# Package ‘tswge’

January 31, 2023

<table>
<thead>
<tr>
<th>Type</th>
<th>Package</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>Time Series for Data Science</td>
</tr>
<tr>
<td>Version</td>
<td>2.1.0</td>
</tr>
<tr>
<td>Date</td>
<td>2023-1-30</td>
</tr>
<tr>
<td>Author</td>
<td>Wayne Woodward</td>
</tr>
<tr>
<td>Maintainer</td>
<td>Bivin Sadler <a href="mailto:bsadler@smu.edu">bsadler@smu.edu</a></td>
</tr>
<tr>
<td>Imports</td>
<td>signal,PolynomF,MASS,waveslim,atsa,tidyverse,zoo,plotrix, dplyr, ggplot2, magrittr,nnfor,forecast</td>
</tr>
<tr>
<td>License</td>
<td>GPL-2</td>
</tr>
<tr>
<td>NeedsCompilation</td>
<td>no</td>
</tr>
<tr>
<td>LazyData</td>
<td>TRUE</td>
</tr>
<tr>
<td>Repository</td>
<td>CRAN</td>
</tr>
<tr>
<td>Date/Publication</td>
<td>2023-01-31 13:10:02 UTC</td>
</tr>
</tbody>
</table>

## R topics documented:

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>tswge-package</td>
<td>5</td>
</tr>
<tr>
<td>aic.ar.wge</td>
<td>6</td>
</tr>
<tr>
<td>aic.burg.wge</td>
<td>7</td>
</tr>
<tr>
<td>aic.wge</td>
<td>8</td>
</tr>
<tr>
<td>aic5.ar.wge</td>
<td>9</td>
</tr>
<tr>
<td>aic5.wge</td>
<td>10</td>
</tr>
<tr>
<td>airline</td>
<td>11</td>
</tr>
<tr>
<td>airlog</td>
<td>11</td>
</tr>
<tr>
<td>ample.spec.wge</td>
<td>12</td>
</tr>
<tr>
<td>appy</td>
<td>13</td>
</tr>
<tr>
<td>artrans.wge</td>
<td>13</td>
</tr>
<tr>
<td>backcast.wge</td>
<td>14</td>
</tr>
<tr>
<td>bat</td>
<td>15</td>
</tr>
</tbody>
</table>
R topics documented:

bitcoin ................................................. 16
Bsales .................................................. 16
bumps16 .................................................. 17
bumps256 ............................................... 17
butterworth.wge ...................................... 18
cardiac .................................................. 19
cement .................................................. 19
chirp .................................................... 20
co.wge .................................................. 21
dfw.2011 ............................................... 22
dfw.mon .................................................. 22
dfw.yr ................................................... 23
doppler ................................................... 24
doppler2 .................................................. 24
dow.annual ............................................ 25
dow.rate ................................................ 25
dow1000 ............................................... 26
dow1985 ................................................ 27
dowjones2014 ........................................ 27
deco.cd6 ............................................... 28
deco.corp.bond ....................................... 28
deco.mort30 .......................................... 29
est.ar.wge ............................................. 30
est.arma.wge .......................................... 31
est.farma.wge ......................................... 32
est.garma.wge ......................................... 33
est.glambda.wge ...................................... 34
expsmooth.wge ....................................... 35
factor.comp.wge ...................................... 36
factor.wge ............................................. 37
fig1.10a ............................................... 38
fig1.10b ............................................... 38
fig1.10c ............................................... 39
fig1.10d ............................................... 40
fig1.16a ............................................... 40
fig1.21a ............................................... 41
fig1.22a ............................................... 42
fig1.5 ................................................... 42
fig10.11x ............................................... 43
fig10.11y ............................................... 44
fig10.1bond ........................................... 44
fig10.1cd ............................................... 45
fig10.1mort ............................................ 46
fig10.3x1 ............................................... 46
fig10.3x2 ............................................... 47
fig11.12 ............................................... 48
fig11.4a ............................................... 48
fig12.1a ............................................... 49
R topics documented:

- fig 12.1b .................................................. 50
- fig 13.18a .................................................. 50
- fig 13.2c .................................................. 51
- fig 3.10d .................................................. 52
- fig 3.16a .................................................. 52
- fig 3.18a .................................................. 53
- fig 3.24a .................................................. 54
- fig 3.29a .................................................. 54
- fig 4.8a .................................................. 55
- fig 5.3c .................................................. 56
- fig 6.11a .................................................. 56
- fig 6.1nf .................................................. 57
- fig 6.2nf .................................................. 58
- fig 6.5nf .................................................. 58
- fig 6.6nf .................................................. 59
- fig 6.7nf .................................................. 60
- fig 6.8nf .................................................. 60
- fig 8.11a .................................................. 61
- fig 8.4a .................................................. 62
- fig 8.6a .................................................. 62
- fig 8.8a .................................................. 63
- flu ......................................................... 64
- fore.arima.wge ........................................... 64
- fore.arma.wge ........................................... 66
- fore.aruma.wge ........................................... 67
- fore.farma.wge ........................................... 68
- fore.garma.wge ........................................... 69
- fore.glambda.wge ......................................... 70
- fore.sigplusnoise.wge .................................... 71
- freeze ..................................................... 72
- freight .................................................... 73
- gegenb.wge .............................................. 73
- gen.arch.wge ............................................ 74
- gen.arima.wge ........................................... 75
- gen.arma.wge ............................................ 76
- gen.aruma.wge .......................................... 77
- gen.garch.wge ........................................... 78
- gen.garma.wge .......................................... 79
- gen.eg.wge ............................................. 80
- gen.glambda.wge ........................................ 81
- gen.sigplusnoise.wge ................................... 82
- global.temp ............................................. 83
- global2020 ............................................... 83
- hadley ..................................................... 84
- hilbert.wge .............................................. 85
- is.glambda.wge .......................................... 85
- is.sample.wge ........................................... 86
- kalman.miss.wge ........................................ 87
R topics documented:

kalman.wge .................................................. 88
kingkong .................................................. 89
lavon .................................................. 90
lavon15 .................................................. 90
linearchirp ............................................... 91
ljung.wge .................................................. 91
llynx .................................................. 92
lynx .................................................. 93
ma.pred.wge .............................................. 94
ma.smooth.wge ........................................... 95
ma2.table7.1 .............................................. 96
macoeff.geg.wge ........................................... 96
mass.mountain ............................................ 97
MedDays .................................................. 98
mm.eq .................................................. 98
mult.wge .................................................. 99
NAICS .................................................... 100
nbumps256 ............................................... 100
nile.min .................................................. 101
noctula .................................................. 102
NSA ..................................................... 102
ozona .................................................. 103
pacfts.wge ............................................... 103
parzen.wge ............................................... 104
patemp .................................................. 105
period.wge ............................................... 106
pi.weights.wge ........................................... 107
plotts.dwt.wge ........................................... 108
plotts.mra.wge ........................................... 109
plotts.parzen.wge ....................................... 110
plotts.sample.wge ....................................... 111
plotts.true.wge ......................................... 112
plotts.wge ............................................... 113
prob10.4 .................................................. 114
prob10.6x .................................................. 115
prob10.6y .................................................. 115
prob10.7x .................................................. 116
prob10.7y .................................................. 116
prob11.5 .................................................. 117
prob12.1c .................................................. 118
prob12.3a .................................................. 119
prob12.3b .................................................. 119
prob12.6c .................................................. 120
prob13.2 .................................................. 121
prob8.1a .................................................. 121
prob8.1b .................................................. 122
prob8.1c .................................................. 123
prob8.1d .................................................. 123
tswge-package

Description

These functions and data sets accompany the book "Applied Time Series Analysis with R, 2nd edition" by Woodward, Gray, and Elliott
Author(s)
Wayne Woodward <waynew@smu.edu>

References

Examples
data(wages)
plotts.wge(wages)

---

aic.ar.wge

AR Model Identification for AR models

Description
AR model identification using either AIC, AICC, or BIC and MLE, Burg or YW

Usage
aic.ar.wge(x, p = 1:5, type = "aic", method = "mle")

Arguments
- x: Realization to be analyzed
- p: Range of p values to be considered
- type: Type of model identification criterion: aic, aicc, or bic
- method: Method used for estimation: MLE, Burg, or YW

Value
- type: Criterion used: aic (default), aicc, or bic
- method: Estimation method used: MLE, Burg, or YW
- min_value: Value of the minimized criterion
- p: AR order for selected model
- phi: AR parameter estimates for selected model
- vara: White noise variance estimate for selected model

Author(s)
Wayne Woodward

References
Examples

```r
data(fig3.18a)
aic.ar.wge(fig3.18a,p=1:5,type='aicc',method='burg')
```

**Description**

AR model identification using either AIC, AICC, or BIC

**Usage**

```r
aic.burg.wge(x, p = 1:5, type = "aic")
```

**Arguments**

- `x` Realization to be analyzed
- `p` Range of p values to be considered
- `type` Type of model identification criterion: aic, aicc, or bic

**Value**

- `type` Criterion used: aic (default), aicc, or bic
- `min_value` Value of the minimized criterion
- `p` AR order for selected model
- `phi` AR parameter estimates for selected model
- `vara` White noise variance estimate for selected model

**Author(s)**

Wayne Woodward

**References**


**Examples**

```r
data(fig3.18a)
aic.burg.wge(fig3.18a,p=1:5,type='aicc')
```
ARMA model identification using either AIC, AICC, or BIC

Usage

```r
aic.wge(x, p = 0:5, q = 0:2, type = "aic")
```

Arguments

- `x`: Realization to be analyzed
- `p`: Range of p values to be considered
- `q`: Range of q values to be considered
- `type`: Type of model identification criterion: aic, aicc, or bic

Value

- `type`: Criterion used: aic (default), aicc, or bic
- `min_value`: Value of the minimized criterion
- `p`: AR order for selected model
- `phi`: AR parameter estimates for selected model
- `q`: MA order for selected model
- `theta`: MA parameter estimates for selected model
- `vara`: White noise variance estimate for selected model

Author(s)

Wayne Woodward

References


Examples

```r
data(fig3.18a)
aic.wge(fig3.18a, p=0:5, q=0:1, type='aicc')
```
Return top 5 AIC, AICC, or BIC picks for AR model fits

Description
You may select either AIC, AICC, or BIC to use model identification. You can also use ML, Burg, or Yule-Walker estimates. Given a range of values for p and q, the program returns the top 5 candidate models.

Usage
```r
aic5.ar.wge(x, p = 0:5, type = "aic", method = "MLE")
```

Arguments
- **x**: Realization to model
- **p**: Range of AR orders to be considered
- **type**: Either 'aic' (default), 'aicc', or 'bic'
- **method**: Either 'MLE' (default), 'Burg', or 'YW'

Value
A list of p, selected criterion for the top 5 models. The identification type and estimation method are printed on the output.

Note
If some model order combinations give explosively nonstationary models, then the program may stop prematurely. You may need to adjust the range of p and q to avoid these models.

Author(s)
Wayne Woodward

References

Examples
```r
data(fig3.18a)
aic5.wge(fig3.18a, p=0:5, q=0:2)
```
aic5.wge  

Return top 5 AIC, AICC, or BIC picks

Description

You may select either AIC, AICC, or BIC to use model identification. Given a range of values for p and q, the program returns the top 5 candidate models.

Usage

```r
aic5.wge(x, p = 0:5, q = 0:2, type = "aic")
```

Arguments

- `x`: Realization to model
- `p`: Range of AR orders to be considered
- `q`: Range of MA orders to be considered
- `type`: Either 'aic' (default, 'aicc', or 'bic')

Value

A list of p,q, and selected criterion for the top 5 models

Note

If some model order combinations give explosively nonstationary models, then the program may stop prematurely. You may need to adjust the range of p and q to avoid these models.

Author(s)

Wayne Woodward

References


Examples

```r
data(fig3.18a)
aic5.wge(fig3.18a,p=0:5,q=0:2)
```
**Description**

Monthly international airline passengers (in 1000s) from January 1949-December 1960. Series G in Box, Jenkings, and Reinsel text

**Usage**

```r
data("airline")
```

**Format**

The format is: num [1:144] 112 118 132 129 121 135 148 148 136 119 ... 

**Source**

"Time Series Analysis: Forecasting and Control" by Box, Jenkins, and Reinsel

**References**


**Examples**

```r
data(airline)
```

---

**Description**

Natural log of monthly international airline passengers (in 1000s) from January 1949-December 1960. Series G in Box, Jenkings, and Reinsel text

**Usage**

```r
data("airlog")
```

**Format**

The format is: num [1:144] 4.72 4.77 4.88 4.86 4.8 ... 

**Source**

"Time Series Analysis: Forecasting and Control" by Box, Jenkins, and Reinsel
ample.spec.wge

References


Examples

data(airlog)

ample.spec.wge

Smoothed Periodogram using Parzen Window

Description

This function calculates and optionally plots the smoothed periodogram using the Parzen window. The truncation point may be chosen by the user

Usage

ample.spec.wge(x, dbcalc = "TRUE", plot = "TRUE")

Arguments

x Vector containing the time series realization
dbcalc If dbcalc=TRUE, the calculation is in the log (dB) scale. If FALSE, then non-log calculations are made
plot If PLOT=TRUE then the smoothed spectral estimate is plotted. If FALSE then no plot is created

Value

freq The frequencies at which the smoothed periodogram is calculated
pzgram The smoothed periodogram using the Parzen window

Author(s)

Wayne Woodward

References


Examples

ample.spec.wge(rnorm(100))
appy

Non-perforated appendicitis data shown in Figure 10.8 (solid line) in Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott

Description
Annual non-perforated appendicitis rates for years 1970-2005

Usage
data("appy")

Format
The format is: num [1:36] 14.8 13.7 14.3 14.2 13 ...

Source
Alder, et al. (2010) Archives of Surgery 145, 63-71

References

Examples
data(appy)

artrans.wge

Perform Ar transformations

Description
Given a time series in the vector x, and AR coefs phi1 and phi2, for example, artrans.wge computes y(t)=x(t)-phi1X(t-1)-phi2x(t-2), for t=3, ..., n

Usage
artrans.wge(x,phi.tr, lag.max=25, plottr = "TRUE")

Arguments
x Vector containing original realization
phi.tr Coefficients of the transformation
lag.max Max lag (k) for sample autocorrelations
plottr If plot=TRUE then plots of the data, transformed data, and sample autocorrelations of original and transformed data
Value
Transformed data

Note
For a difference, use phi.tr=1

Author(s)
Wayne Woodward

References

Examples
```r
data(wtcrude)
difdata=artrans.wge(wtcrude,phi.tr=1,lag.max=30,plottr=TRUE)
```

Description
This function takes either a fitted (or true) model for the realization x and calculates the residuals using the backcasting procedure

Usage
```r
backcast.wge(x, phi = 0, theta = 0, n.back = 50)
```

Arguments
- **x**: realization
- **phi**: AR coefficients
- **theta**: MA coefficients
- **n.back**: Backcast to X(-n.back)

Value
The n backcast residuals are returned

Author(s)
Wayne Woodward
bat

References

Chapter 7 of Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott

Examples

```r
data(fig6.2nf)
backcast.wge(fig6.2nf, phi=c(1.2, -.6), theta=.5, n.back=50)
```

---

**bat**

*Bat echolocation signal shown in Figure 13.11a in Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott*

---

Description

Bat echolocation signal of a big brown bat

Usage

```r
data("bat")
```

Format

The format is: num [1:381] -0.0049 -0.0083 0.0127 0.0068 -0.0259 0.0059 0.0386 -0.0405 -0.0269 0.0474 ...

Source

Al Feng, Beckman Center of the University of Illinois

References


Examples

```r
data(bat)
```
**bitcoin**  
*Daily Bitcoin Prices From May 1, 2020 to April 30, 2021*

**Description**

This dataset contains the daily price of bitcoin from May 1, 2021 to April 30, 2021. The data was gathered from Yahoo Finance on April 30, 2020 and included missing values on October 9, 12 and 13 of 2020. Yahoo Finance has since filled in the correct values which can be compared with the imputed values described in the book.

**Usage**

```r
data("bitcoin")
```

**Format**

The format is: num [1:461] 7200.174 6985.470 7344.884 ...

**Source**

Yahoo Finance

**References**

"Practical Time Series for Data Scientists by Woodward, Sadler and Robertson"

**Examples**

```r
data(bitcoin)
```

---

**Bsales**  
*Toy Data Set of Business Sales Data*

**Description**

100 weeks of sales data with sales, TV advertising budget, Online advertising budget and the amount of a discount if any.

**Usage**

```r
data("Bsales")
```

**References**

The Time Series Toolkit
bumps16

Examples

data(Bsales)

16 point bumps signal

Description

Bumps signal from Donoho and Johnstone (1994) Biometrika 81,425-455

Usage

data("bumps16")

Format

The format is: num [1:16] 0.1 0.4 5.5 0.2 1.4 0.5 0.3 0.7 0.1 2.5 ...

Source

Donoho and Johnstone (1994) Biometrika 81,425-455

References


Examples

data(bumps16)

bumps256

256 point bumps signal

Description

Bumps signal from Donoho and Johnstone (1994) Biometrika 81,425-455

Usage

data("bumps256")

Format

The format is: num [1:256] 0.00016 0.00017 0.000182 0.000195 0.000211 ...
butterworth.wge

Source
Donoho and Johnstone(1994) Biometrika 81, 425-455

References

Examples
data(bumps256)

---

butterworth.wge Perform Butterworth Filter

Description
The user can specify the order of the filter, and whether it is low pass ("low"), high pass ("high"), band stop ("stop"), or band pass ("pass") filter. Requires the CRAN package 'signal'.

Usage
butterworth.wge(x, order, type, cutoff, plot=TRUE)

Arguments
- x: Realization to be filtered
- order: Order of the Butterworth filter
- type: Either "low", "high", "stop", or "pass" as discussed in Descriptions
- cutoff: For "low" and "high": cutoff is a real number. For "stop" and "band": cutoff is a 2-component vector
- plot: If plot=TRUE then plots of the original and filtered data are produced.

