: The candidate bandwidths to be considered.
- `cv`: if TRUE leave-one-out cross-validation will be performed.
- `echo`: if TRUE a counter shows the progress.
Details

The time series is normalized prior to cross-validation.

A Gaussian kernel is used for the locally linear regression.

The bandwidth is optimized using cross-validation. If a single bandwidth is provided, no cross validation of bandwidth will be carried out. Highly nonlinear data will require more narrow bandwidths. If NA is returned it may be because the min bandwidth considered is too small relative to the density of data.

Missing values are NOT permitted.

If deg is set to 0, the order is estimated on the basis of the Nadaraya-Watson (locally constant) estimator of the conditional expectation against lagged-abundances (Cheng and Tong 1992, 1994).

Value

An object of class "ll.order" is returned consisting of the following components:

- grid: the grid of orders, bandwidths, and CV’s.
- grid$order: the orders.
- grid$cv: the cross-validation score across the grid of orders and bandwidths. (If cv = TRUE).
- grid$gcv: the generalized cross-validation score.
- grid$bandwidth: the bandwidths.
- grid$df: the degrees of freedom of the fitted model.
- order: the vector of orders considered.
- deg: The degree of the local polynomial.

References


lpx

Examples

```r
data(plodia)

fit <- ll.order(sqrt(plodia), order=3:3, bandwidth = seq(0.5, 1.5, by = 0.5))

## Not run: plot(fit)
summary(fit)
```

lpx

### Utility function

#### Description

hack to make ll.order work with locfit>1.5. not to be called by the user.

#### Usage

```r
lpx(x, nn = 0, h = 0, adpen = 0, deg = 2, acri = "none",
    scale = FALSE, style = "none")
```

#### Arguments

- `x`: ...
- `nn`: ...
- `h`: ...
- `adpen`: ...
- `deg`: ...
- `acri`: ...
- `scale`: ...
- `style`: ...

#### Details

not to be called by the user.

#### Author(s)

Catherine Loader
**Utility function**

**Description**

A function to create matrix of lagged time series. Called by various functions.

**Usage**

`mkx(x, lags)`

**Arguments**

- `x` A univariate time series.
- `lags` The vector of time lags.

**Details**

If `lags` is `c(1, 4)`, say, then the function returns a matrix that consist of columns `x(t-1), x(t-4), x(t)`. 

**Value**

A matrix of lagged abundances. The last column is the current

**Author(s)**

Upmanu Lall

**References**


---

**Time series of Meal Moth abundance**

**Description**

This is replicate 3 in Bjornstad et al. (1998).

**Usage**

`plodia`

**Format**

A vector containing 55 values
References

plot.contingency.periodogram

Plot contingency periodograms

Description
‘plot’ method for "contingency.periodogram" class object.

Usage

## S3 method for class 'contingency.periodogram'
plot(x, ...)

Arguments

x an object of class "contingency.periodogram", usually, as a result of a call to contingency.periodogram.

... generic plot arguments.

Value
A contingency periodogram is plotted. The line represents the critical value based on the chi-squared test (95%).

See Also

contingency.periodogram

plot.lin.order

Plot linear cross-validation for time-series order

Description
‘plot’ method for class "lin.order".

Usage

## S3 method for class 'lin.order'
plot(x, ...)

...
Arguments

x  an object of class "ll.order", usually, as a result of a call to \texttt{lin.order.cls} or \texttt{lin.order.mle}.

... generic plot arguments.

Value

A xy-plot of order against cross-validation error is produced.

See Also

\texttt{lin.order.cls}

\begin{Verbatim}
\textbf{plot.ll.order} \hspace{2cm} \textit{Plot nonparametric cross-validation for time-series order}
\end{Verbatim}

Description

'plot' method for class "ll.order".

Usage

\begin{verbatim}
## S3 method for class 'll.order'
plot(x, ...)
\end{verbatim}

Arguments

x  an object of class "ll.order", usually, as a result of a call to \texttt{ll.order}.

... generic plot arguments.

Details

See \texttt{ll.order} for details.

Value

A xy-plot of minimum cross-validation error against order is produced.

See Also

\texttt{ll.order}
`plot.lomb`  \hspace{1cm} \textit{Plot Lomb periodograms}

**Description**

‘plot’ method for objects of class "lomb".

**Usage**

```r
## S3 method for class 'lomb'
plot(x, ...)
```

**Arguments**

- `x`: an object of class "lomb", usually, as a result of a call to `spec.lomb`.
- `...`: generic plot arguments.

**Value**

A Lomb periodogram is composed of a xy-plot of amplitude against frequency.

**See Also**

`spec.lomb`

---

`plot.ppll`  \hspace{1cm} \textit{Plot function for prediction profile objects}

**Description**

‘plot’ method for class "ppll".

**Usage**

```r
## S3 method for class 'ppll'
plot(x, ...)
```

**Arguments**

- `x`: an object of class "ppll", usually, as a result of a call to `prediction.profile.ll`.
- `...`: generic plot arguments.

