Package 'astsa'

February 19, 2015

Type Package

Title Applied Statistical Time Series Analysis

Version 1.3

Date 2014-11-01

Author David Stoffer

Maintainer David Stoffer <stoffer@pitt.edu>

Description Data sets and scripts for Time Series Analysis and Its Applications: With R Examples by Shumway and Stoffer, 3rd edition

URL http://www.stat.pitt.edu/stoffer/tsa3/

License GPL-2

LazyLoad yes

LazyData yes

NeedsCompilation no

Repository CRAN

Date/Publication 2014-10-27 10:10:48

R topics documented:

`astsa-package` *Applied Statistical Time Series Analysis*

Description

Includes data and scripts to accompany *Time Series Analysis and Its Applications: With R Examples* (3rd ed) by R.H. Shumway and D.S. Stoffer. Springer Texts in Statistics, 2011.

Details

Package:	astsa
Type:	Package
Version:	1.3
Date:	2014-11-01
License:	GPL-2
LazyLoad:	yes
LazyData:	yes

Author(s)

David Stoffer <stoffer@pitt.edu>

References

See the webpage for the text: http://www.stat.pitt.edu/stoffer/tsa3/ or its mirror http://lib.stat.cmu.edu/general/stoffer/tsa3/

`acf2` *Plot and print ACF and PACF of a time series*

Description

Produces a simultaneous plot (and a printout) of the sample ACF and PACF on the same scale. The zero lag value of the ACF is removed.

Usage

```
acf2(series, max.lag = NULL, ...)
```

Arguments

`series`	The data. Does not have to be a time series object.
`max.lag`	Maximum lag. Can be omitted. Defaults to $\sqrt{n} + 10$ unless $n < 50$.
`...`	Addtional arguements passed to `acf`

Details

This is bacisally a wrapper for `acf()` provided in `tseries`. The error bounds are approximate white noise bounds, $0 \pm 2/\sqrt{n}$; no other option is given.

Value

`ACF`	The sample ACF
`PACF`	The sample PACF

Author(s)

D.S. Stoffer

References

http://www.stat.pitt.edu/stoffer/tsa3/

Examples

```
acf2(rnorm(100))
acf2(rnorm(100), 25)
```

---

ar1boot *Data used in Example 3.35 on page 137.*

Description

Data used in Example 3.35 on page 137.

Usage

```
data(ar1boot)
```

Format

The format is: Time-Series [1:100] from 1 to 100: 61.8 60.1 60.1 59.1 58.3 ...

ar1miss — *Data for Problem 6.14 on page 403.*

Description

Data for Problem 6.14 on page 403.

Usage

```
data(ar1miss)
```

Format

The format is: Time-Series [1:100] from 1 to 100: 1.01 0 1.05 1.75 2.25 ...

References

http://www.stat.pitt.edu/stoffer/tsa3/

arf — *Simulated ARFIMA*

Description

1000 simulated observations from an ARFIMA(1, 1, 0) model with $\phi = .75$ and $d = .4$.

Usage

```
data(arf)
```

Format

The format is: Time-Series [1:1000] from 1 to 1000: -0.0294 0.7487 -0.3386 -1.0332 -0.2627 ...

References

http://www.stat.pitt.edu/stoffer/tsa3/

arma.spec *Spectral Density of an ARMA Model*

Description

Gives the ARMA spectrum (on a log scale), tests for causality, invertibility, and common zeros. See Example 4.6 on page 184.

Usage

```
arma.spec(ar = 0, ma = 0, var.noise = 1, n.freq = 500, ...)
```

Arguments

`ar`	vector of AR parameters
`ma`	vector of MA parameters
`var.noise`	variance of the noise
`n.freq`	number of frequencies
`...`	additional arguments

Details

The basic call is `arma.spec(ar, ma)` where `ar` and `ma` are vectors containing the model parameters. Use `log="no"` if you do not want the plot on a log scale. If the model is not causal or invertible an error message is given. If there are common zeros, a spectrum will be displayed and a warning will be given; e.g., `arma.spec(ar= .9, ma= -.9)` will yield a warning and the plot will be the spectrum of white noise. See Example 4.6 on page 184.

Value

`freq`	frequencies - returned invisibly
`spec`	spectral ordinates - returned invisibly

Note

In `tsa3`, this is called `spec.arma`.