Value
The filtered data

Note
Requires CRAN package 'signal'

Author(s)
Wayne Woodward

References
Examples

```r
data(wages)
butterworth.wge(wages, order=4, type="low", cutoff=.05)
```

---

**cardiac**  
*Weekly Cardiac Mortality Data*

**Description**

Weekly cardiac mortality, temperatures, and pollution measures for the years 1970-1978

**Usage**

```r
data("cardiac")
```

**Format**

ts object consisting of weekly data

**Source**

Shumway and Stoffer, 1999)

**References**

"Time Series for Data Science: Analysis and Forecasting" by Woodward, Sadler, and Robertson

**Examples**

```r
data(cardiac)
```

---

**cement**  
*Cement data shown in Figure 3.30a in Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott*

**Description**

Quarterly usage of metric tons (in thousands) of Portland cement used from the first quarter of 1973 through the fourth quarter of 1993 in Australia

**Usage**

```r
data("cement")
```

**Format**

The format is: num [1:84] 1148 1305 1342 1452 1184 ...
Source

Australian Bureau of Statistics

References


Examples

data(cement)

chirp

Chirp data shown in Figure 12.2a in Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott

Description

256 point linear chirp data

Usage

data("chirp")

Format

The format is: List of 2

$x$ : num [1:256] 1 1 0.98 0.95 0.91 0.86 0.8 0.72 0.63 0.53 ... $spec$ : num

[1:256] 0.511 0.568 0.733 0.991 1.32 ...

Source

Simulated data

References


Examples

data(chirp)
Cochrane-Orcutt test for trend

Description

Performs the Cochrane-Orcutt to test for a linear trend in a time series realization.

Usage

co.wge(x,maxp=5)

Arguments

- **x**: Realization
- **maxp**: Maximum AR order allowed for AR model fit to residuals from least squares line

Value

- **z**: Residuals from the fitted line
- **b0hat**: Estimated y-intercept of the fitted line using the CO method
- **b1hat**: Estimated slope of the fitted line using the CO method
- **z.order**: Order, p, fit to the residuals
- **z.phi**: Coefficients of the AR model fit to the residuals
- **p.value**: P-value of the CO test for the significance of the slope
- **tco**: Cochrane-Orcutt test statistic.

Author(s)

Wayne Woodward

References


Examples

data(global.temp)
co.wge(global.temp,maxp=5)
**dfw.2011**  
*DFW Monthly Temperatures from January 2011 through December 2020*

**Description**

Monthly average temperatures at Dallas Ft. Worth (in Fahrenheit) from January 2011 through December 2020

**Usage**

```r
data("dfw.2011")
```

**Format**

ts object consisting of monthly data from January 1900 through December 2020

**Source**

https://www.weather.gov/fwd/dmotemp

**References**

"Time Series for Data Science: Analysis and Forecasting" by Woodward, Sadler, and Robertson

**Examples**

```r
data(dfw.2011)
```

---

**dfw.mon**  
*DFW Monthly Temperatures*

**Description**

Monthly average temperatures at Dallas Ft. Worth (in Fahrenheit) from January 1900 through December 2020

**Usage**

```r
data("dfw.mon")
```

**Format**

ts object consisting of monthly data from January 1900 through December 2020
**dfw.yr**

**Source**

https://www.weather.gov/fwd/dmotemp

**References**

"Time Series for Data Sience: Analysis and Forecasting" by Woodward, Sadler, and Robertson

**Examples**

```r
data(dfw.yr)
```
**doppler**

*Doppler Data*

**Description**

Generated Doppler data

**Usage**

```
data("doppler")
```

**Format**

The format is: num [1:2000] -0.00644 -0.01739 -0.02961 -0.04091 -0.04952 ...

**Source**

Simulated

**References**


**Examples**

```
data(doppler)
```

---

**doppler2**

*Doppler signal in Figure 13.10 in Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott*

**Description**

Doppler signal with two time-varying frequencies

**Usage**

```
data("doppler2")
```

**Format**

The format is: num [1:200] -0.372 1.246 -1.163 0.261 -0.698 ...

**Source**

Simulated data
**dow.annual**

**References**


**Examples**

```r
data(doppler2)
```

---

**dow.annual**

*DOW Annual Closing Averages*

**Description**

DOW Annual closing averages from 1915 through 2020

**Usage**

```r
data("dow.annual")
```

**Format**

*ts object consisting of DOW Annual closing averages from 19155 through 2020*

**References**

"Time Series for Data Sience: Analysis and Forecasting" by Woodward, Sadler, and Robertson

**Examples**

```r
data(dow.annual)
```

---

**dow.rate**

*DOW Daily Rate of Return Data*

**Description**

DOW daily rate of return data from October 1, 1928 to December 31, 2010

**Usage**

```r
data("dow.rate")
```

**Format**

The format is: num [1:20656] 240 238 238 240 240 ...

Source

Public access

References

"Applied Statistics and Data Analysis with R, 2nd edition" by Woodward, Gray, and Elliott

Examples

data(dow.rate)

data(dow1000)

---

**dow1000**  
*Dow Jones daily rate of return data for 1000 days*

Description

Dow Jones daily rate of return for the 1000 trading days before December 31, 2010.

Usage

data("dow1000")

Format

The format is: num [1:1001] 240 238 238 240 240 ...

Source

Internet and shown in Figure 4.9, "Applied Time Series Analysis with R, 2nd edition", by Woodward, Gray and Elliott

Examples

data(dow1000)
dow1985

**Daily DOW Closing Prices 1985 through 2020**

**Description**

Daily DOW Closing Prices 1985 through 2020

**Usage**

```r
data("dow1985")
```

**Format**


**References**

"Time Series for Data Science: Analysis and Forecasting" by Woodward, Sadler, and Robertson.

**Examples**

```r
data(dow1985)
```

dowjones2014

**Dow Jones daily averages for 2014**

**Description**

Daily Dow Jones averages for 2014

**Usage**

```r
data("dowjones2014")
```

**Format**

The format is: num [1:252] 16441 16470 16425 16531 16463 ...

**Source**

Economic Data: Federal Reserve Bank of St. Louis. Website: https://research.stlouisfed.org/fred2/series/DJIA/downloaddata

**References**


**Examples**

```r
data(dowjones2014)
```
**eco.cd6**

*6-month rates*

**Description**

6-month rates 1/1/1991 through 4/1/2010

**Usage**

```r
data("eco.cd6")
```

**Format**

The format is: num [1:469] 7.25 7.53 7.64 7.64 7.59 7.44 7.39 7.26 7.25 7.19 ...

**Source**

Internet

**References**


**Examples**

```r
data(eco.cd6)
```

---

**eco.corp.bond**

*Corporate bond rates*

**Description**

Corporate bond rates 1/1/1991 through 4/1/2010

**Usage**

```r
data("eco.corp.bond")
```

**Format**

The format is: num [1:469] 4.61 5.22 5.69 6.04 6.06 5.91 5.43 5.04 4.89 4.26 ...

**Source**

Internet
eco.mort30

References


Examples

data(eco.corp.bond)

data(eco.mort30)

---

eco.mort30  30 year mortgage rates

Description

30-year mortgage rates 1/1/1991 through 4/1/2010

Usage

data("eco.mort30")

Format

The format is: num [1:469] 7.31 7.43 7.53 7.6 7.7 7.69 7.63 7.55 7.48 7.44 ...

Source

Internet

References


Examples

data(eco.mort30)
Estimate parameters of an AR(p) model

Description
Estimate parameters of an AR(p) with p assumed known. Outputs residuals (backcast0 and white noise variance estimate.)

Usage
est.ar.wge(x, p = 2, factor = TRUE, method = "mle")

Arguments
x Realization
p AR order
factor If TRUE (default) a factor table is printed for the estimated model
method Either "mle" (default), "burg", or "yw"

Details
The 'type' argument is added for backwards compatibility and if specified will replace the value specified in the 'method' argument.

Value
method Estimation method used: MLE, Burg, or YW
phi.est Estimates of the AR parameters
res Estimated residuals (using backcasting) based on estimated model
avar Estimated white noise variance (based on backcast residuals)
xbar Sample mean of data in x
aic AIC for estimated model
aicc AICC for estimated model
bic BIC for estimated model

Author(s)
Wayne Woodward

References

Examples
data(fig6.1nf)
est.ar.wge(fig6.1nf,p=1)
**est.arma.wge**

*Function to calculate ML estimates of parameters of stationary ARMA models*

---

**Description**

This function calculates ML estimates, computes residuals (using backcasting), estimates white noise variance for a stationary ARMA model

**Usage**

```r
est.arma.wge(x, p = 0, q = 0, factor = TRUE)
```

**Arguments**

- `x`: The realization.
- `p`: The autoregressive order
- `q`: the moving average order
- `factor`: Logical variable. factor=TRUE (default) plots a factor table for estimated AR-part of model

**Details**

This function uses arima from base SAS and is written similarly to itsmr function arma

**Value**

- `phi`: ML estimates of autoregressive parameters
- `theta`: ML estimates of moving average parameters
- `res`: Residuals (calculated using backcasting)
- `avar`: Estimate of white noise variance based on backcast residuals
- `se.phi`: Standard errors of the AR parameter estimates
- `se.theta`: Standard errors of the MA parameter estimates
- `aic`: AIC for estimated model
- `aicc`: AICC for estimated model
- `bic`: BIC for estimated model

**Note**

Requires CRAN package 'itsmr'. The program is based on arima from base R and arma from 'itsmr'

**Author(s)**

Wayne Woodward
References

Examples

```r
data(fig6.2nf)
est.arma.wge(fig6.2nf,p=2,q=1)
est.farma.wge
```

```
est.farma.wge  Estimate the parameters of a FARMA model.
est.farma.wge(x, low.d, high.d, inc.d, p.max, nback = 500)
```

Description

This function uses the grid search algorithm discussed in Section 11.5 of Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott

Usage

```
est.farma.wge(x, low.d, high.d, inc.d, p.max, nback = 500)
est.farma.wge
```

Arguments

- `x`: Realization to be analyzed
- `low.d`: The lower limit for d in the grid search
- `high.d`: The upper limit for d in the grid search
- `inc.d`: The increment, e.g. .01, .001, etc. in the grid search
- `p.max`: Maximum value of p allowed for the AR component of the model
- `nback`: Number of backcasts to be used (see section 11.5 in Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott

Details

We assume q=0 and do not allow moving average terms in the model.

Value

- `d`: Estimate of d
- `phi`: Estimates of the pth order AR component of the model where p is some integer from 0 to p.max
- `vara`: The estimated white noise variance
- `aic`: The aic value associated with the final model

Author(s)

Wayne Woodward
References


Examples

est.farma.wge(Nile, low.d=0.1, high.d=0.5, inc.d=0.01, p.max=3)

---

est.garma.wge  Estimate the parameters of a GARMA model.

Description

This function uses the grid search algorithm discussed in Section 11.5 of Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott.

Usage

est.garma.wge(x, low.u, low.lambda, high.u, high.lambda, inc.u, inc.lambda, p.max, nback=500)

Arguments

x Realization to be analyzed
low.u The lower limit for u in the grid search
low.lambda The lower limit for lambda in the grid search
high.u The upper limit for u in the grid search
high.lambda The upper limit for lambda in the grid search
inc.u The increment, e.g. .01, .001, etc. in the grid search on possible u values
inc.lambda The increment, e.g. .01, .001, etc. in the grid search on possible lambda values
p.max Maximum value of p allowed for the AR component of the model
nback Number of backcasts to be used (see section 11.5 in Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott)

Details

We assume q=0 and do not allow moving average terms in the model.

Value

u Estimate of u
lambda Estimate of lambda
phi Estimates of the pth order AR component of the model where p is some integer from 0 to p.max
vara The estimated white noise variance
aic The aic value associated with the final model
**Author(s)**
Wayne Woodward

**References**


**Examples**

data(llynx)
est.garma.wge(llynx, low.u=.4, high.u=.9, low.lambda=.2, high.lambda=.4, inc.u=.01, inc.lambda=.1, p.max=1)

---

**est.glambda.wge**

*Estimate the value of lambda and offset to produce a stationary dual.*

**Description**

This function uses the technique discussed in Section 13.3.3 of Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott to find the g(lambda) time transformation that most nearly transforms the data to a stationary dual.

**Usage**

est.glambda.wge(data, lambda.range = c(0, 1), offset.range = c(0, 100))

**Arguments**

dataVector containing the TVF realization to be analyzed
lambda.rangeRange of lambda values considered in the search
offset.rangeRange of offset values considered in the search

**Value**

Q A listing of lambda values within the range and offsets for each lambda that provided the best dual. Also a listing of the test statistic, Q, to be minimized
best.lambda See description of best.offset below
best.offset best.lambda and best.offset are the lambda-offset pair that produced the most stationary dual according to the Q criterion

**Author(s)**
Wayne Woodward

**References**

expsmooth.wge

Examples

data(ss08)

est.glambda.wge(ss08, lambda.range=c(-1,1), offset.range=c(0,100))

---

**Description**

Performs exponential smoothing on the data in vector x

**Usage**

expsmooth.wge(x, alpha=NULL, n.ahead=0, plot=TRUE)

**Arguments**

- `x`: Vector containing realization
- `alpha`: Alpha value
- `n.ahead`: Number of steps ahead to forecast
- `plot`: If plot=TRUE then plots of the data along with forecasts

**Value**

- `alpha`: alpha value used in the smoothing
- `u`: forecasts

**Author(s)**

Wayne Woodward

**References**

"Time Series for Data Science" by Woodward, Sadler, and Robertson

**Examples**

data(wtcrude2020)
expsmooth.wge(wtcrude2020)
factor.comp.wge  

Create a factor table and AR components for an AR realization

Description

This program finds the ML estimates of a specified order, then prints a factor table for the estimated model and prints and plots the additive components

Usage

factor.comp.wge(x, aic = FALSE, p, ncomp)

Arguments

x  Realization
aic  The program calls basic R function phi.burg to calculate burg estimates of an AR fit to the data. Aic is turned off and the user specifies the order
p  Order of AR to fit to data
ncomp  Number of additive components to calculate and plot

Value

ncomp  The number of additive components
x.comp  Matrix (i,j) where i designates the component and j denotes time, i.e. (i,j) denotes the ith component at time j

Author(s)

Wayne Woodward

References


Examples

data(ss08)
factor.comp.wge(ss08, p=9, ncomp=4)
factor.wge

Produce factor table for a kth order AR or MA model

Description

This program produces a factor table that reduces a kth order factor into its first and irreducible second order factors as described in Section 3.2.11 of "Applied Time Series Analysis" by Woodward, Gray, and Elliott

Usage

factor.wge(phi=0, theta=0)

Arguments

phi Vector containing the coefficients of the kth order AR factor which is to be factored
theta Vector containing the coefficients of the kth order MA factor which is to be factored

Value

The only output is the factor table, written by default to the console

Author(s)

Wayne Woodward

References

"Applied Time Series Analysis, 2nd edition" by Woodward, Gray, and Elliott

Examples

factor.wge(phi=c(-.3,.44,.29,-.378,-.648))
**fig1.10a**  
*Simulated data shown in Figure 1.10a in Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott*

**Description**  
This is the sum of the three signals in fig1.10b, fig1.10c, and fig1.10d

**Usage**  
```r
data("fig1.10a")
```

**Format**  
The format is: num [1:1000] 0.0217 -0.1528 -0.3141 -0.4613 -0.5934 ...

**Source**  
Simulated data

**References**  

**Examples**  
```r
data(fig1.10a)
```

---

**fig1.10b**  
*Simulated data shown in Figure 1.10b in Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott*

**Description**  
Low frequency component of Figure 1.10a

**Usage**  
```r
data("fig1.10b")
```

**Format**  
The format is: num [1:1000] 1 1 0.999 0.998 0.997 ...

**Source**  
Simulated data
Simulated data in Figure 1.10c in Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott

Description

Middle frequencies component in Figure 1.10a

Usage

data("fig1.10c")

Format

The format is: num [1:1000] 0.73 0.646 0.56 0.471 0.381 ...

Source

Simulated data

References


Examples

data(fig1.10c)
fig1.10d

Simulated data in Figure 1.10d in Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott

Description
High frequency component of Figure 1.10a

Usage
data("fig1.10d")

Format
The format is: num [1:1000] -1.71 -1.8 -1.87 -1.93 -1.97 ...

Source
Simulated data

References

Examples
data(fig1.10d)

fig1.16a

Simulated data for Figure 1.16a in Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott

Description
Data containing two dominant frequencies

Usage
data("fig1.16a")

Format
The format is: num [1:250] -0.89 -3.209 0.929 -0.763 -1.972 ...

Source
Simulated data
References


Examples

data(fig1.16a)

---

Description

Simulated shown in Figure 1.21a of Woodward, Gray, and Elliott text. It illustrates the fact that frequency information is displayed better in the spectrum than the autocorrelations.

Usage

data("fig1.21a")

Format

The format is: num [1:250] -0.89 -3.209 0.929 -0.763 -1.972 ...

Source

Simulated by the authors of the Woodward, Gray, and Elliott text

References


Examples

data(fig1.21a)
White noise data

Description
Realization of length n=250 of white noise data. Figure 1.22a in "Applied Time Series Analysis with R, 2nd edition" by Woodward, Gray, and Elliott

Usage
data("fig1.22a")

Format
The format is: num [1:250] 0.302 -0.691 -0.477 0.814 -0.267 ...

Source
Simulated data

References

Examples
data(fig1.22a)

Simulated data shown in Figure 1.5 in Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott

Description
Simulated data from an ergodic AR(1) process

Usage
data("fig1.5")

Format
The format is: num [1:100] 0.739 -0.39 0.15 -0.627 0.262 ...

Source
Simulated data
References


Examples

data(fig10.11x)

Description

Simulated unobservable AR(1) data in Example 10.11

Usage

data("fig10.11x")

Format

The format is: num [1:75] -0.2497 -0.0812 -0.6463 -1.7653 -2.719 ...