**Details**

See `prediction.profile.ll` for details.
Value

A xy-plot of one minus the cross-validation error (i.e. the prediction accuracy against prediction time step.

See Also

prediction.profile.nl

Description

'plot' method for class "specar.ci".

Usage

## S3 method for class 'specar.ci'
plot(x, period = TRUE, ...)

Arguments

x an object of class "specar.ci", usually, as a result of a call to specar.ci.
period if TRUE x-axis is period, if FALSE frequency.
... generic plot arguments.

Value

A xy-plot of amplitude against period (or frequency).

See Also

specar.ci
portman.Q

Ljung-Box test for whiteness in a time series.

Description

portman.Q uses the cumulative ACF to test for whiteness of a time series.

Usage

portman.Q(x, K)

Arguments

x
A time series (vector without missing values).

K
the maximum lag of the ACF to be used in the test.

Details

This is the Ljung-Box version of the Portemanteau test for whiteness (Tong 1990). It may in particular be useful to test for whiteness in the residuals from time series models.

Value

A vector is returned consisting of the asymptotic chi-square value, the associated d.f. and asymptotic p.val for the test of whiteness.

References


Examples

data(plodia)

portman.Q(sqrt(plodia), K = 10)

fit <- ar(sqrt(plodia))

portman.Q(na.omit(fit$resid), K = 10)
predict.ll.order  
*Predict values from ll.order object.*

**Description**
Calculates the leave-one-out predicted values for the optimal ll.order object

**Usage**
```r
## S3 method for class 'll.order'
predict(object, ...)```

**Arguments**
- `object` - an object of class "ll.order", usually, as a result of a call to `ll.order`.
- `...` - no other arguments currently allowed

**Details**
See `ll.order` for details.

**Value**
A data frame with observed and predicted values for the optimal ll-model is returned.

**See Also**
- `ll.order`

---

**prediction.profile.ll**  
*Nonlinear forecasting at varying lags using local polynomial regression.*

**Description**
A wrapper function around `ll.order` to calculate prediction profiles (a la Sugihara and May 1990 and Yao and Tong 1994). The method uses leave-one-out cross-validation of the local regression (with CV optimized bandwidth) against lagged-abundances at various lags.

**Usage**
```r
prediction.profile.ll(x, step = 1:10, order = 1:5, deg = 2, bandwidth = c(seq(0.3, 1.5, by = 0.1), 2:10))```
Arguments

- **x**
  A time series without missing values.

- **step**
  The vector of time steps for forward prediction.

- **order**
  The candidate orders. The default is 1:5.

- **deg**
  The degree of the local polynomial.

- **bandwidth**
  The candidate bandwidths to be considered.

Details

- see **llNorder** for details.

Value

An object of class "ppll" consisting of a list with the following components:

- **step**
  the prediction steps considered.

- **CV**
  the cross-validation error.

- **order**
  the optimal order for each step.

- **bandwidth**
  the optimal bandwidth for each step.

- **df**
  the degrees of freedom for each step.

References


See Also

- **llNorder**

Examples

```r
data(plodia)

fit <- prediction.profile.ll(sqrt(plodia), step=1:3, order=1:3,
                           bandwidth = seq(0.5, 1.5, by = 0.5))

## Not run: plot(fit)
```
### print.ll.order

*Print nonparametric cross-validation for time-series order*

**Description**

'print' method for class "ll.order".

**Usage**

```r
## S3 method for class 'll.order'
print(x, verbose = FALSE, ...)
```

**Arguments**

- **x**: an object of class "ll.order", usually, as a result of a call to `ll.order`.
- **verbose**: if TRUE provides a raw-printing of the object.
- **...**: no other arguments currently allowed

**Details**

See `ll.order` for details.

**Value**

A matrix summarizing the minimum cross-validation error (cv.min) and the associated Gaussian-kernel bandwidth (bandwidth.opt) and model degrees-of-freedom for each order considered.

**See Also**

`ll.order`

---

### spec.lomb

*The Lomb periodogram for unevenly sampled data*

**Description**

The function to estimate the Lomb periodogram for a spectral analysis of unevenly sampled data.

**Usage**

```r
spec.lomb(y = stop("no data arg"), x = stop("no time arg"), freq = NULL)
```
Arguments

- **y**: vector of length n representing the unevenly sampled time series.
- **x**: the a vector (of length n) representing the times of observation.
- **freq**: the frequencies at which the periodogram is to be calculated. If NULL the canonical frequencies (the Fourier frequencies) are used.

Details

This is the Lomb periodogram to test for periodicity in time series of unevenly sampled data. Missing values should be deleted in both x and y before execution.

Value

An object of class "lomb" is returned consisting of the following components:

- **freq**: the frequencies as supplied.
- **spec**: the estimated amplitudes at the different frequencies.
- **f.max**: the frequency of maximum amplitude.
- **per.max**: the corresponding period of maximum amplitude.
- **p**: the level of significance associated with the max period.

