Package ‘tswge’

October 14, 2022

Type Package
Title Time Series for Data Science
Version 2.0.0
Date 2022-7-22
Author Wayne Woodward
Maintainer Bivin Sadler <bsadler@smu.edu>
Imports signal,PolynomF,MASS,waveslim,astsa,tidyverse,zoo,plotrix,
dplyr, ggplot2, magrittr,nnfor,forecast
License GPL-2
NeedsCompilation no
LazyData true
Repository CRAN
Date/Publication 2022-08-09 08:00:02 UTC

R topics documented:

tswge-package .................................................. 5
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
c.o.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
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.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:

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<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>61</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>
</tbody>
</table>
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
maceof.geom.wge ......................... 96
mass.mountain .......................... 97
MedDays .................................. 98
mn.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 ................................ 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
Author(s)
Wayne Woodward <waynew@smu.edu>

References

Examples

data(wages)
plotts.wge(wages)

---

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

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

---

**aic.burg.wge**

*AR Model Identification using Burg Estimates*

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

Description
ARMA model identification using either AIC, AICC, or BIC

Usage
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
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

\[
aic5.ar.wge(x, p = 0:5, type = "aic", method='mle')
\]

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

\[
data(fig3.18a)
  aic5.wge(fig3.18a,p=0:5,q=0:2)
\]
**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

Classical Airline Passenger Data

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

Usage

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

data(airline)

Description

Natural log of airline data

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

Usage

data("airlog")

Format

The format is: num [1:144] 4.72 4.77 4.88 4.86 4.8 ...
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 coefficients phi1 and phi2, for example, artrans.wge computes $y(t)=x(t)-\phi_1 x(t-1)-\phi_2 x(t-2)$, for $t=3, \ldots, n$

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 plottr=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**

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

---

**backcast.wge**

*Calculate backcast residuals*

**Description**

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

**Usage**

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
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, -0.6), 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
Examples

data(Bsales)

bumps16

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

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 ...
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
**cardiac**

**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)

data(chirp)

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

\texttt{co.wge(x, maxp=5)}

Arguments

\begin{itemize}
\item \texttt{x} \hspace{1cm} \text{Realization}
\item \texttt{maxp} \hspace{1cm} \text{Maximum AR order allowed for AR model fit to residuals from least squares line}
\end{itemize}

Value

\begin{itemize}
\item \texttt{z} \hspace{1cm} \text{Residuals from the fitted line}
\item \texttt{b0hat} \hspace{1cm} \text{Estimated y-intercept of the fitted line using the CO method}
\item \texttt{b1hat} \hspace{1cm} \text{Estimated slope of the fitted line using the CO method}
\item \texttt{z.order} \hspace{1cm} \text{Order, } p \text{, fit to the residuals}
\item \texttt{z.phi} \hspace{1cm} \text{Coefficients of the AR model fit to the residuals}
\item \texttt{pvalue} \hspace{1cm} \text{P-value of the CO test for the significance of the slope}
\item \texttt{tco} \hspace{1cm} \text{Cochrane-Orcutt test statistic.}
\end{itemize}

Author(s)

Wayne Woodward

References


Examples

\begin{verbatim}
data(global.temp)
co.wge(global.temp, maxp=5)
\end{verbatim}
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 Science: Analysis and Forecasting" by Woodward, Sadler, and Robertson

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

---

<table>
<thead>
<tr>
<th><code>dfw.yr</code></th>
<th><strong>DFW Annual Temperatures</strong></th>
</tr>
</thead>
</table>

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

**Usage**
```r
data("dfw.yr")
```

**Format**
ts object consisting of annual data from 1900 through 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.yr)
```
doppler  

**Doppler Data**

### Description
Generated Doppler data

### Usage
```r
data("doppler")
```

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

### Source
Simulated

### References

### Examples
```r
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
```r
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**

The format is: `ts` object consisting of DOW Annual closing averages from 1915 through 2020

**References**

"Time Series for Data Science: 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 ...
dow1000

Source

Public access

References

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

Examples

data(dow.rate)

---------------------------------------------------------------------
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**

`ts` object consisting of daily Dow closing prices from 1985 through 2020

**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)

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)
**est.ar.wge**

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**

```r
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

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.