Author(s)

D.S. Stoffer

References

http://www.stat.pitt.edu/stoffer/tsa3/

Examples

```
arma.spec(ar = c(1, -.9), ma = .8, log="no")
```

beamd — *Infrasonic Signal from a Nuclear Explosion*

Description

Infrasonic signal from a nuclear explosion. See Example 7.2 on page 421.

Usage

```
data(beamd)
```

Format

A data frame with 2048 observations (rows) on 3 numeric variables (columns): `sensor1`, `sensor2`, `sensor3`.

Details

This is a data frame consisting of three columns (that are not time series objects). The data are an infrasonic signal from a nuclear explosion observed at sensors on a triangular array.

References

http://www.stat.pitt.edu/stoffer/tsa3/

birth — *U.S. Monthly Live Births*

Description

Monthly live births (adjusted) in thousands for the United States, 1948-1979.

Usage

```
data(birth)
```

Format

The format is: Time-Series [1:373] from 1948 to 1979: 295 286 300 278 272 268 308 321 313 308 ...

References

http://www.stat.pitt.edu/stoffer/tsa3/

blood *Daily Blood Work*

Description

Multiple time series of measurements made for 91 days on the three variables, log(white blood count) [WBC], log(platelet) [PLT] and hematocrit [HCT]. Missing data code is NA.

Usage

```
data(blood)
```

Format

The format is: mts [1:91, 1:3]

Details

This is the data set used in Example 6.1 and subsequently in Example 6.9 with 0 (zero) as the missing data code.

Source

Jones, R.H. (1984). Fitting multivariate models to unequally spaced data. In *Time Series Analysis of Irregularly Observed Data*, pp. 158-188. E. Parzen, ed. Lecture Notes in Statistics, 25, New York: Springer-Verlag.

References

http://www.stat.pitt.edu/stoffer/tsa3/

See Also

HCT, PLT, WBC

Examples

```
## Not run: plot(blood, type="o", pch=19)
```

bnrf1ebv *Nucleotide sequence - BNRF1 Epstein-Barr*

Description

Nucleotide sequence of the BNRF1 gene of the Epstein-Barr virus (EBV): 1=A, 2=C, 3=G, 4=T. The data are used in Section 7.9.

Usage

```
data(bnrf1ebv)
```

Format

The format is: Time-Series [1:3954] from 1 to 3954: 1 4 3 3 1 1 3 1 3 1 ...

References

http://www.stat.pitt.edu/stoffer/tsa3/

bnrf1hvs *Nucleotide sequence - BNRF1 of Herpesvirus saimiri*

Description

Nucleotide sequence of the BNRF1 gene of the herpesvirus saimiri (HVS): 1=A, 2=C, 3=G, 4=T. The data are used in Section 7.9.

Usage

```
data(bnrf1hvs)
```

Format

The format is: Time-Series [1:3741] from 1 to 3741: 1 4 3 2 4 4 3 4 4 4 ...

References

http://www.stat.pitt.edu/stoffer/tsa3/

climhyd *Lake Shasta infow data*

Description

Lake Shasta infow data; see Example 7.1. This is a data frame.

Usage

```
data(climhyd)
```

Format

A data frame with 454 observations (rows) on the following 6 numeric variables (columns): `Temp`, `DewPt`, `CldCvr`, `WndSpd`, `Precip`, `Inflow`.

Details

The data are 454 months of measured values for the climatic variables: air temperature, dew point, cloud cover, wind speed, precipitation, and inflow, at Lake Shasta, California. The man-made lake is famous for the placard stating, "We don't swim in your toilet, so don't pee in our lake."

References

http://www.stat.pitt.edu/stoffer/tsa3/

cmort *Cardiovascular Mortality from the LA Pollution study*

Description

Average weekly cardiovascular mortality in Los Angeles County; 508 six-day smoothed averages obtained by filtering daily values over the 10 year period 1970-1979.

Usage

```
data(cmort)
```

Format

The format is: Time-Series [1:508] from 1970 to 1980: 97.8 104.6 94.4 98 95.8 ...

References

http://www.stat.pitt.edu/stoffer/tsa3/

See Also

lap

---

econ5 *Five Quarterly Economic Series*

Description

Data frame containing quarterly U.S. unemployment, GNP, consumption, and government and private investment, from 1948-III to 1988-II.

Usage

```
data(econ5)
```

Format

A data frame with 161 observations (rows) on the following 5 numeric variables (columns): `unemp`, `gnp`, `consum`, `govinv`, `prinv`.