Source

Simulated data

References


Examples

data(fig10.11x)
**fig10.11y**  
*Simulated data shown in Figure 10.11 (dashed line) in Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott*

**Description**  
Simulated observed AR(1) plus noise data in Example 10.11

**Usage**  
data("fig10.11y")

**Format**  
The format is: num [1:75] -0.74 0.045 -0.775 -2.944 -2.278 ...

**Source**  
Simulated data

**References**  

**Examples**  
data(fig10.11y)

---

**fig10.1bond**  
*Data for Figure 10.1b in Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott*

**Description**  
Moody’s seasoned Aaa corporate bond rate, January 1, 1991-April1, 2010

**Usage**  
data("fig10.1bond")

**Format**  
The format is: num [1:232] 7.17 6.51 6.5 6.16 6.03 6.26 6.25 5.79 5.6 5.32 ...

**Source**  
Internet
fig10.1cd

References


Examples

data(fig10.1bond)

---

**Description**

6 month CD rate for January 1, 1991 - April 1, 2010

**Usage**

data("fig10.1cd")

**Format**

The format is: num [1:232] 9.04 8.83 8.93 8.86 8.86 9.01 9 8.75 8.61 8.55 ...

**Source**

Internet

**References**


**Examples**

data(fig10.1cd)
**fig10.1mort**  
*Data shown in Figure 10.1c in Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott*

**Description**

30 year conventional mortgage rates: January 1, 1991-April 1, 2010

**Usage**

```r
data("fig10.1mort")
```

**Format**


**Source**

Internet

**References**


**Examples**

```r
data(fig10.1mort)
```

---

**fig10.3x1**  
*Variable X1 for the bivariate realization shown in Figure 10.3"

**Description**

Variable X1 for the bivariate Var(1) realization in Figure 10.3 of "Applied Time Series Analysis with R, 2nd edition" by Woodward, Gray, and Elliott

**Usage**

```r
data("fig10.3x1")
```

**Format**

The format is: num [1:75] -0.0757 -0.2728 -0.8089 -2.4747 -5.9256 ...

**Source**

Simulated Var(1) data
References


Examples

data(fig10.3x1)

---

**Description**

Variable X2 for the bivariate realization in Figure 10.3 of "Applied Time Series Analysis with R, 2nd edition" by Woodward, Gray, and Elliott

**Usage**

data("fig10.3x2")

**Format**

The format is: num [1:75] 0.646 -1.313 -0.191 -2.61 -4.925 ...

**Source**

Simulated Var(1) data

**References**


**Examples**

data(fig10.3x2)
**fig11.12**  
*Data shown in Figure 11.12a in Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott*

**Description**  
Simulated GATMA(1,0) data

**Usage**  
data("fig11.12")

**Format**  
The format is: num [1:500] 2.18 -1.17 -3.13 -1.32 1.69 ...

**Source**  
Simulated data

**References**  

**Examples**  
data(fig11.12)

---

**fig11.4a**  
*Data shown in Figure 11.4a in Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott*

**Description**  
Simulated FARMA(2,0) data

**Usage**  
data("fig11.4a")

**Format**  
The format is: num [1:100] 1.361 -0.369 0.881 2.362 0.236 ...

**Source**  
simulated data
References


Examples

data(fig11.4a)

data(fig12.1a)

Description

Simulated two-frequency data in which the two frequencies are separated in time

Usage

data("fig12.1a")

Format

The format is: num [1:200] -1.22 -6.06 -9.66 -10.14 -8.58 ...

Source

Simulated data

References


Examples

data(fig12.1a)
| fig12.1b | Simulated data with two frequencies shown in Figure 12.1b in Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott |
| fig13.18a | Simulated data shown in Figure 3.18a in Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott |

**Description**

Simulated two-frequency AR(4) data

**Usage**

data("fig12.1b")

data("fig13.18a")

**Format**

The format is: num [1:256] 10.081 10.835 0.532 -5.495 1.294 ...

The format is: num [1:400] 1.251 1.0019 -0.0317 -1.0167 -1.4222 ...

**Source**

Simulated data

**References**


**Examples**

data(fig12.1b)

data(fig13.18a)
\texttt{fig13.2c}  

\textbf{Source}

Simulated data

\textbf{References}


\textbf{Examples}

\begin{verbatim}
  data(fig13.18a)
\end{verbatim}

\begin{verbatim}

\texttt{fig13.2c}  \hspace{1cm} TVF data shown in Figure 13.2c in Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott

\textbf{Description}

Realization from an Euler(2) model

\textbf{Usage}

\begin{verbatim}
  data("fig13.2c")
\end{verbatim}

\textbf{Format}

The format is: num [1:200] -13.14 -11.03 22.06 -8.92 -16.67 ...

\textbf{Source}

Simulated data

\textbf{References}


\textbf{Examples}

\begin{verbatim}
  data(fig13.2c)
\end{verbatim}
Description

AR(2) Realization $(1-.95)^2X(t)=a(t)$ plotted in Figure 3.10d in "Applied Time series Analysis with R, 2nd edition" by Woodward, Gray, and Elliott

Usage

data("fig3.10d")

Format

The format is: num [1:100] 15.3 16.3 18.6 21.2 22.8 ...

Details

This realization is also used in Chapter 7 of text above for testing estimation techniques

Source

Simulated realization

References


Examples

data(fig3.10d)

Description

Realization from the AR(3) model in Figure 3.16a

Usage

data("fig3.16a")

Format

The format is: num [1:200] -0.0686 0.4304 0.4786 0.9899 3.4047 ...
Description
Realization from the AR(3) model in Figure 3.18a

Usage
data("fig3.18a")

Format
The format is: num [1:200] -0.573 -0.837 -1.16 1.078 -0.561 ...

Source
Simulated data

References

Examples
data(fig3.18a)
**fig3.24a ARMA(2,1) realization**

**Description**

ARMA(2,1) realization of length n=200 phi(1)=1.6,phi(2)=-.9,theta(1)=.8 (using Box-Jenkins-Reinsel notation)

**Usage**

data("fig3.24a")

**Format**

The format is: num [1:200] 0.685 -1.234 -0.714 0.796 -0.96 ...

**Source**

Simulated data

**References**

Fig3.24a in "Applied Time Series Analysis with R, 2nd edition" by Woodward, Gray, and Elliott

**Examples**

data(fig3.24a)

---

**fig3.29a Simulated data shown in Figure 3.29a in Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott**

**Description**

Simulated data from stationary seasonal model

**Usage**

data("fig3.29a")

**Format**

The format is: num [1:20] -7.23 -6.99 -6.9 -6.26 -3.79 ...

**Source**

Simulated data
References


Examples

data(fig3.29a)

---

**fig4.8a**

_Gaussian White Noise_

**Description**

Gaussian White Noise, n=1000 shown in Figure 4.8a in "Applied Time Series Analysis with R, 2nd edition" by Woodward, Gray, and Elliott

**Usage**

`data("fig4.8a")`

**Format**

The format is: num [1:1000] -0.585 0.177 0.284 -0.271 0.126 ...

**Source**

Simulated data

**References**


**Examples**

`data(fig4.8a)`
fig5.3c

Data from Figure 5.3c in "Applied Time Series Analysis with R, 2nd edition" by Woodward, Gray, and Elliott

Description
Realization of length 200 from the AR(3) model \((1-0.995B)(1-1.2B+0.8B^2)X(t)=a(t)\)

Usage
```r
data("fig5.3c")
```

Format
The format is: num [1:200] -0.503 -0.811 -0.188 1.34 2.982 ...

Source
Simulated data

References

Examples
```r
data(fig5.3c)
```

---

fig6.11a

Cyclical Data

Description
First 50 points of data in Figure 6.11a, Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott

Usage
```r
data("fig6.11a")
```

Format
The format is: num [1:50] -0.682 0.15 2.262 3.079 4.122 ...

Source
Simulated
References


Examples

data(fig6.11a)

Description

Realization from the AR(1) model \((1-.8B)(X(t)-25)=a(t)\) in Figure 6.2 and also shown in Table 6.1 of "Applied Time Series Analysis with R, 2nd edition" by Woodward, Gray, and Elliott

Usage

data("fig6.1nf")

Format

The format is: num [1:80] 25.1 27.1 27.3 25.7 23.9 ...

Source

Generated data

References


Examples

data(fig6.1nf)
fig6.2nf

Data in Figure 6.2 without the forecasts

Description
Realization from the ARMA(2,1) model \((1-1.2B+.6B^2)(X(t)-50)=(1-.5B)a(t)\) in Figure 6.2 and also shown in Table 6.1 of "Applied Time Series Analysis with R, 2nd edition" by Woodward, Gray, and Elliott

Usage
data("fig6.2nf")

Format
The format is: num [1:25] 49.5 51.1 50 49.7 50.4 ...

Source
Generated data

References

Examples
data(fig6.2nf)

fig6.5nf

Data in Figure 6.5 without the forecasts

Description
Realization from the ARIMA(0,1,0) model for realization in Figure 6.5 of "Applied Time Series Analysis with R, 2nd edition" by Woodward, Gray, and Elliott

Usage
data("fig6.5nf")

Format
The format is: num [1:50] 105 104 103 102 102 ...
Source

Generated data

References


Examples

data(fig6.6nf)

---

**fig6.6nf**  
*Data in Figure 6.6 without the forecasts*

Description

Realization from the ARIMA(1,1,0) model $(1-.8B)(1-B)X(t)=a(t)$ for realization in Figure 6.6 of "Applied Time Series Analysis with R, 2nd edition" by Woodward, Gray, and Elliott

Usage

data("fig6.6nf")

Format

The format is: num [1:50] 139 138 138 140 141 ...

Source

Generated data

References


Examples

data(fig6.6nf)
Data in Figure 6.2 without the forecasts

**Description**
Realization from the ARIMA(0,2,0) model for realization in Figure 6.7 of "Applied Time Series Analysis with R, 2nd edition" by Woodward, Gray, and Elliott

**Usage**
data("fig6.7nf")

**Format**
The format is: num [1:50] -582 -579 -578 -578 -579 ...

**Source**
Generated data

**References**

**Examples**
data(fig6.7nf)

Simulated seasonal data with $s=12$

**Description**
Simulated seasonal data designed for showing seasonal forecasts

**Usage**
data("fig6.8nf")

**Format**
The format is: num [1:48] 5.8 13.66 9.83 7.33 6.96 ...

**Source**
Simulated Data
References


Examples

```r
data(fig6.8nf)
```

---

**Description**

Realization of length n=200 from the model \((1-B)(1-1.79B+1.75B^2-1.61B^3+.765B^4)X(t)=a(t)\)

**Usage**

```r
data("fig8.11a")
```

**Format**

The format is: num [1:200] 83.2 80.9 78.9 80.4 85.4 ...

**Source**

Simulated data

**References**

Applied time series Analysis with R, second edition by Woodward, Gray, and Elliott

**Examples**

```r
data(fig8.11a)
```
Data for Figure 8.4a in Applied time series Analysis with R, second edition by Woodward, Gray, and Elliott

Description
Realization of length n=200 from the model (1-.8B)(1-1.6B+.995B^2)X(t)=a(t)

Usage
data("fig8.4a")

Format
The format is: num [1:200] 13.45 -5.52 -19 -21.26 -13.63 ...

Source
simulated data

References
Applied time series Analysis with R, second edition by Woodward, Gray, and Elliott

Examples
data(fig8.4a)

Data for Figure 8.6a in Applied time series Analysis with R, second edition by Woodward, Gray, and Elliott

Description
The realization of length n=200 is from the model (1-B)^2(1-1.2B+.6B^2)X(t)=a(t)

Usage
data("fig8.6a")

Format
The format is: num [1:200] 354 368 383 399 417 ...

Source
Simulated data
References

Applied time series Analysis with R, second edition by Woodward, Gray, and Elliott

Examples

data(fig8.6a)

data(fig8.8a)

Description

Realization of length \( n=200 \) from the model \((1-B^{12})(1-1.25B+.9B^2)(X(t)-50)=a(t)\)

Usage

data("fig8.8a")

Format

The format is: num [1:200] 48.9 42.9 49.3 57.3 55.5 ...

Source

Simulated data

References

Applied time series Analysis with R, second edition by Woodward, Gray, and Elliott

Examples

data(fig8.8a)
Influenza data shown in Figure 10.8 (dotted line)

Description
Annual influenza rate for years 1970-2005

Usage
data("flu")

Format
The format is: num [1:36] 9.75 5.82 10.99 10.41 8.42 ...

Source
Alder, et al. (2010) Archives of Surgery 145, 63-71

References

Examples
data(flu)

fore.arima.wge  Function for forecasting from known model which may have (1-B)^d and/or seasonal factors

Description
This function calculates forecasts from a known model that may have stationary ARMA components as well as (1-B)^d and/or seasonal factors

Usage
fore.arima.wge(x,phi=0,theta=0,d=0,s=0,n.ahead=5,lastn=FALSE,plot=TRUE,alpha=.05,limits)
fore.arima.wge

Arguments

- **x**: Realization to be forecast from
- **phi**: Vector containing stationary AR parameters
- **theta**: Vector containing MA parameters
- **d**: Order of difference
- **s**: Seasonal order
- **n.ahead**: Number of steps ahead to forecast
- **lastn**: Logical, lastn=TRUE plots forecasts for the last n.ahead values in the realization
- **plot**: Logical, plot=TRUE plots forecasts
- **alpha**: Significance level for prediction limits
- **limits**: Logical, limits=TRUE plots prediction limits

Value

- **f**: Vector of forecasts
- **ll**: Lower limits
- **ul**: Upper limits
- **resid**: Residuals
- **wnv**: White noise variance estimate
- **xbar**: Sample mean of data in x
- **se**: Se for each forecast
- **psi**: Psi weights
- **ptot**: Total order of all AR components, phi, d, and s
- **phtot**: Coefficients after multiplying all stationary and nonstationary components on the AR side of the equation

Author(s)

Wayne Woodward

References


Examples

data(airline)
x=log(airline)
phi12=c(-.36,-.05,-.14,-.11,.04,.09,-.02,.02,.17,.03,-.1,-.38)
s=12
d=1
fore.arima.wge(x,phi=phi12,d=1,s=12,n.ahead=12,limits=FALSE)
Description

Forecasts and associated plots for an ARMA model

Usage

```r
fore.arma.wge(x, phi=0, theta=0, n.ahead=5, lastn=FALSE, plot=TRUE, alpha=.05, limits=TRUE)
```

Arguments

- `x`: Realization
- `phi`: AR vector
- `theta`: MA vector
- `n.ahead`: Number of steps ahead
- `lastn`: Logical variable, TRUE means plot forecast for last n.ahead values of realization
- `plot`: Logical variable, TRUE means plot forecasts
- `alpha`: Significance level for prediction limits
- `limits`: Logical variable, TRUE means plot limits

Value

- `f`: Vector of forecasts
- `ll`: Lower limits
- `ul`: Upper limits
- `resid`: Residuals
- `wnv`: White noise variance estimate
- `xbar`: Sample mean of data in x
- `se`: Se for each forecast
- `psi`: psi weights
- `rmse`: RMSE is output if lastn=TRUE
- `mad`: MAD is output if lastn=TRUE

Author(s)

Wayne Woodward

References

fore.aruma.wge

**Examples**

```r
data(fig6.1nf)
fore.aruma.wge(fig6.1nf, phi=.8, n.ahead=20)
```

---

**fore.aruma.wge**  
Function for forecasting from known model which may have $(1-B)^d$, seasonal, and/or other nonstationary factors

---

**Description**

This function calculates forecasts from a known model that may have stationary ARMA components as well as $(1-B)^d$, seasonal, and/or other nonstationary factors.

**Usage**

```r
fore.aruma.wge(x, phi=0, theta=0, d=0, s=0, lambda=0, n.ahead=5, lastn=FALSE, plot=TRUE, alpha=.05, limits=TRUE)
```

**Arguments**

- `x`: Realization to be forecast from
- `phi`: Vector containing stationary AR parameters
- `theta`: Vector containing MA parameters
- `d`: Order of difference
- `s`: Seasonal order
- `lambda`: Vector containing coefficients of nonstationary factors not covered by the difference or the seasonal factors
- `n.ahead`: Number of steps ahead to forecast
- `lastn`: Logical, lastn=TRUE plots forecasts for the last n.ahead values in the realization
- `plot`: Logical, plot=TRUE plots forecasts
- `alpha`: Alpha for prediction limits
- `limits`: Logical, limits=TRUE plots prediction limits

**Value**

- `f`: Vector of forecasts
- `ll`: Lower limits
- `ul`: Upper limits
- `resid`: Residuals
- `wnv`: White noise variance estimate
- `xbar`: Sample mean of data in x
- `se`: Se for each forecast
psi
 Psi weights
ptot.fore
 Total order of all AR components, phi, d, s, and lambda
phtot.fore
 Coefficients after multiplying all stationary and nonstationary components on the
 AR side of the equation

Author(s)
Wayne Woodward

References

Examples

```r
data(airline)
  x=log(airline)
  phi12=c(-.36,-.05,-.14,-.11,.04,.09,-.02,.02,.17,.03,-.1,-.38)
  s=12
  d=1
  fore.aruma.wge(x,phi=phi12,d=1,s=12,n.ahead=12,limits=FALSE)
```

fore.farma.wge
 Forecast using a FARMA model

Description
Find forecasts using a specified FARMA model

Usage

```r
fore.farma.wge(x, d, phi, theta = 0, n.ahead = 10, lastn = TRUE, plot = TRUE)
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>x</td>
<td>Realization to be analyzed</td>
</tr>
<tr>
<td>d</td>
<td>Parameter d in FARMA model</td>
</tr>
<tr>
<td>phi</td>
<td>Coefficients of the AR component of the FARMA model</td>
</tr>
<tr>
<td>theta</td>
<td>Coefficients of the MA component of the FARMA model</td>
</tr>
<tr>
<td>n.ahead</td>
<td>Number of values to forecast</td>
</tr>
<tr>
<td>lastn</td>
<td>If lastn=TRUE then the last n.ahead values are forecast. Otherwise, if lastn=FALSE the next n.ahead values are forecast</td>
</tr>
<tr>
<td>plot</td>
<td>If plot=TRUE then plots of the data and forecasts are plotted</td>
</tr>
</tbody>
</table>

Details
Forecasts for an AR model fit to the data are also calculated and optionally plotted
Value

- `ar.fit.order` Order of the AR model fit to the data
- `ar.fore` Forecasts based on the AR model
- `farma.fore` Forecasts based on the FARMA model

Author(s)
Wayne Woodward

References


Examples