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

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

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=.1,high.d=.5,inc.d=.01,p.max=3)

est.garma.wge(Nile,low.u=0.1,high.u=0.5,inc.u=0.01,low.lambda=0.1,high.lambda=0.5,inc.lambda=0.01,p.max=3,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.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
data Vector containing the TVF realization to be analyzed
lambda.range Range of lambda values considered in the search
offset.range Range 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
Exponential Smoothing

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))
<table>
<thead>
<tr>
<th>fig1.10a</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Description</strong></td>
</tr>
<tr>
<td>This is the sum of the three signals in fig1.10b, fig1.10c, and fig1.10d</td>
</tr>
<tr>
<td><strong>Usage</strong></td>
</tr>
<tr>
<td>data(&quot;fig1.10a&quot;)</td>
</tr>
<tr>
<td><strong>Format</strong></td>
</tr>
<tr>
<td>The format is: num [1:1000] 0.0217 -0.1528 -0.3141 -0.4613 -0.5934 ...</td>
</tr>
<tr>
<td><strong>Source</strong></td>
</tr>
<tr>
<td>Simulated data</td>
</tr>
<tr>
<td><strong>References</strong></td>
</tr>
<tr>
<td><strong>Examples</strong></td>
</tr>
<tr>
<td>data(fig1.10a)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>fig1.10b</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Description</strong></td>
</tr>
<tr>
<td>Low frequency component of Figure 1.10a</td>
</tr>
<tr>
<td><strong>Usage</strong></td>
</tr>
<tr>
<td>data(&quot;fig1.10b&quot;)</td>
</tr>
<tr>
<td><strong>Format</strong></td>
</tr>
<tr>
<td>The format is: num [1:1000] 1 1 0.999 0.998 0.997 ...</td>
</tr>
<tr>
<td><strong>Source</strong></td>
</tr>
<tr>
<td>Simulated data</td>
</tr>
</tbody>
</table>
References


Examples

```r
data(fig1.10b)
```

Description

Middle frequencies component in Figure 1.10a

Usage

```r
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

```r
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)
**fig1.22a**  \hspace{2cm} *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**

```r
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**

```r
data(fig1.22a)
```

---

**fig1.5**  \hspace{2cm} *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**

```r
data("fig1.5")
```

**Format**

The format is: num [1:100] 0.739 -0.390 0.150 -0.627 0.262 ...

**Source**

Simulated data
References


Examples

```r
data(fig1.5)
```

---

**Description**

Simulated unobservable AR(1) data in Example 10.11

**Usage**

```r
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**

```r
data(fig10.11x)
```
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)

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-April 1, 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**

data("fig10.1mort")

**Format**


**Source**

Internet

**References**


**Examples**

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**

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)

Variable X2 for the bivariate realization shown in Figure 10.3

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**
```r
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**
```r
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**
```r
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)

---

**fig12.1a**  
*Simulated data with two frequencies shown in Figure 12.1a in Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott*

**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*

**Description**

Simulated two-frequency AR(4) data

**Usage**

```r
data("fig12.1b")
```

**Format**

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

**Source**

Simulated data

**References**


**Examples**

```r
data(fig12.1b)
```

---

**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 AR(4) data

**Usage**

```r
data("fig13.18a")
```

**Format**

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

---
**Source**

Simulated data

**References**


**Examples**

data(fig13.18a)

---

**Description**

Realization from an Euler(2) model

**Usage**

data("fig13.2c")

**Format**

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

**Source**

Simulated data

**References**


**Examples**

data(fig13.2c)
**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 ...
Source
Simulated data

References

Examples
data(fig3.16a)

data(fig3.18a)

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)
**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)

---

**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)

data(fig4.8a)

---

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("fig4.8")

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 \((1-0.995B)(1-1.2B+.8B^2)X(t)=a(t)\)

**Usage**
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**
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**
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)

---

**fig6.1nf**

*Data in Figure 6.1 without the forecasts*

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)
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)`

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
fig6.6nf

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

Usage
data("fig6.6nf")

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

Source
Generated data

References

Examples
data(fig6.6nf)
**fig6.7nf**  
*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**
```r
data("fig6.7nf")
```

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

**Source**
Generated data

**References**

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

---

**fig6.8nf**  
*Simulated seasonal data with s=12*

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

**Usage**
```r
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

data(fig6.8nf)

---

fig8.11a  Data for Figure 8.11a in Applied Time Series Analysis with R, second edition by Woodward, Gray, and Elliott

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

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

data(fig8.11a)
**fig8.4a**

*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)