References


Examples

data(plodia)
y <- sqrt(plodia)
x <- 1:length(y)

#make some missing values
#omit NAs
y <- na.omit(y); x <- na.omit(x)

#the lomb p'gram
fit <- spec.lomb(y, x)
summary(fit)
## Not run: plot(fit)
Description

A function to estimate a "confidence interval" for the power spectrum and in particular a confidence interval for the dominant period. The function uses resampling of the autoregressive parameters to attain the estimate.

Usage

specar.ci(x, order, resamp = 500, nfreq = 100, echo = TRUE)

Arguments

x A time series without missing values.
order a scalar representing the order to be considered. If "aic" the order is be selected automatically using the AIC criterion.
resamp the number of resamples of the ar-coefficients from the covariance matrix.
nfreq the number of points at which to save the value for the power spectrum (and confidence envelope).
echo If TRUE, a counter for each run shows the progress.

Details

A "confidence interval" for the periodogram is obtained by resampling the ar-coefficients using the variance-covariance matrix from the ar.mle object.

If a zero'th order process is chosen by using the AIC criterion, a first order process will be used.

If the dynamics is highly nonlinear, the parametric estimate of the power spectrum may be inappropriate.

Value

An object of class "specar.ci" is returned consisting of the following components:

order the ar-order.
spectrum.freq the spectral frequencies.
spectrum.spec the estimated power-spectrum of the data.
resamp.spectrum gives the quantile summary for the resampling distribution of the spectral powers.
resamp.maxfreq the full vector of output for the resampled max.frequencies.

See Also

plot.specar.ci summary.specar.ci
Examples

data(plodia)

fit <- specar.ci(sqrt(plodia), order=3, resamp=10)

## Not run: plot(fit, period=FALSE)
summary(fit)

summary.lin.order

Summarize linear cross-validation for time-series order

Description

‘summary’ method for class "lin.order".

Usage

## S3 method for class 'lin.order'
summary(object, ...)

Arguments

object an object of class "lin.order", usually, as a result of a call to lin.order.cls or lin.order.mle.

... no other arguments currently allowed

Value

A slightly prettyfied version of the object is printed.

See Also

lin.order.cls
Summary

'Summary' method for class 'll.order'.

Usage

```r
## S3 method for class 'll.order'
summary(object, GCV = FALSE, ...)
```

Arguments

- `object`: an object of class "ll.order", usually, as a result of a call to `ll.order`.
- `GCV`: if TRUE (or if cross-validation was not done), uses GCV values.
- `...`: no other arguments currently allowed

Details

See `ll.order` for details.

Value

A matrix summarizing the minimum cross-validation error (cv.min) and the associated Gaussian-kernel bandwidth (bandwidth.opt) and model degrees-of-freedom for each order considered.

See Also

`ll.order`

Summary

'Summary' method for objects of class "lomb".

Usage

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

Description

'Summary' method for objects of class "lomb".

See Also

`lomb`
Arguments

- **object**: an object of class "lomb", usually, as a result of a call to `spec.lomb`.
- **...**: generic plot arguments.

Value

A list summarizing the analysis is printed:

- **period**: the dominant period.
- **p.val**: the p.value.

See Also

- `spec.lomb`

Description

‘summary’ method for objects of class "specar.ci".

Usage

```r
## S3 method for class 'specar.ci'
summary(object, period = TRUE, ...)
```

Arguments

- **object**: an object of class "specar.ci", usually, as a result of a call to `specar.ci`.
- **period**: If TRUE the summary is given in terms of the period, if false it is in terms of the frequency.
- **...**: generic plot arguments.

Value

A list summarizing the analysis is printed:

- **period**: the dominant period.
- **p.val**: the p.value.

See Also

- `specar.ci`
Index

*Topic datasets
  plodia, 10
*Topic misc
  lpx, 9
  mkx, 10
*Topic ts
  add.test, 2
  contingency.periodogram, 3
  lin.order.cls, 4
  lin.test, 5
  ll.edm, 6
  ll.order, 7
  plot.contingency.periodogram, 11
  plot.lin.order, 11
  plot.ll.order, 12
  plot.lomb, 13
  plot.ppll, 13
  plot.specar.ci, 14
  portman.Q, 15
  predict.ll.order, 16
  prediction.profile.ll, 16
  print.ll.order, 18
  spec.lomb, 18
  specar.ci, 20
  summary.lin.order, 21
  summary.ll.order, 22
  summary.lomb, 22
  summary.specar.ci, 23

add.test, 2

contingency.periodogram, 3, 11

lin.order.cls, 4, 12, 21
lin.test, 5
ll.edm, 6
ll.order, 4–7, 12, 16–18, 22
lpx, 9

mkx, 10

plodia, 10
plot.contingency.periodogram, 11
plot.lin.order, 11
plot.ll.order, 12
plot.lomb, 13
plot.ppll, 13
plot.specar.ci, 14, 20
portman.Q, 15
predict.ll.order, 16
prediction.profile.ll, 13, 14, 16
print.ll.order, 18
print.ppll (prediction.profile.ll), 16
spec.lomb, 13, 18, 23
specar.ci, 14, 20, 23
summary.lin.order, 21
summary.ll.order, 22
summary.lomb, 22
summary.specar.ci, 20, 23