Source

Young, P.C. and Pedregal, D.J. (1999). Macro-economic relativity: government spending, private investment and unemployment in the USA 1948-1998. *Structural Change and Economic Dynamics*, 10, 359-380.

References

http://www.stat.pitt.edu/stoffer/tsa3/

EM0 *EM Algorithm for Time Invariant State Space Models*

Description

Estimation of the parameters in the model (6.1)–(6.2) via the EM algorithm. See Example 6.8 on page 342

Usage

```
EM0(num, y, A, mu0, Sigma0, Phi, cQ, cR, max.iter = 50, tol = 0.01)
```

Arguments

`num`	number of observations
`y`	observation vector or time series
`A`	time-invariant observation matrix
`mu0`	initial state mean vector
`Sigma0`	initial state covariance matrix
`Phi`	state transition matrix
`cQ`	Cholesky-like decomposition of state error covariance matrix Q – see details below
`cR`	Cholesky-like decomposition of state error covariance matrix R – see details below
`max.iter`	maximum number of iterations
`tol`	relative tolerance for determining convergence

Details

Practically, the script only requires that Q or R may be reconstructed as `t(cQ)%*%(cQ)` or `t(cR)%*%(cR)`, respectively.

Value

`Phi`	Estimate of Phi
`Q`	Estimate of Q
`R`	Estimate of R
`mu0`	Estimate of initial state mean
`Sigma0`	Estimate of initial state covariance matrix
`like`	-log likelihood at each iteration
`niter`	number of iterations to convergence
`cvg`	relative tolerance at convergence

Author(s)

D.S. Stoffer

References

http://www.stat.pitt.edu/stoffer/tsa3/

EM1 *EM Algorithm for General State Space Models*

Description

Estimation of the parameters in the model (6.1) – (6.2) via the EM algorithm. For a demonstration, see Example 6.12 on page 357. Inputs are not allowed; see the note.

Usage

```
EM1(num, y, A, mu0, Sigma0, Phi, cQ, cR, max.iter = 100, tol = 0.001)
```

Arguments

`num`	number of observations
`y`	observation vector or time series; use 0 for missing values
`A`	observation matrices, an array with `dim=c(q,p,n)`; use 0 for missing values
`mu0`	initial state mean
`Sigma0`	initial state covariance matrix
`Phi`	state transition matrix
`cQ`	Cholesky-like decomposition of state error covariance matrix Q – see details below
`cR`	R is diagonal here, so `cR = sqrt(R)` – also, see details below
`max.iter`	maximum number of iterations
`tol`	relative tolerance for determining convergence

Details

Practically, the script only requires that Q or R may be reconstructed as `t(cQ)%*%(cQ)` or `t(cR)%*%(cR)`, respectively.

Value

`Phi`	Estimate of Phi
`Q`	Estimate of Q
`R`	Estimate of R
`mu0`	Estimate of initial state mean
`Sigma0`	Estimate of initial state covariance matrix
`like`	-log likelihood at each iteration
`niter`	number of iterations to convergence
`cvg`	relative tolerance at convergence

Note

Inputs are not allowed (and hence not estimated). The script uses Ksmooth1 and everything related to inputs are set equal to zero when it is called. This is a change from the previous version.

It would be relatively easy to include estimates of 'Ups' and 'Gam' because conditional on the states, these are just regression coefficients. If you decide to alter EM1 to include estimates of the 'Ups' or 'Gam', feel free to notify me with a workable example and I'll include it in the next update.

Author(s)

D.S. Stoffer

References

http://www.stat.pitt.edu/stoffer/tsa3/

EQ5 *Seismic Trace of Earthquake number 5*

Description

Seismic trace of an earthquake [two phases or arrivals along the surface, the primary wave ($t = 1, \dots, 1024$) and the shear wave ($t = 1025, \dots, 2048$)] recorded at a seismic station.

Usage

```
data(EQ5)
```

Format

The format is: Time-Series [1:2048] from 1 to 2048: 0.01749 0.01139 0.01512 0.01477 0.00651 ...

References

http://www.stat.pitt.edu/stoffer/tsa3/

See Also

eqexp

eqexp — *Earthquake and Explosion Seismic Series*

Description

This is a data frame of the earthquake and explosion seismic series used throughout the text.

Usage

```
data(eqexp)
```

Format

A data frame with 2048 observations (rows) on 17 variables (columns). Each column is a numeric vector.

Details

The matrix has 17 columns, the first eight are earthquakes, the second eight are explosions, and the last column is the Novaya Zemlya event of unknown origin.