---

**fig8.6a**

*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(fig8.8a)

Fig 8.8a

Data for Figure 8.8a in Applied time series Analysis with R, second edition by Woodward, Gray, and Elliott

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)
flu

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)
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

```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.arima.wge(x,phi=phi12,d=1,s=12,n.ahead=12,limits=FALSE)
```
Description

Forecasts and associated plots for an ARMA model

Usage

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

**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
- **x**: Realization to be analyzed
- **d**: Parameter d in FARMA model
- **phi**: Coefficients of the AR component of the FARMA model
- **theta**: Coefficients of the MA component of the FARMA 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
fore.garma.wge

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)
```

---

**fore.garma.wge**

*Forecast using a GARMA model*

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
Forecasts based on a G(lambda) model

**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)
```

**Description**

Find forecasts using a specified G(lambda) model

**Usage**

```r
fore.glalpha.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

- **f.ar**: Forecasts using AR model fit to data in original time
- **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

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

<table>
<thead>
<tr>
<th>f</th>
<th>The n.ahead forecasts</th>
</tr>
</thead>
<tbody>
<tr>
<td>l1</td>
<td>The lower limits for the forecasts. zeros are returned if limits were not requested</td>
</tr>
<tr>
<td>ul</td>
<td>The upper limits for the forecasts. zeros are returned if limits were not requested</td>
</tr>
<tr>
<td>res</td>
<td>Residuals</td>
</tr>
<tr>
<td>wnv</td>
<td>The estimated white noise variance based on the residuals</td>
</tr>
<tr>
<td>se</td>
<td>se is the estimated standard error of the k step ahead forecast. zeros are returned if limits were not requested</td>
</tr>
<tr>
<td>xi</td>
<td>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</td>
</tr>
</tbody>
</table>

Author(s)

Wayne Woodward

References


Examples

data(illynx)
illynx.for=fore.sigplusnoise.wge(illynx,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

data(freeze)

Description

9 years of monthly freight shipment data

Usage

data("freight")

Format

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

Source

Unknown

References


Examples

data(freight)

gegenb.wge

Calculates Gegenbauer polynomials

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

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

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

Description

Generates a realization of length n from the GARCH(q0) model (4.23) in "Applied Time Series Analysis with R, 2nd edition" by Woodward, Gray, and Elliott

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 alphaq0

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))
```

---

**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

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

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

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n</td>
<td>Length of realization to be generated</td>
</tr>
<tr>
<td>phi</td>
<td>Vector of AR coefficients</td>
</tr>
<tr>
<td>theta</td>
<td>Vector of MA coefficients</td>
</tr>
<tr>
<td>vara</td>
<td>White noise variance, default=1</td>
</tr>
<tr>
<td>mu</td>
<td>Theoretical mean, default=0</td>
</tr>
<tr>
<td>plot</td>
<td>Logical: TRUE=plot, FALSE=no plot</td>
</tr>
<tr>
<td>sn</td>
<td>determines the seed used in the simulation. sn=0 produces new/random realization each time. sn=positive integer produces same realization each time</td>
</tr>
</tbody>
</table>

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

Author(s)
Wayne Woodward

References
Examples

\texttt{gen.aruma.wge(n=100, phi=c(1.6,-.9), theta=.8, mu=50, vara=1, plot=TRUE)}

---

\textbf{Description}

This function calls \texttt{arima.sim} but in a similar manner to \texttt{gen.ns.arma.wge} and \texttt{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)

\textbf{Usage}

\texttt{gen.aruma.wge(n,phi=0,theta=0,d=0,s=0,lambda=0,vara=1,plot=TRUE,sn=0)}

\textbf{Arguments}

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

\textbf{Value}

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

\textbf{Author(s)}

Wayne Woodward

\textbf{References}


\textbf{Examples}

\texttt{gen.aruma.wge(n=100,phi=.7,theta=0, d=1, s=4,lambda=c(1.8,-1),vara=1, plot=TRUE)}
Generate a realization from a GARCH\((p_0,q_0)\) model

Description

Generates a realization of length \(n\) from the GARCH\((p_0,q_0)\) model (4.26) in "Applied Time Series Analysis with R, 2nd edition" by Woodward, Gray, and Elliott.