```r
fore.farma.wge(Nile, d=.37, phi=0, theta = 0, n.ahead = 30, lastn = TRUE, plot = TRUE)
```

Description

Find forecasts using a specified GARMA model

Usage

```r
fore.garma.wge(x,u,lambda,phi,theta=0,n.ahead=10,lastn=TRUE,plot=TRUE)
```

Arguments

- `x` Realization to be analyzed
- `u` Parameter u in GARMA model
- `lambda` Parameter lambda in GARMA model
- `phi` Coefficients of the AR component of the GARMA model
- `theta` Coefficients of the MA component of the GARMA model
- `n.ahead` Number of values to forecast
- `lastn` If lastn=TRUE then the last n.ahead values are forecast. Otherwise, if lastn=FALSE the next n.ahead values are forecast
- `plot` If plot=TRUE then plots of the data and forecasts are plotted

Details

Forecasts for an AR model fit to the data are also calculated and optionally plotted
Value

- **ar.fit.order**: Order of the AR model fit to the data
- **ar.fore**: Forecasts based on the AR model
- **garma.fore**: Forecasts based on the GARMA model

Author(s)

Wayne Woodward

References


Examples

```r
data(llynx)
fore.garma.wge(llynx,u=.796,lambda=.4,phi=.51,theta=0,n.ahead=30,lastn=TRUE,plot=TRUE)
```

---

**fore.glambda.wge**  
**Forecast using a G(lambda) model**

Description

Find forecasts using a specified G(lambda) model

Usage

```r
fore.glambda.wge(data.orig=lambda=0,offset=60,phi=0,h=0,n.ahead=10,lastn=TRUE,plot=TRUE)
```

Arguments

- **data.orig**: Time series data in the original time scale
- **lambda**: The value of lambda under the Box-Cox time transformation with parameter lambda.
- **offset**: Offset (or shift) value in the G(lambda) model.
- **phi**: Coefficients of the AR component of the AR model fit to the dual data
- **h**: Value of h which will be calculated to produce the desired number of forecasts in the original time scale
- **n.ahead**: Number of values to forecast
- **lastn**: If lastn=TRUE then the last n.ahead values are forecast. Otherwise, if lastn=FALSE the next n.ahead values are forecast
- **plot**: If plot=TRUE then plots of the data and forecasts are plotted
**Details**

Forecasts for an AR model fit to the data in the original time scale are also calculated and optionally plotted.

**Value**

- \texttt{f.ar}
  - Forecasts using AR model fit to data in original time
- \texttt{f.glam}
  - Forecasts using AR model fit to the dual and then reinterpolated

**Author(s)**

Wayne Woodward

**References**


**Examples**

```r
data(fig13.2c)
fore.glambda.wge(fig13.2c,lambda=-.4,offset=63,phi=c(0.93,-0.32,-0.15,-0.15,-0.17),n.ahead=30)
```

---

**Description**

Forecast models of the form line plus AR noise or cosine plus AR noise with known frequency

**Usage**

```r
fore.sigplusnoise.wge(x,linear=TRUE,method="mle",freq=0,max.p=5,
                      n.ahead=10,lastn=FALSE,plot=TRUE,alpha=.05,limits=TRUE)
```

**Arguments**

- \texttt{x}
  - The variable containing the realization to be analyzed
- \texttt{linear}
  - If TRUE then the program forecasts a line plus noise model. If FALSE the model is cosine plus noise
- \texttt{method}
  - Estimation method
- \texttt{freq}
  - Frequency of the cosine term. \texttt{freq} is ignored when using line plus noise
- \texttt{max.p}
  - Max value of \texttt{p} for the ARp model fit to the noise
- \texttt{n.ahead}
  - The number of steps ahead to forecast
- \texttt{lastn}
  - If TRUE then the function forecasts the last \texttt{n.ahead} values of the realization. If FALSE the forecasts are for \texttt{n.ahead} steps beyond the end of the realization
- \texttt{plot}
  - If TRUE then the forecasts and realization are plotted
- \texttt{alpha}
  - Significance level
- \texttt{limits}
  - If TRUE the forecast limits calculated and plotted
Value

- **f**: The n.ahead forecasts
- **ll**: The lower limits for the forecasts. Zeros are returned if limits were not requested
- **ul**: The upper limits for the forecasts. Zeros are returned if limits were not requested
- **res**: Residuals
- **wnv**: The estimated white noise variance based on the residuals
- **se**: `se` is the estimated standard error of the k step ahead forecast. Zeros are returned if limits were not requested
- **xi**: `xi` is the kth psi weight associated with the fitted AR model and used to calculate the `se` above. Note that psi0 is 1. Zeros are returned if limits were not requested

Author(s)

Wayne Woodward

References


Examples

data(lynx)
lynx.for=fore.sigplusnoise.wge(lynx,linear=FALSE,freq=.1,max.p=5,n.ahead=20)

freeze  Minimum temperature data

Description

Each data value represents the minimum temperature over 10-day period at a location in South America

Usage

data("freeze")

Format

The format is: `num [1:500] 8.2 12.3 9.2 8.4 10 8.8 6.8 4.8 5.2 1.7 ...`

Source

Unknown

References

**freight**

**Examples**

```r
data(freeze)
```

---

<table>
<thead>
<tr>
<th>freight</th>
<th>Freight data</th>
</tr>
</thead>
</table>

**Description**

9 years of monthly freight shipment data

**Usage**

```r
data("freight")
```

**Format**

The format is: num [1:120] 1299 1148 1345 1363 1374 ...

**Source**

Unknown

**References**


**Examples**

```r
data(freight)
```

---

<table>
<thead>
<tr>
<th>gegenb.wge</th>
<th>Calculates Gegenbauer polynomials</th>
</tr>
</thead>
</table>

**Description**

Calculates Gegenbauer polynomials of order n with parameters u and lambda - see (11.9) in Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott

**Usage**

```r
gegenb.wge(u, d, n)
```
Arguments


n  Order of Gegenbauer polynomial in (11.9)

Details

This function is called by gen.garma.wge

Value

The coefficients of the nth order Gegenbauer polynomial

Author(s)

Wayne Woodward

References


Examples

gegenb.wge(u=.8,d=.3,n=6)

description

Generate a realization from an ARCH(q0) model

Usage

gen.arch.wge(n, alpha0, alpha, plot = TRUE, sn = 0)

Arguments

n  Length of realization to be generated

alpha0  The constant alpha0 in model (4.23)

alpha  A vector of length q0 containing alpha1 through alphan0

plot  If plot=TRUE (default) the generated realization is plotted

sn  determines the seed used in the simulation. sn=0 produces new/random realization each time. sn=positive integer produces same realization each time
Value
returns the generated realization

Author(s)
Wayne Woodward

References

Examples

```r
gen.arch.wge(n=200,alpha0=.1,alpha=c(.36,.27,.18,.09))
```

---

**gen.arima.wge** Function to generate an ARIMA (or ARMA) realization

Description
This function calls arima.sim but with more simple parameter structure for stationary ARIMA (or ARMA) models

Usage

```r
gen.arima.wge(n, phi=0, theta=0, d=0, s=0, mu=0, vara=1, plot=TRUE, sn=0)
```

Arguments
- **n** Length of realization to be generated
- **phi** Vector of AR coefficients
- **theta** Vector of MA coefficients
- **d** Order of the difference
- **s** Seasonal order
- **vara** White noise variance, default=1
- **mu** Theoretical mean of data in x, default=0
- **plot** Logical: TRUE=plot, FALSE=no plot
- **sn** determines the seed used in the simulation. sn=0 produces new/random realization each time. sn=positive integer produces same realization each time

Value
This function simply generates and (optionally plots) an ARIMA (or ARMA) realization
Author(s)
Wayne Woodward

References

Examples
```r
gen.arima.wge(n=100, phi=c(1.6,-.9), theta=.8, d=1, vara=1, plot=TRUE)
```

Description
This function calls arima.sim but with more simple parameter structure for stationary ARMA models

Usage
```r
gen.arima.wge(n, phi=0, theta=0, mu=0, vara=1, plot=TRUE, sn=0)
```

Arguments
- `n` Length of realization to be generated
- `phi` Vector of AR coefficients
- `theta` Vector of MA coefficients
- `vara` White noise variance, default=1
- `mu` Theoretical mean, default=0
- `plot` Logical: TRUE=plot, FALSE=no plot
- `sn` Determines the seed used in the simulation. sn=0 produces new/random realization each time. sn=positive integer produces same realization each time

Value
This function simply generates and (optionally plots) an ARMA realization

Author(s)
Wayne Woodward

References
**Examples**

```r
gen.aruma.wge(n=100, phi=c(1.6,-.9), theta=.8, mu=50, vara=1, plot=TRUE)
```

---

**Description**

This function calls arima.sim but in a similar manner to gen.ns.arma.wge and gen.ns.arima.wge but allows for generation of realizations from ARUMA models (see Chapter 5 of "Applied Time Series Analysis with R, 2nd edition" by Woodward, Gray, and Elliott)

**Usage**

```r
gen.aruma.wge(n,phi=0,theta=0,d=0,s=0,lambda=0,vara=1,plot=TRUE,sn=0)
```

**Arguments**

- `n` Length of realization to be generated
- `phi` Vector of AR coefficients
- `theta` Vector of MA coefficients
- `d` Order of the difference
- `s` Order of seasonal operator
- `lambda` Vector of nonstationary coefficients not associated with `d` or `s` (see Def. 5.1(b) in Woodward, Gray, and Elliott text)
- `vara` White noise variance, default=1
- `plot` Logical: TRUE=plot, FALSE=no plot
- `sn` determines the seed used in the simulation. `sn`=0 produces new/random realization each time. `sn`=positive integer produces same realization each time

**Value**

This function generates and (optionally plots) an ARMA or ARIMA or ARUMA realization

**Author(s)**

Wayne Woodward

**References**


**Examples**

```r
gen.aruma.wge(n=100,phi=.7,theta=0, d=1, s=4,lambda=c(1.8,-1),vara=1, plot=TRUE)
```
gen.garch.wge  Generate a realization from a GARCH(p0,q0) model

Description
Generates a realization of length n from the GARCH(p0,q0) model (4.26) in "Applied Time Series Analysis with R, 2nd edition" by Woodward, Gray, and Elliott

Usage
gen.garch.wge(n, alpha0, alpha, beta, plot=TRUE, sn=0)

Arguments

n  Length of realization to be generated
alpha0  The constant alpha0 in model (4.23)
alpha  A vector of length q0 containing alpha1 through alphaq0
beta  A vector of length p0 containing beta1 through betap0
plot  If plot=TRUE (default) the generated realization is plotted
sn  determines the seed used in the simulation. sn=0 produces new/random realization each time. sn=positive integer produces same realization each time

Value
returns the generated realization

Author(s)
Wayne Woodward

References

Examples
gen.garch.wge(n=200, alpha0=.1, alpha=.45, beta=.45)
Function to generate a GARMA realization

Description

This function calls gen.geg.wge and arima.sim

Usage

```r
gen.garma.wge(n, u, lambda, phi = 0, theta = 0, trun = 300000, burn_in = 600, vara = 1, plot = TRUE, sn = 0)
```

Arguments

- `n`: the realization length to be generated
- `u`: Parameter u in the GARMA model given in (11.16) of Woodward, Gray, and Elliott text
- `lambda`: Parameter lambda in the GARMA model given in (11.16) of Woodward, Gray, and Elliott text
- `phi`: vector of AR parameters of ARMA part of GARMA model
- `theta`: vector of MA parameters of ARMA part of GARMA model using signs as given in the Woodward, Gray, and Elliott text
- `trun`: the truncation point of the infinite GLP form
- `burn_in`: is the burning-in period for the simulation
- `vara`: White noise variance, default=1
- `plot`: Logical: TRUE=plot, FALSE=no plot
- `sn`: determines the seed used in the simulation. sn=0 produces new/random realization each time. sn=positive integer produces same realization each time

Value

This function generates and (optionally plots) an GARMA realization

Author(s)

Wayne Woodward

References


Examples

```r
gen.garma.wge(n=100, u=.8, lambda=.4, phi=.9)
```
Description

This function calls macoef.wge

Usage

gen.geg.wge(n, u, lambda, trun = 30000, vara=1 ,sn = 0)

Arguments

n  the realization length to be generated  
u  Parameter u in the Gegenbauer model given in (11.12) of Woodward, Gray, and Elliott text
lambda  Parameter lambda in the Gegenbauer model given in (11.12) of Woodward, Gray, and Elliott text
trun  the truncation point of the infinite GLP form
vara  White noise variance, default=1
sn  determines the seed used in the simulation. sn=0 produces new/random realization each time. sn=positive integer produces same realization each time

Details

This function is called by gen.garma.wge and does not have a burn-in time. Thus, we recommend using est.garma.wge for generating realizations from a Gegenbauer model.

Value

This function generates a Gegenbauer realization

Author(s)

Wayne Woodward

References


Examples

gen.geg.wge(n=100, u=.8,lambda=.4)
Function to generate a \( g(\lambda) \) realization

Description

This function generates a \( g(\lambda) \) TVF realization as discussed in Chapter 13 of Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott.

Usage

\[
\text{gen.glambda.wge}(n, \lambda, \phi = 0, \text{offset} = 20, \text{vara} = 1, \text{plot} = \text{TRUE, sn} = 0)
\]

Arguments

- \( n \): Length of realization to be generated
- \( \lambda \): The lambda involved in the \( g(\lambda) \) time transformation - see Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott
- \( \phi \): Vector of AR coefficients
- \( \text{vara} \): White noise variance, default=1
- \( \text{offset} \): The offset parameter in a \( g(\lambda) \) process. See section 13.2 in Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott
- \( \text{plot} \): Logical: TRUE=plot, FALSE=no plot
- \( \text{sn} \): determines the seed used in the simulation. \( \text{sn}=0 \) produces new/random realization each time. \( \text{sn}=\text{positive integer} \) produces same realization each time

Value

This function simply generates and (optionally plots) an ARMA realization.

Author(s)

Wayne Woodward

References


Examples

\[
\text{gen.glambda.wge}(n=500, \lambda=0.5, \phi=c(1.9,-.99), \text{vara}=1, \text{plot}=\text{TRUE, sn}=0)
\]
Generate data from a signal-plus-noise model

Description
Generate a realization from the model \( x(t) = \text{coef}[1] \cdot \cos(2\pi \cdot \text{freq}[1] \cdot t + \text{psi}[1]) + \text{coef}[2] \cdot \cos(2\pi \cdot \text{freq}[2] \cdot t + \text{psi}[2]) + a(t) \)

Usage
\[
\text{gen.sigplusnoise.wge}(n, b0, b1=0, \text{coef}, \text{freq}, \text{psi}, \phi=0, \text{vara}=1, \text{plot}=\text{TRUE}, \text{sn}=0)
\]

Arguments
- \( n \): length of realization to be generated
- \( b0 \): y intercept of the linear component
- \( b1 \): slope of the linear component
- \( \text{coef} \): a 2-component vector specifying the coefficients (if only one cosine term is desired define \( \text{coef}[2]=0 \))
- \( \text{freq} \): a 2-component vector specifying the frequency components (0 to .5)
- \( \text{psi} \): a 2-component vector specifying the phase shift (0 to 2\( \pi \))
- \( \phi \): a vector of coefficients of the coefficients of the AR noise
- \( \text{vara} \): \( \text{vara} \) is the variance of the noise. NOTE: \( a(t) \) is a vector of \( N(0,\text{WNV}) \) noise generated within the function (default=1)
- \( \text{plot} \): if TRUE then plot the data generated (default=TRUE)
- \( \text{sn} \): determines the seed used in the simulation (default=0 indicating new realization each time). \( \text{sn} \)=positive integer, then the same realization is generated each time

Value
- \( x \): realization generated

Author(s)
Wayne Woodward

References

Examples
\[
\text{x}=\text{gen.sigplusnoise.wge}(n=100, \text{coef}=c(3,1), \text{freq}=c(.1,.4), \text{psi}=c(0,0), \text{vara}=2)
\]
**global.temp**  
*Global Temperature Data: 1850-2009*

**Description**

Annual temperature anomalies from the average for the years 1850-2009

**Usage**

```r
data("global.temp")
```

**Format**

The format is: List of 2  
$ year : num [1:160] 1850 1851 1852 1853 1854 ... $  
$ annual: num [1:160] -0.447 -0.292 -0.294 -0.337 -0.307 -0.321 -0.406 -0.503 -0.513 -0.349 ...`

**Source**

Climatic Research Unit at East Anglia, England, in conjunction with the Met Office Hadley Centre

**References**


**Examples**

```r
data(global.temp)
```

---

**global2020**  
*Global Temperature Data: 1880-2009*

**Description**

Annual temperature anomalies from the average for the years 1850-2009

**Usage**