The column names are: `EQ1, EQ2,...,EQ8; EX1, EX2,...,EX8; NZ`. The first 1024 observations correspond to the P wave, the second 1024 observations correspond to the S wave.

References

http://www.stat.pitt.edu/stoffer/tsa3/

EXP6 — *Seismic Trace of Explosion number 6*

Description

Seismic trace of an explosion [two phases or arrivals along the surface, the primary wave ($t = 1, \ldots, 1024$) and the shear wave ($t = 1025, \ldots, 2048$)] recorded at a seismic station.

Usage

```
data(EXP6)
```

Format

The format is: Time-Series [1:2048] from 1 to 2048: -0.001837 -0.000554 -0.002284 -0.000303 -0.000721 ...

References

http://www.stat.pitt.edu/stoffer/tsa3/

See Also

eqexp

FDR *Basic False Discovery Rate*

Description

Computes the basic false discovery rate given a vector of p-values; see Example 7.4 on page 427 for a demonstration.

Usage

```
FDR(pvals, qlevel = 0.05)
```

Arguments

pvals	a vector of pvals on which to conduct the multiple testing
qlevel	the proportion of false positives desired

Value

fdr.id	NULL if no significant tests, or the index of the maximal p-value satisfying the FDR condition.

References

http://www.stat.berkeley.edu/~paciorek/code/fdr/fdr.R

---

flu *Monthly pneumonia and influenza deaths in the U.S., 1968 to 1978.*

Description

Monthly pneumonia and influenza deaths per 10,000 people in the United States for 11 years, 1968 to 1978.

Usage

```
data(flu)
```

Format

The format is: Time-Series [1:132] from 1968 to 1979: 0.811 0.446 0.342 0.277 0.248 ...

References

http://www.stat.pitt.edu/stoffer/tsa3/

`fmri` *fMRI - complete data set*

Description

Data (as a vector list) from an fMRI experiment in pain, listed by location and stimulus. The data are BOLD signals when a stimulus was applied for 32 seconds and then stopped for 32 seconds. The signal period is 64 seconds and the sampling rate was one observation every 2 seconds for 256 seconds ($n = 128$). The number of subjects under each condition varies.

Usage

```
data(fmri)
```

Details

The LOCATIONS of the brain where the signal was measured were [1] Cortex 1: Primary Somatosensory, Contralateral, [2] Cortex 2: Primary Somatosensory, Ipsilateral, [3] Cortex 3: Secondary Somatosensory, Contralateral, [4] Cortex 4: Secondary Somatosensory, Ipsilateral, [5] Caudate, [6] Thalamus 1: Contralateral, [7] Thalamus 2: Ipsilateral, [8] Cerebellum 1: Contralateral and [9] Cerebellum 2: Ipsilateral.

The TREATMENTS or stimuli (and number of subjects in each condition) are [1] Awake-Brush (5 subjects), [2] Awake-Heat (4 subjects), [3] Awake-Shock (5 subjects), [4] Low-Brush (3 subjects), [5] Low-Heat (5 subjects), and [6] Low-Shock (4 subjects). Issue the command `summary(fmri)` for further details. In particular, awake (Awake) or mildly anesthetized (Low) subjects were subjected levels of periodic brushing (Brush), application of heat (Heat), and mild shock (Shock) effects.

As an example, `fmri$L1T6` (Location 1, Treatment 6) will show the data for the four subjects receiving the Low-Shock treatment at the Cortex 1 location; note that `fmri[[6]]` will display the same data.

References

http://www.stat.pitt.edu/stoffer/tsa3/

fmri1 *fMRI Data Used in Chapter 1*

Description

A data frame that consists of average fMRI BOLD signals at eight locations.

Usage

```
data(fmri1)
```

Format

The format is: mts [1:128, 1:9]

Details

Multiple time series consisting of fMRI BOLD signals at eight locations (in columns 2-9, column 1 is time period), when a stimulus was applied for 32 seconds and then stopped for 32 seconds. The signal period is 64 seconds and the sampling rate was one observation every 2 seconds for 256 seconds ($n = 128$). The columns are labeled: "time" "cort1" "cort2" "cort3" "cort4" "thal1" "thal2" "cere1" "cere2".

References

http://www.stat.pitt.edu/stoffer/tsa3/

gas *Gas Prices*

Description

New York Harbor conventional regular gasoline weekly spot price FOB (in cents per gallon) from 2000 to mid-2010.