Usage

\[
\text{gen.garch.wge}(n, \alpha_0, \alpha, \beta, \text{plot=TRUE}, \text{sn=0})
\]

Arguments

- \(n\): Length of realization to be generated
- \(\alpha_0\): The constant \(\alpha_0\) in model (4.23)
- \(\alpha\): A vector of length \(q_0\) containing \(\alpha_1\) through \(\alpha_{q_0}\)
- \(\beta\): A vector of length \(p_0\) containing \(\beta_1\) through \(\beta_{p_0}\)
- \(\text{plot}\): If \(\text{plot=TRUE}\) (default) the generated realization is plotted
- \(\text{sn}\): Determines the seed used in the simulation. \(\text{sn=0}\) produces new/random realization each time. \(\text{sn=positive integer}\) produces same realization each time

Value

Returns the generated realization

Author(s)

Wayne Woodward

References


Examples

\[
\text{gen.garch.wge}(n=200, \alpha_0=.1, \alpha=.45, \beta=.45)
\]
Function to generate a GARMA realization

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

Usage
```
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
```
gen.garma.wge(n=100, u=.8,lambda=.4,phi=.9)
```
Function to generate a Gegenbauer realization

Description
This function calls macoef.wge

Usage
```
gen.geg.wge(n, u, lambda, trun = 300000, 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

gen.glambda.wge(n, lambda, phi =0, offset = 20, vara = 1, plot = 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
- **vara**: White noise variance, default=1
- **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
- **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

gen.glambda.wge(n=500, lambda=0.5, phi=c(1.9,-.99), vara=1, plot=TRUE, sn=0)
Description

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

Usage

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

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)
- \( \psi \): a 2-component vector specifying the phase shift (0 to 2\( \pi \))
- \( \phi \): a vector of coefficients of the coefficients of the AR noise
- \( \text{vara} \): vara is the variance of the noise. NOTE: \( a(t) \) is a vector of \( N(0, WNV) \) 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

\[
\text{x=} \text{gen.sigplusnoise.wge}(n=100, \text{coef}=\text{c}(3,1), \text{freq}=\text{c}(.1,.4), \psi=\text{c}(0,0), \text{vara}=2)
\]
**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

```r
data(global2020)
```

---

```r
data(hadley)
```

### 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

```r
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

```r
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.glambda.wge

**Instantaneous spectrum**

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

**Usage**
is.glambda.wge(n, phi = 0, sigma2 = 1, lambda, offset)
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.glambdawge(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 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

```r
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)
```

---

<table>
<thead>
<tr>
<th>kalman.wge</th>
<th>Kalman filter for simple signal plus noise model</th>
</tr>
</thead>
</table>

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

```r
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
```r
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
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**

```r
data("linearchirp")
```

**Format**

The format is: List of 2  

```r
$x:$ 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**

```r
data(linearchirp)
```

---

**ljung.wge**  
*Ljung-Box Test*

**Description**

Performs Ljung-Box Test for white noise

**Usage**

```r
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)

**Table 1**

| **1lynx** | 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("1lynx")

Format

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

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

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
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
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
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
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
data("mass.mountain")

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

Source

References

Examples
data(mass.mountain)
MedDays

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

**Examples**

```r
data(mm.eq)
```

### 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

```r
mult.wge(fac1 = 0, fac2 = 0, fac3 = 0, fac4 = 0, fac5 = 0, 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
- **model.coef**: Model coefficients of the full model using notation in "Applied Time Series Analysis, 2nd edition" by Woodward, Gray, and Elliott

### Note

Requires CRAN package 'PolynomF'

### Author(s)

Wayne Woodward

### References


### Examples

```r
fac1=c(1.6,-.9)
fac2=.8
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)

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)

---

**ozona**  

*Daily Number of Chicken-Fried Steaks Sold*

**Description**

Daily number of chicken-fried steaks sold at Ozona Bar and Grill during June and July 2019

**Usage**

```
data("ozona")
```

**Format**

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)
```

---

**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')

parzen.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

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**

```r
parzen.wge(rnorm(100))
```

---

**patemp**

*Pennsylvania average monthly temperatures*

**Description**

Pennsylvania average monthly temperatures

**Usage**

```r
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**

```r
data(patemp)
```
period.wge Calculate the periodogram

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))
pi.weights.wge

Description

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 pi(1), pi(2), ..., pi(lag.max). Note that psi(0)=1.