```r
data("global.temp")
```

**Format**

The format is: ts file containing annual temperatures from 1880 through 2020

**Source**

ncdc.noaa.gov
References

"Time Series for Data Science: Analysis and Forecasting" by Woodward, Sadler, and Robertson

Examples

data(global2020)

hadley  Global temperature data

Description

Global temperature data for 1850-2009. The data are temperature anomalies, i.e. departures from the average for 1850-2009

Usage

data("hadley")

Format

The format is: num [1:160] -0.447 -0.292 -0.294 -0.337 -0.307 -0.321 -0.406 -0.503 -0.513 -0.349 ...

Source

Met Office Hadley Centre

References


Examples

data(hadley)
hilbert.wge

Function to calculate the Hilbert transformation of a given real valued signal (even length)

Description

Function is used with the tswge function wv.wge

Usage

hilbert.wge(input)

Arguments

input  realization to be analyzed

Value

ans  Hilbert transformation of the input

Author(s)

Wayne Woodward

References


Examples

data(airline)
hilbert.wge(airline)

is.glambdawge

Instantaneous spectrum

Description

Calculates instantaneous spectrum (in dB) based on a G(\lambda) time transformation

Usage

is.glambdawge(n, phi = 0, sigma2 = 1, lambda, offset)
is.sample.wge

Arguments

n               Length of realization.
phi             Coefficients of AR model fit to dual data.
sigma2          White noise variance
lambda          Lambda in the G(lambda) time transformation used
offset          Offset in the G(lambda) time transformation used

Value

Simply a plot of the realization

Author(s)

Wayne Woodward

References


Examples

is.glambd.wge(n=200,phi=c(.93,-.32,-.15,-.15,-.17),lambda=-.4,offset=63)

is.sample.wge               Sample instantaneous spectrum based on periodogram

Description

Calculates sample instantaneous spectrum (in dB) based on a G(lambda) time transformation

Usage

is.sample.wge(data, lambda, offset)

Arguments

data             Realization to be analyzed.
lambda           Lambda in the G(lambda) time transformation used
offset           Offset in the G(lambda) time transformation used

Value

Simply a plot of the realization

Author(s)

Wayne Woodward
References


Examples

data(ss08)
is.sample.wge(data=ss08,lambda=-.4,offset=63)

kalman.miss.wge

Kalman filter for simple signal plus noise model with missing data

Description

Kalman function to predict, filter, and smooth in the presence of missing data; see Section 10.6 4 in Applied Time Series Analysis with R

Usage

kalman.miss.wge(y,start, gam0, F, gamV, Gtmiss, gamW)

Arguments

y the univariate data set to be analyzed
start the scalar version of X(0) in item (c) following the state equation (10.47) of the text

Gam0 the scalar version of Gamma(0) discussed in item (c) following the state equation
F scalar version of the matrix F in the state equation

GamV the value Gamma(v) specified in item (b) following the state equation
Gtmiss specifies which items that are missing
gamW the variance of the (univariate) white noise denoted by Gamma(w) in item (c) following (10.48)

Value

pfs a table giving results such as those in Table 10.1 in Woodward, Gray, and Elliott book

Note

Calls Ksmooth1 in CRAN package 'astsa'

Author(s)

Wayne Woodward
References


Examples

data(table10.1.signal)
data(table10.1.noise)
spn = table10.1.signal + table10.1.noise
n = 75
Gtmiss = array(1, dim = c(1, 1, n))
Gtmiss[1, 1, 2] = 0
Gtmiss[1, 1, 5] = 0
kalman.miss.wge(y = spn, start = 0, gam0 = 1, F = .9, gamV = 1, Gtmiss, gamW = .75)

kalman.wge
Kalman filter for simple signal plus noise model

Description

Kalman filter program to predict, filter, and smooth related to the material in Section 10.6 4 in Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott

Usage

kalman.wge(y, start, gam0, F, gamV, G, gamW)

Arguments

y the univariate data set to be analyzed
start the scalar version of Xo in item (c) following the state equation (10.47) of the text
gam0 the scalar version of Gamma(0) discussed in item (c) following the state equation
F scalar version of the matrix F in the state equation
gamV the value Gamma(v) specified in item (b) following the state equation
G the scalar observation matrix specified in the observation equation as G(t)
gamW the variance of the (univariate) white noise denoted by Gamma(w) in item (c) following (10.48)

Value

pfs a table giving results such as those in Table 10.1 in Woodward, Gray, and Elliott book

Note

Requires CRAN package 'astsa'
kingkong

Author(s)

Wayne Woodward

References


Examples

data(table10.1.signal)
data(table10.1.noise)
spn=table10.1.signal+table10.1.noise
kalman.wge(y=spn,start=0,gam0=1,F=.9,gamV=1,G=1,gamW=.75)

---

kingkong King Kong Eats Grass

Description

Digitized record taken at 8,000 Hz of voltage readings obtained from the acoustical energy generated by Wayne Woodward speaking the words "King Kong eats grass" while a fan was blowing in the background

Usage

data("kingkong")

Format

The format is: num [1:15418] -0.001831 -0.000916 -0.003357 -0.002716 -0.000977 ...

Source

See description above

References


Examples

data(kingkong)
lavon

**Lavon lake water levels**

**Description**
Data given in feet above sea level. Quarterly data, 1982-2009

**Usage**
data("lavon")

**Format**
The format is: num [1:112] 495 492 500 491 492 ...

**Source**
http://lavon.uslakes.info/levelcal.asp

**References**

**Examples**
data(lavon)

lavon15

**Lavon Lake Levels to September 30, 2015**

**Description**
Feet above sea level for Lavon Lake, quarterly data through September 2015. An extension of data lavon

**Usage**
data("lavon15")

**Format**
The format is: num [1:135] 495 492 500 491 492 ...

**Source**
Lake Data internet
linearchirp

Examples

data(lavon15)

---

linearchirp  Linear chirp data.

Description

256 point linear chirp data, the first 150 points of which are shown in Figure 3.16(a) Time Series Analysis for Data Science: Analysis and Forecasting by Woodward, Sadler, and Robertson

Usage

data("linearchirp")

Format

The format is: List of 2 $ x: num [1:256] 1 1 0.98 0.95 0.91 0.86 0.8 0.72 0.63 0.53 ... $ spec: num [1:256] 0.511 0.568 0.733 0.991 1.32 ...

Source

Simulated data

References

Time Series Analysis for Data Science: Analysis and Forecasting by Woodward, Sadler, and Robertson

Examples

data(linearchirp)

---

ljung.wge  Ljung-Box Test

Description

Performs Ljung-Box Test for white noise

Usage

ljung.wge(x, K = 24, p = 0, q = 0)
Arguments

x  Realization to assess for white noise
K  Maximum lag for sample autocorrelations to be used in test
p  If x is a realization of residuals from an ARMA(p,q) fit then p=AR order. Otherwise, p=0
q  If x is a realization of residuals from an ARMA(p,q) fit then q=MA order. Otherwise, q=0

Value

test  Name of test for output: Ljung-Box Test
K    Maximum lag : same as input value
chi.square  Value of chi-square statistic
df   Degrees of freedom = K-p-q
pvalue  pvalue for testing null hypothesis of white noise

Author(s)

Wayne Woodward

References


Examples

data(fig1.22a)
ljung.wge(fig1.22a, K=24, p=0, q=0)

___

**Illynx**  
*Log (base 10) of lynx data*

___

Description

The log (base 10) of the annual number of lynx trapped in the Mackenzie River district of the North-West Canada (dataset lynx in this package)

Usage

data("Illynx")

Format

The format is: Time-Series [1:114] from 1821 to 1934: 2.43 2.51 2.77 2.94 3.17 ...
lynx

Source


References


Examples

data(lynx)

 lynx                  Lynx data

Description

The lynx data are the annual number of lynx trapped in the Mackenzie River district of Canada

Usage

data("lynx")

Format

The format is: Time-Series [1:114] from 1821 to 1934: 269 321 585 871 1475 ...

Source


References


Examples

data(lynx)
ma.pred.wge  Predictive or rolling moving average

Description
Given a time series in the vector x and order (either an odd or even integer) ma.pred.wge computes a predictive moving average giving 1-step ahead predictions through x(n+1). Optionally, you can specify k-step ahead forecasts beyond the end of the data.

Usage
ma.pred.wge(x, order=3, n.ahead=1, plot=TRUE)

Arguments
- x  Vector containing original realization
- order  Order (odd or even integer) of moving average predictor (default=3)
- n.ahead  Number of steps ahead to forecast beyond the end of the data (default=1)
- plot  If plot=TRUE then plots of the data and moving average predictors are plotted

Value
- x  Original data
- pred  Data file showing 1-step ahead predictors up to x(k.nahead)
- order  Order (odd or even integer) of the moving average predictor

Author(s)
Wayne Woodward

References
"Practical Time Series Analysis with R" by Woodward, Sadler, and Robertson"

Examples
data(wtcrude)
sm=ma.pred.wge(x=wtcrude, order=5, n.ahead=10)
**ma.smooth.wge**  

*Centered Moving Average Smoother*

**Description**

Given a time series in the vector `x` and order (either an odd or even integer) `ma.smooth.wge` computes a centered moving average smoother and optionally plots the data and smoothed data.

**Usage**

```r
ma.smooth.wge(x, order=3, plot=TRUE)
```

**Arguments**

- `x`: Vector containing original realization
- `order`: Order (odd or even integer) of moving average smoother
- `plot`: If plot=TRUE then plots of the data and smoothed data are plotted

**Value**

- `x`: Original data
- `smooth`: Data after application of centered average filter
- `order`: Order (odd or even integer) of the smoother

**Author(s)**

Wayne Woodward

**References**

"Practical Time Series Analysis with R" by Woodward, Sadler, and Robertson

**Examples**

```r
data(wtcrude)
sm=ma.smooth.wge(x=wtcrude, order=5)
```
ma2.table7.1  

*Simulated MA(2) data*

**Description**

This realization is used to obtain the innovations estimates shown in Table 7.1

**Usage**

```r
data("ma2.table7.1")
```

**Format**

The format is: `num [1:400] 1.299 1.831 -0.162 -0.648 1.243 ...`

**Source**

Simulated data

**References**


**Examples**

```r
data(ma2.table7.1)
```

macoef.geg.wge

*Calculate coefficients of the general linear process form of a Gegenbauer process*

**Description**

Calculate coefficients of the general linear process form of a Gegenbauer process based on formula (8), page 6 of Ferrara and Guegan(2001).

**Usage**

```r
macoef.geg.wge(u, lambda, trun = 300000)
```

**Arguments**

- `u` The value of u in the Gegenbauer model
- `lambda` The value of lambda in the Gegenbauer model
- `trun` The truncation point of the infinite GLP form
**Details**

This function is called by gen.geg.wge

**Value**

A vector of length trun containing the GLP coefficients

**Author(s)**

Wayne Woodward

**References**


**Examples**

```r
mageg=macoef.geg.wge(u=.8,lambda=.3)
```

---

**mass.mountain**

**Massachusetts Mountain Earthquake Data**

**Description**

Lg wave from from an earthquake known as Massachusetts Mountain Earthquake(5 August 1971), which was recorded at the Mina Nevada station

**Usage**

```r
data("mass.mountain")
```

**Format**

The format is: num [1:454] -0.03655 -0.01774 0.00218 0.01193 0.00915 ...

**Source**


**References**


**Examples**

```r
data(mass.mountain)
```
MedDays

Median days a house stayed on the market

Description
Median days a house stayed on the market between July 2016 and April 2020

Usage
data("MedDays")

Format
ts object consisting of monthly data from July 2016 through April 2020

References
"Time Series for Data Science: Analysis and Forecasting" by Woodward, Sadler, and Robertson

Examples
data(MedDays)

mm.eq

Massachusetts Mountain Earthquake data shown in Figure 13.13a in Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott

Description
Lg wave for Massachusetts Mountain Earthquake

Usage
data("mm.eq")

Format
The format is: num [1:454] -0.03655 -0.01774 0.00218 0.01193 0.00915 ...

Source

References
Description

The function multiplies the AR (or MA) factors of a model to produce the model in unfactored form. Requires the CRAN package 'PolynomF'.

Usage

\[
\text{mult.wge}(\text{fac1} = 0, \text{fac2} = 0, \text{fac3} = 0, \text{fac4} = 0, \text{fac5} = 0, \text{fac6} = 0)
\]

Arguments

- fac1: First factor to be multiplied
- fac2: Second factor to be multiplied
- fac3: Third factor to be multiplied (you may use a maximum of 6 factors)
- fac4: Fourth factor to be multiplied (you may use a maximum of 6 factors)
- fac5: Fifth factor to be multiplied (you may use a maximum of 6 factors)
- fac6: Sixth factor to be multiplied (you may use a maximum of 6 factors)

Value

- char.poly: The characteristics polynomial of the full model

Note

Requires CRAN package 'PolynomF'

Author(s)

Wayne Woodward

References


Examples

\[
\text{fac1} = c(1.6, -.9)
\]

\[
\text{fac2} = .8
\]

\[
\text{mult.wge(fac1, fac2)}
\]
NAICS

Monthly Retail Sales Data

Description

Monthly sales for the North American Industry Classification System (NAICS) code 44X72: Retail Trade and Food Services: 1992-2019

Usage

data("NAICS")

Format

ts object consisting of monthly data from January 1992- December 2019

Source

https://www.weather.gov/fwd/dmotemp

References

"Kaggle" and "US Census Bureau" websites

Examples

data(NAICS)

nbumps256

256 noisy bumps signal

Description

Noisy bumps signal shown in Figure 12.11(a) in Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott

Usage

data("nbumps256")

Format

The format is: num [1:256] -0.234 0.123 0.303 0.134 -0.513 ...

Source

Donoho and Johnstone(1994) Biometrika 81,425-455
References


Examples

data(nbumps256)

nile.min

Annual minimal water levels of Nile river

Description

Water levels for 622 through 1284 measured at Roda gauge near Cairo (Tousson, 1925)

Usage

data("nile.min")

Format

The format is: Time-Series [1:663] from 622 to 1284: 1157 1088 1169 1169 984 ...

Source


References


Examples

data(nile.min)
noctula: *Nyctalus noctula echolocation data*

**Description**

Echolocation signal for the Nyctalus noctula hunting bat

**Usage**

```r
data("noctula")
```

**Format**

The format is: num [1:96] -18 16 -5 -17 21 -6 -17 20 -6 -16 ...

**Source**

Internet

**References**


NSA: *Monthly Total Vehicle Sales*

**Description**

Monthly Total Vehicle Sales (TOTALNSA) in the United States from January 1976 - December 2019

**Usage**

```r
data("NSA")
```

**Format**

`ts` object consisting of monthly data from January 1976 - December 2019

**Source**

https://www.weather.gov/fwd/dmotemp

**References**

"Kaggle" and "US Census Bureau" websites
Examples

data(NSA)

data("ozona")

ts object consisting of number of chicken fried steaks sold daily during June and July, 2019

Source

"Time Series for Data Science: Analysis and Forecasting" by Woodward, Sadler, and Robertson

References

"Time Series for Data Science: Analysis and Forecasting" by Woodward, Sadler, and Robertson

Examples

data(ozona)

data(ozona)

pacfts.wge

Compute partial autocorrelations

Description

Compute partial autocorrelations using either YW (default and the classical method), Burg, or ML estimates.

Usage

pacfts.wge(x,lag.max=5, plot=TRUE,na.action,limits=FALSE,method = 'yw')
Arguments

x  Realization
lag.max  Max lag
plot  Logical variable
na.action  Not used
limits  Logical variable
method  Either "mle" (default), "burg", or "yw"

Value

method  Estimation method used: MLE, Burg, or YW
pacf  PACF estimates using estimation method specified

Author(s)

Wayne Woodward

References

"Time Series for Data Science: Analysis and Forecasting with R" by Woodward, Sadler, and Gray

Examples

data(sunspot2.0)
pacfts.wge(sunspot2.0, lag.max=10, method="burg")

Description

This function calculates and optionally plots the smoothed periodogram using the Parzen window. The truncation point may be chosen by the user

Usage

parzen.wge(x, dbcalc = "TRUE", trunc = 0, plot = "TRUE")

Arguments

x  Vector containing the time series realization
dbcalc  If dbcalc=TRUE, the calculation is in the log (dB) scale. If FALSE, then non-log calculations are made
trunc  if M=0 (default) then the function uses the truncation point 2*sqrt(n). If M>0, then the function uses the given value of M as the truncation point
plot  If PLOT=TRUE then the smoothed spectral estimate is plotted. If FALSE then no plot is created
Value

freq  The frequencies at which the smoothed periodogram is calculated
pzgram  The smoothed periodogram using the Parzen window

Author(s)

Wayne Woodward

References


Examples

parzen.wge(rnorm(100))

data(patemp)

Description

Pennsylvania average monthly temperatures

Usage

data("patemp")

Format

The format is: num [1:180] 38.1 38.3 44.5 52.3 59.2 70.6 73.9 71.3 63.9 57.3 ...

Source

Internet

References


Examples

data(patemp)
Description

Given a realization contained in a vector, this function calculates and optionally plots the periodogram in either log or non-log scale.

Usage

`period.wge(x, dbcalc = "TRUE", plot = "TRUE")`

Arguments

- `x` The vector containing the time series realization
- `dbcalc` if `dbcalc=TRUE` (default) then the periodogram is calculated in log scale (in dB). If `dbcalc` is `FALSE` then the non-log periodogram is calculated
- `plot` if `plot=TRUE` (default) the periodogram is plotted. If `plot=FALSE` no plot is created

Value

- `freq` Frequencies at which the periodogram is calculated
- `pgram` Periodogram values evaluated at the frequencies in `freq`

Author(s)

Wayne Woodward

References


Examples

`period.wge(rnorm(100))`
Given the coefficients of the AR and MA parts of an ARMA model, this function calculates the pi weights.

Usage

pi.weights.wge(phi = 0, theta = 0, lag.max = 5)

Arguments

phi Vector of AR coefficients (as in "Applied Time Series Analysis with R, 2nd edition" by Woodward, Gray, and Elliott (uses Box and Jenkins notation))
theta Vector of MA coefficients (as in ATSA and Box Jenkins texts)
lag.max The function will calculates psi weights $\psi(1), \psi(2), ..., \psi(lag.max)$. Note that $\psi(0)=1$.

Value

A vector containing $\psi(1), ..., \psi(lag.max)$

Author(s)

Wayne Woodward

References


Examples

pi.weights.wge(phi=c(1.2,-.6), theta=.5, lag.max=5)
Description

Plots DWT obtained using function `dwt` from `waveslim`.

Usage

`plotts.dwt.wge(x, n.levels, type='S8')`

Arguments

- `x` Realization (must be of length $2^k$ for some integer $k$ between 2 and 14
- `n.levels` Maximum order of discrete wavelet transforms to be calculated. `n.levels` must be less than or equal to $k$ where $n=2^k$
- `type` Discrete wavelet to use: options include 'haar', 'S8', 'D4', 'D6', 'D8'

Details

The `wavelsim dwt` function names these: 'haar', 'la8', 'd4', 'd6', and 'd8' respectively and the conversion is done transparently within the R code.

Value

The output is a plot of the DWT.

Note

Requires CRAN package 'waveslim'

Author(s)

Wayne Woodward

References


Examples