Usage

```
data(gas)
```

Format

The format is: Time-Series [1:545] from 2000 to 2010: 70.6 71 68.5 65.1 67.9 ...

Details

Pairs with series `oil`

Source

http://tonto.eia.doe.gov/dnav/pet/pet_pri_spt_s1_w.htm

References

http://www.stat.pitt.edu/stoffer/tsa3/

See Also

oil

---

gnp *Quarterly U.S. GNP*

Description

Quarterly U.S. GNP from 1947(1) to 2002(3).

Usage

```
data(gnp)
```

Format

The format is: Time-Series [1:223] from 1947 to 2002: 1489 1497 1500 1524 1547 ...

References

http://www.stat.pitt.edu/stoffer/tsa3/

---

gtemp *Global mean land-ocean temperature deviations*

Description

Global mean land-ocean temperature deviations (from 1951-1980 average), measured in degrees centigrade, for the years 1880-2009.

Usage

```
data(gtemp)
```

Format

The format is: Time-Series [1:130] from 1880 to 2009: -0.28 -0.21 -0.26 -0.27 -0.32 -0.32 -0.29 -0.36 -0.27 -0.17 ...

Source

http://data.giss.nasa.gov/gistemp/graphs/

References

http://www.stat.pitt.edu/stoffer/tsa3/

See Also

gtemp2

---

gtemp2 *Global Mean Surface Air Temperature Deviations*

Description

Similar to gtemp but the data are based only on surface air temperature data obtained from meteorological stations. The data are temperature deviations (from 1951-1980 average), measured in degrees centigrade, for the years 1880-2009.

Usage

```
data(gtemp2)
```

Format

The format is: Time-Series [1:130] from 1880 to 2009: -0.24 -0.19 -0.14 -0.19 -0.45 -0.32 -0.42 -0.54 -0.24 -0.05 ...

Source

http://data.giss.nasa.gov/gistemp/graphs/

References

http://www.stat.pitt.edu/stoffer/tsa3/

See Also

gtemp

`HCT` *Hematocrit Levels*

Description

HCT: Measurements made for 91 days on the three variables, log(white blood count) [WBC], log(platelet) [PLT] and hematocrit [HCT]. Missing data code is 0 (zero).

Usage

```
data(HCT)
```

Format

The format is: Time-Series [1:91] from 1 to 91: 30 30 28.5 34.5 34 32 30.5 31 33 34 ...

Details

See Examples 6.1 and 6.9 for more details.

Source

Jones, R.H. (1984). Fitting multivariate models to unequally spaced data. In *Time Series Analysis of Irregularly Observed Data*, pp. 158-188. E. Parzen, ed. Lecture Notes in Statistics, 25, New York: Springer-Verlag.

References

http://www.stat.pitt.edu/stoffer/tsa3/

See Also

`blood`, `PLT`, `WBC`

`jj` *Johnson and Johnson Quarterly Earnings Per Share*

Description

Johnson and Johnson quarterly earnings per share, 84 quarters (21 years) measured from the first quarter of 1960 to the last quarter of 1980.

Usage

```
data(jj)
```

Format

The format is: Time-Series [1:84] from 1960 to 1981: 0.71 0.63 0.85 0.44 0.61 0.69 0.92 0.55 0.72 0.77 ...

Details

This data set is also included with the R distribution as `JohnsonJohnson`

References

http://www.stat.pitt.edu/stoffer/tsa3/

`Kfilter0` *Kalman Filter - Time Invariant Model*

Description

Returns the filtered values in Property 6.1 on page 326 for the state-space model, (6.1) – (6.2). In addition, returns the evaluation of the likelihood at the given parameter values and the innovation sequence. For demonstrations, see Example 6.5 on page 331,and Example 6.10 on page 350.

Usage

```
Kfilter0(num, y, A, mu0, Sigma0, Phi, cQ, cR)
```

Arguments

`num`	number of observations
`y`	data matrix, vector or time series
`A`	time-invariant observation matrix
`mu0`	initial state mean vector
`Sigma0`	initial state covariance matrix
`Phi`	state transition matrix
`cQ`	Cholesky-type decomposition of state error covariance matrix Q – see details below
`cR`	Cholesky-type decomposition of observation error covariance matrix R – see details below

Details

Practically, the script only requires that Q or R may be reconstructed as `t(cQ)%*%(cQ)` or `t(cR)%*%(cR)`, respectively.