Value

A vector containing pi(1), ..., pi(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

```r
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. \(n\) 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. This 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')
```
Description

Plots MRA :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

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

Description

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

Usage

plotts.parzen.wge(x, m2=c(0,0))

Arguments

x  The vector containing the time series realization

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

freq  Frequencies at which the periodogram and parzen widow estimates are calculated

db  Periodogram (in dB) calculated at the frequencies in freq

dbz  Parzen window estimate (in dB) calculated at the frequencies in freq using truncation point $2\sqrt{n}$

dbz1  Parzen window estimate (in dB) calculated at the frequencies in freq using truncation point $m2[1]$ 

dbz2  Parzen window estimate (in dB) calculated at the frequencies in freq using truncation point $m2[2]$

Author(s)

Wayne Woodward

References

Examples

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

---

**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**

```r
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

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

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**

```r
plotts.true.wge(n=100, phi=c(1.6,-.9), theta=.8, lag.max=25, vara = 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**

```r
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

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

---

prob10.4


Description

Matrix containing a bivariate VAR data set

Usage

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

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
**References**


**Examples**

data(prob11.5)

data(prob12.1c)

---

**prob12.1c** | *Data for Problem 12.1c and 12.3c 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**

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

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

data(prob12.3a)

prob12.3b

Data for Problem 12.3b 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

data("prob12.3b")

Format

The format is: num [1:256] 1 1 1 1 ...

Source

Simulated data
References


Examples

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

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

data(prob12.6c)
prob13.2


Description
Simulated data from cosine-plus-noise model

Usage
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
data(prob13.2)

prob8.1a


Description
See title above

Usage
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)

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)
**prob8.1c**

*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("prob8.1c")

**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)

---

**prob8.1d**

*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("prob8.1d")

**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)

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)

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)
psi.weights.wge

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**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>series</code></td>
<td>The data</td>
</tr>
<tr>
<td><code>horizon</code></td>
<td>The number of observations ahead to be forecasted.</td>
</tr>
<tr>
<td><code>fit_model</code></td>
<td>The mlp object (model) to be evaluated. This model will have been fit before the call to this function.</td>
</tr>
</tbody>
</table>
**Value**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>rwRMSE</td>
<td>The average of the individual RMSEs of each window</td>
</tr>
<tr>
<td>numwindows</td>
<td>The number of windows</td>
</tr>
<tr>
<td>horizon</td>
<td>The number of observations ahead to be forecasted.</td>
</tr>
</tbody>
</table>

**Author(s)**

Bivin Sadler

**References**

"The Time Series Tool Kit"

---

**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.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**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>rwRMSE</td>
<td>The average of the individual RMSEs of each window</td>
</tr>
<tr>
<td>numwindows</td>
<td>The number of windows</td>
</tr>
<tr>
<td>horizon</td>
<td>The number of observations ahead to be forecasted.</td>
</tr>
<tr>
<td>s</td>
<td>Order of the seasonal difference, default=1</td>
</tr>
<tr>
<td>d</td>
<td>Order of the difference</td>
</tr>
<tr>
<td>phis</td>
<td>Vector of AR coefficients</td>
</tr>
<tr>
<td>thetas</td>
<td>Vector of MA coefficients</td>
</tr>
<tr>
<td>RMSEs</td>
<td>Vector of RMSEs ... one for each window</td>
</tr>
</tbody>
</table>
Author(s)

Bivin Sadler

References

"The Time Series Tool Kit"

---

**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**

`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)

data("starwort.ex")

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**

```r
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**

```r
data(sunspot.classic)
```

sunspot2.0  

**Annual Sunspot2.0 Numbers**

**Description**

Annual sunspot2.0 numbers from 1700 through 2020

**Usage**

```r
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)

ds

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(table7.1)

---

**table7.1**  
*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 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**

**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
trans.to.original.wge

**Value**

<table>
<thead>
<tr>
<th>intX</th>
<th>See intY description</th>
</tr>
</thead>
<tbody>
<tr>
<td>intY</td>
<td>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</td>
</tr>
<tr>
<td>h</td>
<td>The output value of the scaling parameter that assures that the dual realization and the original realization are of the same length</td>
</tr>
</tbody>
</table>

**Author(s)**

Wayne Woodward

**References**


**Examples**

data(fig13.2c)

```r
y = trans.to.dual.wge(x = fig13.2c, lambda = -.4, offset = 63)
```

---

trans.to.original.wge  Transforms dual data set back to original time scale

**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**