```r
data(bumps256)
plotts.dwt.wge(bumps256, n.levels=4, type='S8')
```
Plots MRA plot

Description

Plots MAR : plot associated with a multiresolution analysis using function mra from waveslim

Usage

`plotts.mra.wge(x, n.levels, type='S8')`

Arguments

- **x**: Realization (must be of length $2^k$ for some integer $k$ between 2 and 14
- **n.levels**: Maximum order of discrete wavelet transforms to be calculated. `n.levels` must be less than or equal to $k$ where $n=2^k$
- **type**: Discrete wavelet to use: options include 'haar', 'S8','D4','D6','D8'

Details

The wavelsim mra function names these :'haar', 'la8','d4','d6',and 'd8' respectively and the conversion is done transparently within the R code. This is done transparently within the R code.

Value

The output is a plot of the MRA.

Note

Requires CRAN package 'waveslim'

Author(s)

Wayne Woodward

References


Examples

```r
data(bumps256)
plotts.mra.wge(bumps256,n.levels=4,type='S8')
```
Calculate and plot the periodogram and Parzen window estimates with differing truncation points

Description

Given a time series contained in the vector \( x \), \texttt{plot.sp.parzen.wge} calculates and plots the periodogram and Parzen window estimates at the default truncation point \( M = 2 \times \sqrt{n} \) and up to 2 additional user specified truncation points.

Usage

\texttt{plot.sp.parzen.wge}(x, m2=c(0,0))

Arguments

- \texttt{x} - The vector containing the time series realization
- \texttt{m2} - A 2-component vector specifying up to 2 additional truncation points

Details

\( m2 = c(10,24) \) indicates that in addition to the default truncation point, the smoothed spectral estimator is to be calculated using truncation points 10 and 24, \( m2 = c(0,0) \) indicates that no additional truncation points are to be used, and \( m2 = c(10,0) \) indicates the use of one additional truncation point (10)

Value

- \texttt{freq} - Frequencies at which the periodogram and parzen widow estimates are calculated
- \texttt{db} - Periodogram (in dB) calculated at the frequencies in \texttt{freq}
- \texttt{dbz} - Parzen window estimate (in dB) calculated at the frequencies in \texttt{freq} using truncation point \( 2 \times \sqrt{n} \)
- \texttt{dbz1} - Parzen window estimate (in dB) calculated at the frequencies in \texttt{freq} using truncation point \( m2[1] \)
- \texttt{dbz2} - Parzen window estimate (in dB) calculated at the frequencies in \texttt{freq} using truncation point \( m2[2] \)

Author(s)

Wayne Woodward

References

Examples

data(ss08)
m2=c(10,50)
plotts.parzen.wge(ss08,m2)

---

plotts.sample.wge  Plot Data, Sample Autocorrelations, Periodogram, and Parzen Spectral Estimate

Description

For a given realization, this function plots the data, and calculates and plots the sample autocorrelations, periodogram, and Parzen window spectral estimator in a 2x2 array of plots.

Usage

plotts.sample.wge(x, lag.max = 25, trunc = 0, arlimits=FALSE, speclimits=c(0,0), periodogram=FALSE)

Arguments

x  A vector containing the realization
lag.max  The maximum lag at which to calculate the sample autocorrelations
trunc  The truncation point M for the Parzen spectral estimator. If M=0 then M=2sqrt(n). If M>0 then M is the value entered
arlimits  Logical variable. TRUE plots 95 percent limit lines on sample autocorrelation plots
periodogram  Logical variable. TRUE plots periodogram, default=FALSE
speclimits  User supplied limits for Parzen spectral density and periodogram, default=function decides limits

Value

xbar  The sample mean of the realization
autplt  A vector containing sample autocorrelations from 0, 1, ..., aut.lag
freq  A vector containing the frequencies at which the periodogram and window estimate are calculated
db  Periodogram (in dB) calculated at the frequencies in freq
freq  Parzen spectral estimate (in dB) calculated at the frequencies in freq

Author(s)

Wayne Woodward
References


Examples

```r
data(wages)
plotts.sample.wge(wages, trunc=0)
```

---

`plotts.true.wge` *Plot of generated data, true autocorrelations and true spectral density for ARMA model*

Description

For a given ARMA model, this function plots a realization, the true autocorrelations, and the true spectral density. This plot is typical of many plots in Applied Time Series Analysis by Woodward, Gray, and Elliott. For example, see Figure 1.21 and Figure 3.23.

Usage

```r
plotts.true.wge(n=100, phi=0, theta=0, lag.max=25, mu=0, vara = 1, sn=0, plot.data=TRUE)
```

Arguments

- `n` Length of time series realization to be generated. Default is 100
- `phi` Vector containing AR parameters
- `theta` Vector containing MA parameters
- `lag.max` Maximum lag for calculating and plotting autocorrelations
- `mu` True mean
- `vara` White noise variance: default=1
- `sn` determines the seed used in the simulation of plotted realization. sn=0 produces new/random realization each time. sn=positive integer produces same realization each time
- `plot.data` Logical variable: If TRUE a simulated realization is plotted

Value

- `data` Realization of length `n` that is generated from the ARMA model
- `aut1` True autocorrelations from the ARMA model for lags 0 to `lag.max`
- `acv` True autocovariances from the ARMA model for lags 0 to `lag.max`
- `spec` Spectral density (in dB) for the ARMA model calculated at frequencies f=0, .002, .004, ...., .5
Note

gvar=g[1], i.e. autocovariance at lag 0

Author(s)

Wayne Woodward

References


Examples

plotts.true.wge(n=100, phi=c(1.6,-.9), theta=.8, lag.max=25, var=1)

plotts.wge

Plot a time series realization

Description

Given a realization contained in a vector, this function plots it as a time series realization

Usage

plotts.wge(x, style = 0, xlab = "Time", ylab = "", main="", col="black", text_size=12, lwd=0.75, cex=0.5, cex.lab=0.75, cex.axis=0.75, xlim=NULL, ylim=NULL)

Arguments

x The vector containing the time series realization to be plotted
style If style is 0 then a simple plot of the realization is rendered. If style is 1 then a ggplot is rendered.
xlab A string that represents the x-axis label.
ylab A string that represents the y-axis label.
main A string that represents the main title.
col Color of plot.
text_size Text size.
lwd Line width.
cex See R documentation.
cex.lab See R documentation.
cex.axis See R documentation.
xlim String giving x-axis plot limits.
ylim String giving y-axis plot limits.
**Value**

Simply a plot of the realization

**Author(s)**

Wayne Woodward

**References**


**Examples**

```r
data(sunspot2.0); plotts.wge(sunspot2.0)
```

---

**prob10.4**  
*Data matrix for Problem 10.4 in "Applied Time Series Analysis with R, 2nd edition" by Woodward, Gray, and Elliott*

---

**Description**

Matrix containing a bivariate VAR data set

**Usage**

```r
data("prob10.4")
```

**Format**

The format is: num [1:100, 1:2] 0 0.7184 -0.3448 -2.1638 -0.0342 ... - attr(*, "dimnames")=List of 2  ..$ : NULL  ..$ : chr [1:2] "X1" "X2"

**Source**

Simulated data

**References**


**Examples**

```r
data(prob10.4)
```
prob10.6x

Data for Problem 10.6 in Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott

Description
This realization is the unobservable data associated with the observed data in prob10.6y

Usage
data("prob10.6x")

Format
The format is: num [1:9] 2.61 0.69 0.64 0.37 -0.79 -1.63 -1.14 -1.2 -3.13

Source
Simulated data

References

Examples
data(prob10.6x)

prob10.6y

Simulated observed data for Problem 10.6 in Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott

Description
Kalman filter example data

Usage
data("prob10.6y")

Format
The format is: num [1:9] 3.28 -0.05 0.64 0.31 -0.9 -2.4 -1.83 -1.93 -3.52

Source
Simulated data
References


Examples

data(prob10.6y)

Data for Problem 10.7 in Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott

Description

This realization is the same unobservable data as in prob10.6x

Usage

data("prob10.7x")

Format

The format is: num [1:9] 2.61 0.69 0.64 0.37 -0.79 -1.63 -1.14 -1.2 -3.13

Source

Simulated data

References


Examples

data(prob10.7x)
**prob10.7y**

*Simulated observed data for Problem 10.6 in Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott*

**Description**

Kalman filter example data

**Usage**

```r
data("prob10.7y")
```

**Format**

The format is: num [1:9] 3.28 -0.05 0.64 0.31 -0.9 -2.4 -1.83 -1.93 -3.52

**Source**

Simulated data

**References**


**Examples**

```r
data(prob10.7y)
```

---

**prob11.5**

*Data for Problem 11.5 in Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott*

**Description**

Simulated fractional long memory data

**Usage**

```r
data("prob11.5")
```

**Format**

The format is: num [1:10] 4.2 -2.5 8.4 14.6 7 9.6 19.8 4.8 6.5 8.3

**Source**

Simulated data
prob12.1c

References


Examples

data(prob11.5)

data(prob12.1c)

Description

Data from a problem set in the wavelet chapter

Usage

data("prob12.1c")

Format

The format is: num [1:200] 9.49 8.01 3.43 -1.85 -4.99 -7.21 -5.61 -2.34 2.16 3.88 ...

Source

Simulated data

References


Examples

data(prob12.1c)
**prob12.3a**

*Data for Problem 12.3a in Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott*

**Description**

Data from a problem set in the wavelet chapter

**Usage**

```r
data("prob12.3a")
```

**Format**

The format is: `num [1:512] -3.09 8.43 -9.74 8.44 -3.46 ...`

**Source**

Simulated data

**References**


**Examples**

```r
data(prob12.3a)
```
References


Examples

```r
data(prob12.3b)
```

---

**prob12.6c**

Data set for Problem 12.6(C) in Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott

Description

Simulated TVF data set

Usage

```r
data("prob12.6c")
```

Format

The format is: num [1:512] -0.482 -0.569 -0.656 -0.743 -0.83 ...

Source

Simulated data

References


Examples

```r
data(prob12.6c)
```
### prob13.2

*Data for Problem 13.2 in Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott*

**Description**

Simulated data from cosine-plus-noise model

**Usage**

```r
data("prob13.2")
```

**Format**

The format is: `num [1:256] 1.524 5.886 5.939 4.319 0.573 ...`

**Source**

Simulated data

**References**


**Examples**

```r
data(prob13.2)
```

---

### prob8.1a

*Data for Problem 8.1 in "Applied Time Series Analysis with R, 2nd edition" by Woodward, Gray, and Elliott*

**Description**

See title above

**Usage**

```r
data("prob8.1a")
```

**Format**

The format is: `num [1:200] 2.19 0.48 0.06 3.86 3.6 -3.38 6.23 1.95 1.4 -5.35 ...`

**Source**

Simulated data
References


Examples

data(prob8.1a)

---

prob8.1b  
*Data for Problem 8.1 in "Applied Time Series Analysis with R, 2nd edition" by Woodward, Gray, and Elliott*

Description

See title above

Usage

data("prob8.1b")

Format

The format is: num [1:200] 1.54 -0.13 1.93 0.29 -0.13 -0.23 1.27 1.01 -0.65 1.68 ...  

Source

Simulated data

References


Examples

data(prob8.1b)

**Description**

See title above

**Usage**

data("prob8.1c")

**Format**

The format is: num [1:200] 0.33 -0.53 -2.36 2.48 -0.36 -2.02 1.87 -0.73 0.41 2.41 ...

**Source**

Simulated data

**References**


**Examples**

data(prob8.1c)

---


**Description**

See title above

**Usage**

data("prob8.1d")

**Format**

The format is: num [1:200] -0.07 -1.74 -1.37 -0.52 0.14 0.07 -1.5 1.88 -0.03 -1.81 ...

**Source**

Simulated data
References


Examples

data(prob8.1d)

prob9.6c1

Data set 1 for Problem 6.1c

Description

Data set 1 for Problem 6.1c in "Applied Time Series Analysis with R, 2nd edition" by Woodward, Gray, and Elliott. It is either from line plus noise or random walk with drift.

Usage

data("prob9.6c1")

Format

The format is: num [1:100] -0.2924 0.0206 0.6595 0.3819 0.0269 ...

Source

Simulated data

References


Examples

data(prob9.6c1)
**prob9.6c2**  
*Data set 2 for Problem 6.1c*

**Description**
Data set 2 for Problem 6.1c in "Applied Time Series Analysis with R, 2nd edition" by Woodward, Gray, and Elliott. It is either from line plus noise or random walk with drift.

**Usage**
data("prob9.6c2")

**Format**
The format is: num [1:100] -0.925 -2.679 -2.378 -3.03 -2.157 ...  

**Source**
Simulated data

**References**

**Examples**
data(prob9.6c2)

---

**prob9.6c3**  
*Data set 3 for Problem 6.1c*

**Description**
Data set 3 for Problem 6.1c in "Applied Time Series Analysis with R, 2nd edition" by Woodward, Gray, and Elliott. It is either from line plus noise or random walk with drift.

**Usage**
data("prob9.6c3")

**Format**
The format is: num [1:100] -2.79 -3.32 -3.51 -5.13 -3.51 ...  

**Source**
Simulated data
References


Examples

data(prob9.6c3)

data(prob9.6c4)

Description

Data set 4 for Problem 6.1c in "Applied Time Series and Data Analysis with R, 2nd edition" by Woodward, Gray, and Elliott. It is either from line plus noise or random walk with drift.

Usage

data("prob9.6c4")

Format

The format is: num [1:100] -0.0599 -0.0214 0.6589 -0.151 0.4043 ...

Source

Simulated data

References

"Applied Time Series and Data Analysis with R, 2nd edition" by Woodward, Gray, and Elliott

Examples

data(prob9.6c4)
Calculate psi weights for an ARMA model

Description

Given the coefficients of the AR and MA parts of an ARMA model, this function calculates the psi weights.

Usage

psi.weights.wge(phi = 0, theta = 0, lag.max = 5)

Arguments

phi Vector of AR coefficients (as in "Applied Time Series Analysis with R, 2nd edition" by Woodward, Gray, and Elliott (uses Box and Jenkins notation))

theta Vector of MA coefficients (as in ATSA and Box Jenkins texts)

lag.max The function will calculates psi weights psi(1), psi(2), ..., psi(lag.max). Note that psi(0)=1.

Value

A vector containing psi(1), ..., psi(lag.max)

Author(s)

Wayne Woodward

References


Examples

psi.weights.wge(phi=c(1.2,-.6), theta=.5, lag.max=5)
**rate**  
*Daily DOW rate of Return*

**Description**

Daily DOW rate of return from 1971 through 2020

**Usage**

```r
data("rate")
```

**Format**

`ts` object consisting of daily Dow rate of return from 1971 through 2020

**References**

"Time Series for Data Science: Analysis and Forecasting" by Woodward, Sadler, and Robertson

**Examples**

```r
data(rate)
```

---

**roll.win.rmse.nn.wge**  
*Function to Calculate the Rolling Window RMSE*

**Description**

This function creates as many "windows" as is possible with the data and calculates an RMSE for each window. The resulting "rolling window RMSE" is the average of the individual RMSEs from each window.

**Usage**

```r
roll.win.rmse.nn.wge(series, horizon = 1, fit_model)
```

**Arguments**

- **series**: The data
- **horizon**: The number of observations ahead to be forecasted.
- **fit_model**: The `mlp` object (model) to be evaluated. This model will have been fit before the call to this function.
roll.win.rmse.wge

Value

rwRMSE  The average of the individual RMSEs of each window
numwindows  The number of windows
horizon  The number of observations ahead to be forecasted.

Author(s)

Bivin Sadler

References

"The Time Series Tool Kit"

roll.win.rmse.wge  Function to Calculate the Rolling Window RMSE

Description

This function creates as many "windows" as is possible with the data and calculates an RMSE for each window. The resulting "rolling window RMSE" is the average of the individual RMSEs from each window.

Usage

roll.win.rmse.wge(series, horizon = 2, s = 0, d = 0, phi = 0, theta = 0)

Arguments

series  The data
horizon  The number of observations ahead to be forecasted.
s  Order of the seasonal difference, default=1
d  Order of the difference
phi  Vector of AR coefficients
theta  Vector of MA coefficients

Value

rwRMSE  The average of the individual RMSEs of each window
numwindows  The number of windows
horizon  The number of observations ahead to be forecasted.
s  Order of the seasonal difference, default=1
d  Order of the difference
phis  Vector of AR coefficients
thetas  Vector of MA coefficients
RMSEs  Vector of RMSEs ... one for each window
slr.wge

**Simple Linear Regression**

**Description**

Uses Base R routine `lm` to simplify call for SLR where independent variable is automatically $t=1:n$

**Usage**

```r
slr.wge(x)
```

**Arguments**

- `x` The TVF data set

**Value**

- `res` Residuals
- `b0hat` Estimate $b0$ in model $y=b0+b1*t+Z$
- `b1hat` Estimate $b1$
- `pvalue` pvalue for test: $slope=0$
- `tstatistic` tstatistic associated with test: $slope=0$

**Author(s)**

Wayne Woodward

**References**


**Examples**

```r
x = gen.arma.wge(n=100, phi=.96, sn=10)
y = slr.wge(x)
```
**ss08**  
*Sunspot Data*

**Description**  
Annual average sunspot numbers for the years 1749-2008

**Usage**  
```r
data("ss08")
```

**Format**  
The format is: num [1:260] 80.9 83.4 47.7 47.8 30.7 ...

**Source**  
Internet-open source

**References**  

**Examples**  
```r
data(ss08)
```

---

**ss08.1850**  
*Sunspot data from 1850 through 2008 for matching with global temperature data (hadley)*

**Description**  
Sunspot data from 1850 through 2008 for matching with global temperature data (hadley) for purposes of testing for association in Example 10.5 of "Applied Time Series Analysis with R, 2nd edition" by Woodward, Gray, and Elliott

**Usage**  
```r
data("ss08.1850")
```

**Format**  
The format is: num [1:160] 66.6 64.5 54.1 39 20.6 ...
Source

Internet

References


Examples

data(ss08.1850)

---

starwort.ex  Starwort Explosion data shown in Figure 13.13a in Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott

Description

Lg wave for Starwort explosion data

Usage

data("starwort.ex")

Format

The format is: num [1:420] 43245 48408 47565 7372 -62277 ...

Source


References


Examples

data(starwort.ex)
sunspot.classic

Classic Sunspot Data: 1749-1924

Description
The classic 176 point sunspot data from 1749-1924 that has been widely modeled

Usage
data("sunspot.classic")

Format
The format is: num [1:176] 80.9 83.4 47.7 47.8 30.7 12.2 9.6 10.2 32.4 47.6 ...

Source
Internet

References

Examples
data(sunspot.classic)

sunspot2.0

Annual Sunspot2.0 Numbers

Description
Annual sunspot2.0 numbers from 1700 through 2020

Usage
data("sunspot2.0")

Format
ts object consisting of annual data from 1700 through 2020

Source
https://www.sidc.oma.be/silso
References

"Time Series for Data Science: Analysis and Forecasting" by Woodward, Sadler, and Robertson

Examples

data(sunspot2.0)

Description

Monthly sunspot2.0 numbers from January 1749 through December 2020

Usage

data("sunspot2.0.month")

Format

ts object consisting of monthly data from January 1749 through December 2020

Source

https://www.sidc.oma.be/silso

References

"Time Series for Data Science: Analysis and Forecasting" by Woodward, Sadler, and Robertson

Examples

data(sunspot2.0.month)
**table10.1.noise**  
*Noise related to data set, the first 5 points of which are shown in Table 10.1 in Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott*

---

**Description**

The data in Table 10.1 are of the form \( Y(t) = X(t) + n(t) \). This data set contains the values for \( n(t) \).

**Usage**

```r
data("table10.1.noise")
```

**Format**

The format is: `num [1:75] -0.49 0.126 -0.129 -1.179 0.441 ...

**Source**

Simulated data

**References**


**Examples**

```r
data(table10.1.noise)
```

---

**table10.1.signal**  
*Underlying, unobservable signal (X(t), the first 5 points of which are shown in Table 10.1 in Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott)*

---

**Description**

The \( X(t) \) data is unobservable, and is a realization from an AR(1) model.

**Usage**

```r
data("table10.1.signal")
```

**Format**

The format is: `num [1:75] -0.2497 -0.0812 -0.6463 -1.7653 -2.719 ...

---
Source
Simulated data

References

Examples
data(table10.1.signal)

table7.1  \hspace{1cm} MA(2) data for Table 7.1

Description
MA(2) data for Table 7.1 in "Applied Time Series Analysis with R, 2nd edition" by Woodward, Gray, and Elliott. Uses function ia in package itsmr to show steps in the innovations algorithm for estimating the MA parameters and white noise variance

Usage
data("table7.1")

Format
The format is: num [1:400] 0.4481 0.5497 -1.6586 -3.1653 -0.0314 ...

Source
Generated data

References

Examples
data(table7.1)
**tesla**

*Tesla Stock Prices*

**Description**
Teslas daily stock prices from January 1, 2020 through April 30, 2021

**Usage**
data("tesla")

**Format**
ts object consisting of daily adjusted close price for TSLA from January 1, 2020 through April 30, 2021

**Source**
https://finance.yahoo.com

**References**
"Time Series for Data Science: Analysis and Forecasting" by Woodward, Sadler, and Robertson

**Examples**
data(tesla)

---

**trans.to.dual.wge**

*Transforms TVF data set to a dual data set*

**Description**
Using the specified values for lambda and offset, this function transforms a TVF data set to a dual data set based on a Glambda time transformation.

**Usage**
trans.to.dual.wge(x, lambda, offset = 60, h = 0, plot = TRUE)

**Arguments**
- **x** The TVF data set
- **lambda** The value of lambda in the Glambda time transformation
- **offset** The value of offset in the Glambda time transformation
- **h** Scaling variable, initialized at zero, which assures that the dual data set has the same number of points as the original TVF data set
- **plot** Logical: TRUE=plot, FALSE=no plot
Value

\textbf{intX} See intY description
\textbf{intY} The input realization x is of length n, and the values of x are available at the time points t= 1 to n. The values intY are n interpolated values of the original time series at the values of intX in the original time scale. The dual data set is obtained by associating the n values of intY with t = 1 to n respectively
\textbf{h} The output value of the scaling parameter that assures that the dual realization and the original realization are of the same length

Author(s)
Wayne Woodward

References

Examples
\begin{verbatim}
data(fig13.2c) y=trans.to.dual.wge(x=fig13.2c,lambda=-.4,offset=63)
\end{verbatim}

Description
Using the specified values for lambda and offset, this function transforms a dual data set, based on a Glambda time transformation, back to the original time scale

Usage
\begin{verbatim}
trans.to.original.wge(xd, lambda, offset, h, plot = TRUE)
\end{verbatim}

Arguments
\begin{itemize}
\item \textbf{xd} The dual data set
\item \textbf{lambda} The value of lambda in the Glambda time transformation
\item \textbf{offset} The value of offset in the Glambda time transformation
\item \textbf{h} Scaling variable obtained as output from transform.to.dual.wge that assures that the dual data set has the same number of points as the original TVF data set
\item \textbf{plot} Logical: TRUE=plot, FALSE=no plot
\end{itemize}

Value
Returns the y values to be plotted at time points t=1 to n that approximate the original TVF data set
true.arma.aut.wge

**Author(s)**
Wayne Woodward

**References**

**Examples**
```r
data(fig13.2c)
yd=trans.to.dual.wge(fig13.2c,lambda=-.4,offset=63)
yo=trans.to.original.wge(yd$intY,lambda=-.4,offset=63,h=yd$h)
```

**true.arma.aut.wge**  
*True ARMA autocorrelations*

**Description**
R function to calculate the autocovariances and autocorrelations and optionally plot the true autocorrelations of a stationary ARMA model

**Usage**
```r
true.arma.aut.wge(phi = 0, theta = 0, lag.max = 25, vara = 1, plot=TRUE)
```

**Arguments**
- **phi**: Vector containing AR coefficients
- **theta**: Vector containing MA coefficients
- **lag.max**: Maximum lag at which to calculate the true autocorrelations
- **vara**: White noise variance of the ARMA model
- **plot**: Logical: TRUE=plot, FALSE=no plot

**Value**
- **acf**: Vector of length max.lag+1 containing true autocorrelations at lags 0, 1, ..., lag.max
- **acv**: Vector of length max.lag+1 containing true autocovariances at lags 0, 1, ..., lag.max

**Author(s)**
Wayne Woodward

**References**
true.arma.spec.wge

Example

true.arma.aut.wge(\phi=c(1.6, -0.9), \theta=-0.8, \text{lag.max}=15, \text{vara}=1)

---

true.arma.spec.wge True ARMA Spectral Density

Description

R function to calculate and optionally plot the spectral density of a stationary ARMA model

Usage

true.arma.spec.wge(\phi=0, \theta=0, \text{vara}=1, \text{plot}=\text{TRUE})

Arguments

- \phi: Vector containing AR coefficients
- \theta: Vector containing MA coefficients
- vara: White noise variance of the ARMA model
- plot: Logical: TRUE=plot, FALSE=no plot

Value

- \textbf{f}: Frequencies at which true spectral density is evaluated: 0, 1/500, 2/500, ..., .5
- \text{spec}: True spectral density calculated at the frequencies in \text{f}

Author(s)

Wayne Woodward

References


Examples

true.arma.spec.wge(\phi=c(1.6, -0.9), \theta=-0.7)
true.farma.aut.wge  True FARMA autocorrelations

Description
Calculate the autocovariances and autocorrelations and optionally plot the true autocorrelations of a FARMA model.

Usage
true.farma.aut.wge(d, phi=0, theta=0, lag.max=50, trunc=1000, vara=1, plot=TRUE)

Arguments
d Fractional difference parameter
phi vector of AR parameters of ARMA part of FARMA model
theta vector of MA parameters of ARMA part of FARMA model using signs as given in the Woodward, Gray, and Elliott text
lag.max Maximum lag at which the autocorrelations and autocovariances will be calculated
trunc Number of terms used in sum
vara White noise variance
plot Logical: TRUE=plot, FALSE=no plot

Details
For fractional model use phi=theta=0

Value
acf Vector of length max.lag+1 containing true autocorrelations at lags 0, 1, ..., lag.max
acv Vector of length max.lag+1 containing true autocovariances at lags 0, 1, ..., lag.max

Author(s)
Wayne Woodward

References

Examples
y=true.farma.aut.wge(d=.4, phi=c(0,-.8))
true.garma.aut.wge  

True GARMA autocorrelations

Description

Calculate the autocovariances and autocorrelations and optionally plot the true autocorrelations of a 1-factor based on formula (11.25) of "Applied Time Series Analysis with R, second edition" Woodward, Gray, and Elliott

Usage

true.garma.aut.wge(u,lambda,phi=0,theta=0,lag.max=50,vara=1,plot=TRUE)

Arguments

- **u**: Parameter u in the GARMA model given in (11.16) of Woodward, Gray, and Elliott text
- **lambda**: Parameter lambda in the GARMA model given in (11.16) of Woodward, Gray, and Elliott text
- **phi**: vector of AR parameters of ARMA part of GARMA model
- **theta**: vector of MA parameters of ARMA part of GARMA model using signs as given in the Woodward, Gray, and Elliott text
- **lag.max**: Maximum lag at which the autocorrelations and autocovariances will be calculated
- **vara**: White noise variance
- **plot**: Logical: TRUE=plot, FALSE=no plot

Details

For Gegenbauer model use phi=theta=0

Value

- **acf**: Vector of length max.lag+1 containing true autocorrelations at lags 0, 1, ..., lag.max
- **acv**: Vector of length max.lag+1 containing true autocovariances at lags 0, 1, ..., lag.max

Author(s)

Wayne Woodward

References

Examples

\[ y = \text{true.garma.aut.wge}(u=.8,\lambda=.4,\phi=.8) \]

**tx.unemp.adj**  
*Texas Seasonally Adjusted Unemployment Rates*

**Description**

Monthly seasonally adjusted unemployment rate in Texas for the years 2000-2019

**Usage**

```r
data("tx.unemp.adj")
```

**Format**

`ts` object consisting of monthly seasonally adjusted unemployment rate from January 2000 through December 2019

**Source**

https://twc.texas.gov

**References**

"Time Series for Data Science: Analysis and Forecasting" by Woodward, Sadler, and Robertson

**Examples**

```r
data(tx.unemp.adj)
```

**tx.unemp.unadj**  
*Texas Unadjusted Unemployment Rates*

**Description**

Monthly unemployment rate in Texas for the years 2000-2019

**Usage**

```r
data("tx.unemp.unadj")
```

**Format**

`ts` object consisting of monthly unadjusted unemployment rate from January 2000 through December 2019
Source

https://twc.texas.gov

References

“Time Series for Data Science: Analysis and Forecasting” by Woodward, Sadler, and Robertson

Examples

data(tx.unemp.unadj)

```
unit.circle.wge = function(real = .9, imaginary = .95)
```

Description

This function plots the roots of the characteristic equation on the complex plain and super imposes the Unit Circle to show if a root is inside, outside or on the Unit Circle. The modulus and absolute reciprocal are also displayed.

Usage

```
unit.circle.wge(real = 0, imaginary = 0)
```

Arguments

- `real`: the real part of the root
- `imaginary`: the imaginary part of the root

Value

returns a plot of the root with respect to the unit circle

Author(s)

Bivin Sadler

References


Examples

```
unit.circle.wge = function(real = .9, imaginary = .95)
```
### us.retail

**Quarterly US Retail Sales**

**Description**
Quarterly US retail sales (in $millions) from the fourth quarter of 1999 through the second quarter of 2021

**Usage**
```r
data("us.retail")
```

**Format**
ts object consisting of quarterly US retail sales (in $millions) from the fourth quarter of 1999 through the second quarter of 2021

**Source**
https://www.fred.stlouis.org

**References**
"Time Series for Data Science: Analysis and Forecasting" by Woodward, Sadler, and Robertson

**Examples**
```r
data(us.retail)
```

### uspop

**US population**

**Description**
US estimated annual population from 1900 through 2020.

**Usage**
```r
data("uspop")
```

**Format**
ts object consisting of annual data from 1700 through 2020

**Source**
Internet
References

"Time Series for Data Science: Analysis and Forecasting" by Woodward, Sadler, and Robertson

wages

Daily wages in Pounds from 1260 to 1944 for England

Description
This data set contains the average English daily wages in pounds for each year from 1260 to 1944, inclusive.

Usage
data("wages")

Format
The format is: num [1:735] 4.41 4.63 4.38 4.52 4.42 4.64 4.44 5.15 5.23 4.42...

Source
Data Market Time Series Data Library (citing: Makridakis, Wheelwright and Hyndman (1998))

Examples
data(wages)

wbg.boot.wge

Woodward-Bottone-Gray test for trend

Description
Performs the Woodward-Bottone-Gray (WBG) bootstrap-based test for a linear trend in a time series realization.

Usage
wbg.boot.wge(x,nb=399,alpha=.05,pvalue=TRUE,sn=0)

Arguments
x Realization
nb The number of Bootstrap replications (default is 399)
alpha The significance level of the test (default is .05)
pvalue Logical variable. TRUE(default) prints out the p-value of the test.
sn Sets the seed for the simulations (default = 0)
Value

- **p**: AR order used for the bootstrap simulations
- **phi**: The AR coefficients of the AR model fit to data
- **pv**: The p-value of the test

Author(s)

Wayne Woodward

References


Examples

```r
data(global.temp)
wbg.boot.wge(global.temp)
```

---

**whale**

*Whale click data*

Description

256 point whale click echolocation signal

Usage

```r
data("whale")
```

Format

The format is: num [1:286] 0.0014 -0.008 0.01126 0.00412 0.0069 ... 

Source

Stan Kuczaj from University of Southern Mississippi

References


Examples

```r
data(whale)
```
wtcrude

**West Texas Intermediate Crude Oil Prices**

**Description**

Monthly West Texas intermediate crude oil prices from January 2000 through October 2009.

**Usage**

```r
data("wtcrude")
```

**Format**

The format is: num [1:118] 27.2 29.4 29.9 25.7 28.8 ...

**Source**

Internet

**References**


wtcrude2020

**Monthly WTI Crude Oil Prices**

**Description**

Monthly WTI crude oil prices from January 1990 through December 2020

**Usage**

```r
data("wtcrude2020")
```

**Format**

ts object consisting of monthly data from January 1990 through December 2020

**Source**

https://fred.stlouis.org

**References**

"Time Series for Data Sience: Analysis and Forecasting" by Woodward, Sadler, and Robertson

**Examples**