```r
trans.to.original.wge(xd, lambda, offset, h, plot = TRUE)
```

**Arguments**

<table>
<thead>
<tr>
<th>xd</th>
<th>The dual data set</th>
</tr>
</thead>
<tbody>
<tr>
<td>lambda</td>
<td>The value of lambda in the Glambda time transformation</td>
</tr>
<tr>
<td>offset</td>
<td>The value of offset in the Glambda time transformation</td>
</tr>
<tr>
<td>h</td>
<td>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</td>
</tr>
<tr>
<td>plot</td>
<td>Logical: TRUE=plot, FALSE=no plot</td>
</tr>
</tbody>
</table>

**Value**

Returns the y values to be plotted at time points t=1 to n that approximate the original TVF data set
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

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, vara=1, plot=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
f
  Frequencies at which true spectral density is evaluated: 0, 1/500, 2/500, .... 0.5
spec
  True spectral density calculated at the frequencies in f

Author(s)
Wayne Woodward

References

Examples
true.arma.spec.wge(phi=c(1.6, -.9), theta=-.8, lag.max=15, vara=1)
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 editon" 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

"Applied Time Series Analysis with R, second editon" by Woodward, Gray, and Elliott
tx.unemp.adj

Examples
   y=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
   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
   data(tx.unemp.adj)

---

tx.unemp.unadj Texas Unadjusted Unemployment Rates

Description
   Monthly unemployment rate in Texas for the years 2000-2019

Usage
   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

Plot the roots of the characteristic equation on the complex plane.

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 Sience: 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)
whale

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

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

data(whale)

whale  Whale click data

Description

256 point whale click echolocation signal

Usage

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

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 Science: 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
```r
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
```r
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
```r
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.arma.wge, 66
  fore.aruma.wge, 67
* Fractional
  fore.farma.wge, 68
* G(lambda) model
  is.glambda.wge, 85
  is.sample.wge, 86
* G(lambda)
  fore.glambda.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.greg.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.glambda.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

fig6.1nf, 57
fig6.2nf, 58
fig6.5nf, 58
fig6.6nf, 59
fig6.7nf, 60
fig6.8nf, 60
fig8.11a, 61
flu, 64
freeze, 72
freight, 73
global.temp, 83
global2020, 83
hadley, 84
kingKong, 89
lavon, 90
lavon15, 90
linearChirp, 91
lllynx, 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
* g(\(\lambda\)) process
  gen.glamb.a. 81
* g(\(\lambda\))
  est.glamb.a. 34
* generate
  gen.sigm.plusnoise.a. 82
* instantaneous spectrum
  is.glamb.a. 85
  is.sample.a. 86
* missing
  kalman.miss.a. 87
* multiresolution analysis
  plotts.mra.a. 109
* partial autocorrelations
  pacfts.a. 103
* periodogram
  period.a. 106
  plotts.parzen.a. 110
* plot
  plotts.dwt.a. 108
  plotts.mra.a. 109
* predicting
  kalman.miss.a. 87
  kalman.a. 88
* residuals
  backcast.a. 14
* rolling
  roll.win.rmse.nn.a. 128
  roll.win.rmse.a. 129
* signal plus noise
  fore.sig.plusnoise.a. 71
* signal-plus-noise
  gen.sigplusnoise.a. 82
* smoothing
  expsmooth.a. 35
  kalman.miss.a. 87
  kalman.a. 88
* spectral density
  true arma.spec.a. 140
* spectrum
  plotts.parzen.a. 110
* time varying spectrum
  is.glamb.a. 85
  is.sample.a. 86
* transformation
  artrans.a. 13
* white noise
  ljung.a. 91
* window
  plotts.parzen.a. 110
  roll.win.rmse.nn.a. 128
  roll.win.rmse.a. 129
  aic.ar.a. 6
  aic.burg.a. 7
  aic.a. 8
  aic5.ar.a. 9
  aic5.a. 10
  airline.11
  airlog.11
  ample.spec.a. 12
  appy.13
  artrans.a. 13
  backcast.a. 14
  bat.15
  bitcoin.16
  Bsales.16
  bumps16.17
  bumps256.17
  butterworth.a. 18
  cardiac.19
  cement.19
  chirp.20
  co.a. 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
  eco.cd6.28
  eco.corp.bond.28
  eco.mort30.29
  est.ar.a. 30
  est.arma.a. 31
  est.farma.a. 32
  est.garma.a. 33
  est.glamb.a. 34
  expsmooth.a. 35
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.pre cleaned, 149