```r
data(wtcrude2020)
```
**wv.wge**

*Function to calculate Wigner Ville spectrum*

**Description**
Calculates and plots Wigner-Ville spectrum for a realization

**Usage**
wv.wge(x)

**Arguments**

x Realization to be analyzed

**Value**
Plots Wigner-Ville spectrum

**Author(s)**
Wayne Woodward

**References**
Boashash (2003). Time Frequency Analysis

**Examples**

data(doppler)
wv.dop=wv.wge(doppler)

---

**yellowcab.precleaned**  *Precleaned Yellow Cab data*

**Description**
The number of Yellow Cab Trips in NYC before and during the COVID outbreak: January 2019 through February 2021

**Usage**
data("yellowcab.precleaned")

**Format**
The format is: Time-Series [1:26] from 2019 to 2021: 247315 250654 252634 247742 ...
Source

NYC Taxi and Limousine website

References

Time Series for Data Science Woodward, Sadler, and Robertson

Examples

data(yellowcab.precleaned)
Index

* AICC
  aic.ar.wge, 6
  aic.burg.wge, 7
  aic.wge, 8
  aic5.ar.wge, 9
  aic5.wge, 10

* AIC
  aic.ar.wge, 6
  aic.burg.wge, 7
  aic.wge, 8
  aic5.ar.wge, 9
  aic5.wge, 10

* AR Model Identification
  aic.ar.wge, 6
  aic5.ar.wge, 9

* ARCH
  gen.arch.wge, 74

* ARIMA
  fore.arima.wge, 64
  fore.aruma.wge, 67
  gen.arima.wge, 75

* ARMA
  fore.arma.wge, 66
  gen.arma.wge, 76
  plotts.true.wge, 112
  true.arma.aut.wge, 139
  true.arma.spec.wge, 140

* ARUMA
  fore.aruma.wge, 67
  gen.aruma.wge, 77

* Additive components
  factor.comp.wge, 36

* Autocorrelations
  plotts.true.wge, 112
  true.arma.aut.wge, 139

* Autoregressive
  aic.ar.wge, 6
  aic5.ar.wge, 9
  est.ar.wge, 30

  factor.comp.wge, 36

* BIC
  aic.ar.wge, 6
  aic.burg.wge, 7
  aic.wge, 8
  aic5.ar.wge, 9
  aic5.wge, 10

* Backcasting
  est.arma.wge, 31

* Bootstrap
  wbg.boot.wge, 146

* Burg
  aic.ar.wge, 6
  aic5.ar.wge, 9
  est.ar.wge, 30
  pacfts.wge, 103

* Butterworth
  butterworth.wge, 18

* Centered
  ma.smooth.wge, 95

* Cochrane-Orcutt
  co.wge, 21

* Conditional variance
  gen.arch.wge, 74
  gen.garch.wge, 78

* DWT
  plotts.dwt.wge, 108

* FARMA
  est.farma.wge, 32
  fore.farma.wge, 68
  true.farma.aut.wge, 141

* Factor table
  factor.wge, 37

* Factors
  mult.wge, 99

* Forecasts
  fore.arima.wge, 64
  fore.aruma.wge, 66
  fore.aruma.wge, 67

151
* Fractional  
  fore.farma.wge, 68

* G(\lambda) model  
  is.glambd.wge, 85  
  is.sample.wge, 86

* G(\lambda)  
  fore.glambd.wge, 70

* GARCH  
  gen.garch.wge, 78

* GARMA  
  est.garma.wge, 33  
  fore.garma.wge, 69  
  gen.garma.wge, 79

* GLP  
  macoef.geg.wge, 96

* Gegenbauer  
  est.garma.wge, 33  
  fore.garma.wge, 69  
  gegenb.wge, 73  
  gen.garma.wge, 79  
  gen.geg.wge, 80  
  macoef.geg.wge, 96  
  true.garma.aut.wge, 142

* Glambda  
  trans.to.dual.wge, 137  
  trans.to.original.wge, 138

* Hilbert  
  hilbert.wge, 85

* Integrated  
  gen.arima.wge, 75

* Kalman filter  
  kalman.miss.wge, 87  
  kalman.wge, 88

* Linear  
  slr.wge, 130

* Ljung-Box test  
  ljung.wge, 91

* Long memory  
  gen.geg.wge, 80

* MLE  
  aic5.ar.wge, 9  
  est.ar.wge, 30  
  pacfts.wge, 103

* ML  
  aic.ar.wge, 6

* MRA  
  plots.mra.wge, 109

* Maximum likelihood  
  est.arma.wge, 31

* Model Identification  
  aic.burg.wge, 7  
  aic.wge, 8  
  aic5.wge, 10

* Moving Average Predictor  
  ma.pred.wge, 94

* Moving Average Smoother  
  ma.smooth.wge, 95

* Nonstationary  
  gen.aruma.wge, 77

* One-sided Moving Average  
  ma.pred.wge, 94

* PACF  
  pacfts.wge, 103

* Parzen  
  ample.spec.wge, 12  
  parzen.wge, 104  
  plotts.sample.wge, 111

* Periodogram  
  plotts.sample.wge, 111

* Pi weights  
  psi.weights.wge, 127

* Plot  
  ample.spec.wge, 12  
  parzen.wge, 104  
  plotts.sample.wge, 111  
  plotts.wge, 113

* Psi weights  
  psi.weights.wge, 127

* RMSE  
  roll.win.rmse.nn.wge, 128  
  roll.win.rmse.wge, 129

* Realization  
  gen.arima.wge, 75  
  gen.arma.wge, 76  
  gen.aruma.wge, 77  
  gen.garma.wge, 79  
  gen.geg.wge, 80  
  gen.glambd.wge, 81  
  plotts.wge, 113

* Regression  
  co.wge, 21  
  slr.wge, 130  
  wbg.boot.wge, 146

* Rolling Moving Average  
  ma.pred.wge, 94

* Root
INDEX

unit.circle.wge, 144

* Seasonal
  fore.arima.wge, 64
  fore.aruma.wge, 67

* Spectral density
  plots.true.wge, 112

* Stationary
  unit.circle.wge, 144

* TVF data
  is.glambda.wge, 85
  is.sample.wge, 86

* TVF
  est.glambda.wge, 34
  fore.glambda.wge, 70
  gen.glambda.wge, 81
  trans.to.dual.wge, 137
  trans.to.original.wge, 138
  wv.wge, 149

* Trend Test
  co.wge, 21
  wbg.boot.wge, 146

* Unit Circle
  unit.circle.wge, 144

* Wigner-Ville spectrum
  wv.wge, 149

* Wigner-Ville
  hilbert.wge, 85

* Woodward-Bottone-Gray
  wbg.boot.wge, 146

* YW
  aic.ar.wge, 6
  aic5.ar.wge, 9

* Yule Walker
  est.ar.wge, 30
  pacfts.wge, 103

* autocorrelations
  true.farma.aut.wge, 141
  true.garma.aut.wge, 142

* backcasting
  backcast.wge, 14

* dB
  period.wge, 106

* datasets
  airline, 11
  airlog, 11
  appy, 13
  bat, 15
  bitcoin, 16
  Bsales, 16
  bumps16, 17
  bumps256, 17
  cardiac, 19
  cement, 19
  chirp, 20
  dfw.2011, 22
  dfw.mon, 22
  dfw.yr, 23
  doppler, 24
  doppler2, 24
  dow.annual, 25
  dow.rate, 25
  dow1000, 26
  dow1985, 27
  dowjones2014, 27
  eco.cd6, 28
  eco.corp.bond, 28
  eco.mort30, 29
  fig1.10a, 38
  fig1.10b, 38
  fig1.10c, 39
  fig1.10d, 40
  fig1.16a, 40
  fig1.21a, 41
  fig1.22a, 42
  fig1.5, 42
  fig10.11x, 43
  fig10.11y, 44
  fig10.1bond, 44
  fig10.1cd, 45
  fig10.1mort, 46
  fig10.3x1, 46
  fig10.3x2, 47
  fig11.12, 48
  fig11.4a, 48
  fig12.1a, 49
  fig12.1b, 50
  fig13.18a, 50
  fig13.2c, 51
  fig3.10d, 52
  fig3.16a, 52
  fig3.18a, 53
  fig3.24a, 54
  fig3.29a, 54
  fig4.8a, 55
  fig5.3c, 56
  fig6.11a, 56
fig6.1nf, 57
fig6.2nf, 58
fig6.5nf, 58
fig6.6nf, 59
fig6.7nf, 60
fig6.8nf, 60
fig8.11a, 61
fig8.4a, 62
fig8.6a, 62
fig8.8a, 63
flu, 64
freeze, 72
freight, 73
global.temp, 83
global2020, 83
hadley, 84
kingKong, 89
lavon, 90
lavon15, 90
linearchirp, 91
lynx, 92
lynx, 93
ma2.table7.1, 96
mass.mountain, 97
MedDays, 98
mm.eq, 98
NAICS, 100
nbumps256, 100
nile.min, 101
noctula, 102
NSA, 102
ozona, 103
patemp, 105
prob10.4, 114
prob10.6x, 115
prob10.6y, 115
prob10.7x, 116
prob10.7y, 117
prob11.5, 117
prob12.1c, 118
prob12.3a, 119
prob12.3b, 119
prob12.6c, 120
prob13.2, 121
prob8.1a, 121
prob8.1b, 122
prob8.1c, 123
prob8.1d, 123
prob9.6c1, 124
prob9.6c2, 125
prob9.6c3, 125
prob9.6c4, 126
rate, 128
ss08, 131
ss08.1850, 131
starwort.ex, 132
sunspot.classic, 133
sunspot2.0, 133
sunspot2.0.month, 134
table10.1.noise, 135
table10.1.signal, 135
table7.1, 136
tesla, 137
tx.unemp.adj, 143
tx.unemp.unadj, 143
us.retail, 145
uspop, 145
wages, 146
whale, 147
wtcrude, 148
wtcrude2020, 148
yellowcab.precleaned, 149

* difference
  artrans.wge, 13
* discrete wavelet transform
  plotts.dwt.wge, 108
* dual
  trans.to.dual.wge, 137
  trans.to.original.wge, 138
* estimation
  est.farma.wge, 32
  est.garma.wge, 33
* filtering
  butterworth.wge, 18
  kalman.miss.wge, 87
  kalman.wge, 88
* forecasting
  expsmooth.wge, 35
* forecasts
  fore.farma.wge, 68
  fore.garma.wge, 69
  fore.glambda.wge, 70
* forecast
  fore.sigplusnoise.wge, 71
* fractional
  true.farma.aut.wge, 141
INDEX

* g(lambda) process
  gen.glamb.d.wge, 81
* g(lambda)
  est.glamb.d.wge, 34
* generate
  gen.sigplusnoise.wge, 82
* instantaneous spectrum
  is.glamb.d.wge, 85
  is.sample.wge, 86
* missing
  kalman.miss.wge, 87
* multiresolution analysis
  plotts.mra.wge, 109
* partial autocorrelations
  pacfts.wge, 103
* periodogram
  period.wge, 106
  plotts.parzen.wge, 110
* plot
  plotts.dwt.wge, 108
  plotts.mra.wge, 109
* predicting
  kalman.miss.wge, 87
  kalman.wge, 88
* residuals
  backcast.wge, 14
* rolling
  roll.win.rmse.nn.wge, 128
  roll.win.rmse.wge, 129
* signal plus noise
  fore.sigplusnoise.wge, 71
* signal-plus-noise
  gen.sigplusnoise.wge, 82
* smoothing
  expsmooth.wge, 35
  kalman.miss.wge, 87
  kalman.wge, 88
* spectral density
  true arma.spec.wge, 140
* spectrum
  plotts.parzen.wge, 110
* time varying spectrum
  is.glamb.d.wge, 85
  is.sample.wge, 86
* transformation
  artrans.wge, 13
* white noise
  ljung.wge, 91

* window
  plotts.parzen.wge, 110
  roll.win.rmse.nn.wge, 128
  roll.win.rmse.wge, 129

aic.ar.wge, 6
aic.burg.wge, 7
aic.wge, 8
aic5.ar.wge, 9
aic5.wge, 10
airline, 11
airlog, 11
ample.spec.wge, 12
appy, 13
artrans.wge, 13
backcast.wge, 14
bat, 15
bitcoin, 16
Bsales, 16
bumps16, 17
bumps256, 17
butterworth.wge, 18
cardiac, 19
cement, 19
chirp, 20
cow, 21
dfw.2011, 22
dfw.mon, 22
dfw.yr, 23
doppler, 24
doppler2, 24
dowannual, 25
dow.rate, 25
dow1000, 26
dow1985, 27
dowjones2014, 27
eco.cd6, 28
eco.corp.bond, 28
eco.mort30, 29
est.ar.wge, 30
est.arma.wge, 31
est.farma.wge, 32
est.garma.wge, 33
est.glamb.d.wge, 34
expsmooth.wge, 35
<table>
<thead>
<tr>
<th>Factor</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>factor.comp.wge</td>
<td>36</td>
</tr>
<tr>
<td>factor.wge</td>
<td>37</td>
</tr>
<tr>
<td>fig1.10a</td>
<td>38</td>
</tr>
<tr>
<td>fig1.10b</td>
<td>38</td>
</tr>
<tr>
<td>fig1.10c</td>
<td>39</td>
</tr>
<tr>
<td>fig1.10d</td>
<td>40</td>
</tr>
<tr>
<td>fig1.16a</td>
<td>40</td>
</tr>
<tr>
<td>fig1.21a</td>
<td>41</td>
</tr>
<tr>
<td>fig1.22a</td>
<td>42</td>
</tr>
<tr>
<td>fig1.5</td>
<td>42</td>
</tr>
<tr>
<td>fig10.11x</td>
<td>43</td>
</tr>
<tr>
<td>fig10.11y</td>
<td>44</td>
</tr>
<tr>
<td>fig10.1bond</td>
<td>44</td>
</tr>
<tr>
<td>fig10.1cd</td>
<td>45</td>
</tr>
<tr>
<td>fig10.1mort</td>
<td>46</td>
</tr>
<tr>
<td>fig10.3x1</td>
<td>46</td>
</tr>
<tr>
<td>fig10.3x2</td>
<td>47</td>
</tr>
<tr>
<td>fig11.12</td>
<td>48</td>
</tr>
<tr>
<td>fig11.4a</td>
<td>48</td>
</tr>
<tr>
<td>fig12.1a</td>
<td>49</td>
</tr>
<tr>
<td>fig12.1b</td>
<td>50</td>
</tr>
<tr>
<td>fig13.18a</td>
<td>50</td>
</tr>
<tr>
<td>fig13.2c</td>
<td>51</td>
</tr>
<tr>
<td>fig3.10d</td>
<td>52</td>
</tr>
<tr>
<td>fig3.16a</td>
<td>52</td>
</tr>
<tr>
<td>fig3.18a</td>
<td>53</td>
</tr>
<tr>
<td>fig3.24a</td>
<td>54</td>
</tr>
<tr>
<td>fig3.29a</td>
<td>54</td>
</tr>
<tr>
<td>fig4.8a</td>
<td>55</td>
</tr>
<tr>
<td>fig5.3c</td>
<td>56</td>
</tr>
<tr>
<td>fig6.11a</td>
<td>56</td>
</tr>
<tr>
<td>fig6.1nf</td>
<td>57</td>
</tr>
<tr>
<td>fig6.2nf</td>
<td>58</td>
</tr>
<tr>
<td>fig6.5nf</td>
<td>58</td>
</tr>
<tr>
<td>fig6.6nf</td>
<td>59</td>
</tr>
<tr>
<td>fig6.7nf</td>
<td>60</td>
</tr>
<tr>
<td>fig6.8nf</td>
<td>60</td>
</tr>
<tr>
<td>fig8.11a</td>
<td>61</td>
</tr>
<tr>
<td>fig8.4a</td>
<td>62</td>
</tr>
<tr>
<td>fig8.6a</td>
<td>62</td>
</tr>
<tr>
<td>fig8.8a</td>
<td>63</td>
</tr>
<tr>
<td>flu</td>
<td>64</td>
</tr>
<tr>
<td>fore.arima.wge</td>
<td>64</td>
</tr>
<tr>
<td>fore.arma.wge</td>
<td>66</td>
</tr>
<tr>
<td>fore.aruma.wge</td>
<td>67</td>
</tr>
<tr>
<td>fore.farma.wge</td>
<td>68</td>
</tr>
<tr>
<td>fore.garma.wge</td>
<td>69</td>
</tr>
<tr>
<td>fore.glambda.wge</td>
<td>70</td>
</tr>
<tr>
<td>fore.sigplusnoise.wge</td>
<td>71</td>
</tr>
<tr>
<td>freeze</td>
<td>72</td>
</tr>
<tr>
<td>freight</td>
<td>73</td>
</tr>
<tr>
<td>gegenb.wge</td>
<td>73</td>
</tr>
<tr>
<td>gen.arch.wge</td>
<td>74</td>
</tr>
<tr>
<td>gen.arima.wge</td>
<td>75</td>
</tr>
<tr>
<td>gen.arma.wge</td>
<td>76</td>
</tr>
<tr>
<td>gen.aruma.wge</td>
<td>77</td>
</tr>
<tr>
<td>gen.garch.wge</td>
<td>78</td>
</tr>
<tr>
<td>gen.garma.wge</td>
<td>79</td>
</tr>
<tr>
<td>gen.geg.wge</td>
<td>80</td>
</tr>
<tr>
<td>gen.glambda.wge</td>
<td>81</td>
</tr>
<tr>
<td>gen.sigplusnoise.wge</td>
<td>82</td>
</tr>
<tr>
<td>global.temp</td>
<td>83</td>
</tr>
<tr>
<td>global2020</td>
<td>83</td>
</tr>
<tr>
<td>hadley</td>
<td>84</td>
</tr>
<tr>
<td>hilbert.wge</td>
<td>85</td>
</tr>
<tr>
<td>is.glambda.wge</td>
<td>85</td>
</tr>
<tr>
<td>is.sample.wge</td>
<td>86</td>
</tr>
<tr>
<td>kalman.miss.wge</td>
<td>87</td>
</tr>
<tr>
<td>kalman.wge</td>
<td>88</td>
</tr>
<tr>
<td>kingkong</td>
<td>89</td>
</tr>
<tr>
<td>lavon</td>
<td>90</td>
</tr>
<tr>
<td>lavon15</td>
<td>90</td>
</tr>
<tr>
<td>linearchirp</td>
<td>91</td>
</tr>
<tr>
<td>ljung.wge</td>
<td>91</td>
</tr>
<tr>
<td>llynx</td>
<td>92</td>
</tr>
<tr>
<td>lynx</td>
<td>93</td>
</tr>
<tr>
<td>ma.pred.wge</td>
<td>94</td>
</tr>
<tr>
<td>ma.smooth.wge</td>
<td>95</td>
</tr>
<tr>
<td>ma2.table7.1</td>
<td>96</td>
</tr>
<tr>
<td>macoef.geg.wge</td>
<td>96</td>
</tr>
<tr>
<td>mass.mountain</td>
<td>97</td>
</tr>
<tr>
<td>MedDays</td>
<td>98</td>
</tr>
<tr>
<td>mm.eq</td>
<td>98</td>
</tr>
<tr>
<td>mult.wge</td>
<td>99</td>
</tr>
<tr>
<td>NAICS</td>
<td>100</td>
</tr>
<tr>
<td>nbumps256</td>
<td>100</td>
</tr>
<tr>
<td>nile.min</td>
<td>101</td>
</tr>
<tr>
<td>noctula</td>
<td>102</td>
</tr>
<tr>
<td>NSA</td>
<td>102</td>
</tr>
<tr>
<td>ozona</td>
<td>103</td>
</tr>
</tbody>
</table>
INDEX

pacfts.wge, 103  
parzen.wge, 104  
patemp, 105  
period.wge, 106  
pi.weights.wge, 107  
plotts.dwt.wge, 108  
plotts.mra.wge, 109  
plotts.parzen.wge, 110  
plotts.sample.wge, 111  
plotts.true.wge, 112  
plotts.wge, 113  
prob10.4, 114  
prob10.6x, 115  
prob10.6y, 115  
prob10.7x, 116  
prob10.7y, 117  
prob11.5, 117  
prob12.1c, 118  
prob12.3a, 119  
prob12.3b, 119  
prob12.6c, 120  
prob13.2, 121  
prob8.1a, 121  
prob8.1b, 122  
prob8.1c, 123  
prob8.1d, 123  
prob9.6c1, 124  
prob9.6c2, 125  
prob9.6c3, 125  
prob9.6c4, 126  
psi.weights.wge, 127

rate, 128
roll.win.rmse.nn.wge, 128
roll.win.rmse.wge, 129

sample.spec.wge (ample.spec.wge), 12
slr.wge, 130
ss08, 131
ss08.1850, 131
starwort.ex, 132
sunspot.classic, 133
sunspot2.0, 133
sunspot2.0.month, 134

table10.1.noise, 135
table10.1.signal, 135
table7.1, 136
tesla, 137

trans.to.dual.wge, 137
trans.to.original.wge, 138
true.arma.aut.wge, 139
true.arma.spec.wge, 140
true.farma.aut.wge, 141
true.garma.aut.wge, 142
tswge (tswge-package), 5
tswge-package, 5
tx.unemp.adj, 143
tx.unemp.unadj, 143
unit.circle.wge, 144
us.retail, 145
uspop, 145

wages, 146
wbg.boot.wge, 146
whale, 147
wtcrude, 148
wtcrude2020, 148
wv.wge, 149

yellowcab.precleaned, 149