Value

xp	one-step-ahead state prediction
Pp	mean square prediction error
xf	filter value of the state
Pf	mean square filter error
like	the negative of the log likelihood
innov	innovation series
sig	innovation covariances
Kn	last value of the gain, needed for smoothing

Author(s)

D.S. Stoffer

References

http://www.stat.pitt.edu/stoffer/tsa3/

See also http://www.stat.pitt.edu/stoffer/tsa3/chap6.htm for an explanation of the difference between levels 0, 1, and 2.

Kfilter1 *Kalman Filter - Model may be time varying or have inputs*

Description

Returns both the filtered values in Property 6.1 on page 326 for the state-space model, (6.3) – (6.4). For demonstrations, see Example 6.7 on page 338 and Example 6.9 on page 348.

Usage

```
Kfilter1(num, y, A, mu0, Sigma0, Phi, Ups, Gam, cQ, cR, input)
```

Arguments

num	number of observations
y	data matrix, vector or time series
A	time-varying observation matrix, an array with `dim=c(q,p,n)`
mu0	initial state mean
Sigma0	initial state covariance matrix
Phi	state transition matrix
Ups	state input matrix; use `Ups = 0` if not needed

`Gam`	observation input matrix; use `Gam = 0` if not needed
`cQ`	Cholesky-type decomposition of state error covariance matrix Q – see details below
`cR`	Cholesky-type decomposition of observation error covariance matrix R – see details below
`input`	matrix or vector of inputs having the same row dimension as y; use `input = 0` if not needed

Details

Practically, the script only requires that Q or R may be reconstructed as `t(cQ)%*%(cQ)` or `t(cR)%*%(cR)`, respectively.

Value

`xp`	one-step-ahead prediction of the state
`Pp`	mean square prediction error
`xf`	filter value of the state
`Pf`	mean square filter error
`like`	the negative of the log likelihood
`innov`	innovation series
`sig`	innovation covariances
`Kn`	last value of the gain, needed for smoothing

Author(s)

D.S. Stoffer

References

http://www.stat.pitt.edu/stoffer/tsa3/

See also http://www.stat.pitt.edu/stoffer/tsa3/chap6.htm for an explanation of the difference between levels 0, 1, and 2.

`Kfilter2` *Kalman Filter - Model may be time varying or have inputs or correlated errors*

Description

Returns the filtered values in Property 6.5 on page 354 for the state-space model, (6.97) – (6.99). In addition, the script returns the evaluation of the likelihood at the given parameter values and the innovation sequence. For demonstrations, see Example 6.11 on page 356 and Example 6.13 on page 361.

Usage

```
Kfilter2(num, y, A, mu0, Sigma0, Phi, Ups, Gam, Theta, cQ, cR,
          S, input)
```

Arguments

`num`	number of observations
`y`	data matrix, vector or time series
`A`	time-varying observation matrix, an array with `dim = c(q,p,n)`
`mu0`	initial state mean
`Sigma0`	initial state covariance matrix
`Phi`	state transition matrix
`Ups`	state input matrix; use `Ups = 0` if not needed
`Gam`	observation input matrix; use `Gam = 0` if not needed
`Theta`	state error pre-matrix
`cQ`	Cholesky decomposition of state error covariance matrix Q – see details below
`cR`	Cholesky-type decomposition of observation error covariance matrix R – see details below
`S`	covariance-type matrix of state and observation errors
`input`	matrix or vector of inputs having the same row dimension as y; use `input = 0` if not needed

Details

Practically, the script only requires that Q or R may be reconstructed as `t(cQ)%*%(cQ)` or `t(cR)%*%(cR)`, respectively.

Value

`xp`	one-step-ahead prediction of the state
`Pp`	mean square prediction error
`xf`	filter value of the state
`Pf`	mean square filter error
`like`	the negative of the log likelihood
`innov`	innovation series
`sig`	innovation covariances
`K`	last value of the gain, needed for smoothing

Author(s)

D.S. Stoffer

References

http://www.stat.pitt.edu/stoffer/tsa3/

See also http://www.stat.pitt.edu/stoffer/tsa3/chap6.htm for an explanation of the difference between levels 0, 1, and 2.

Ksmooth0 *Kalman Filter and Smoother - Time invariant model without inputs*

Description

Returns both the filtered values in Property 6.1 on page 326 and the smoothed values in Property 6.2 on page 330 for the state-space model, (6.1) – (6.2). For demonstrations, see Example 6.5 on page 331, and Example 6.10 on page 350.

Usage

```
Ksmooth0(num, y, A, mu0, Sigma0, Phi, cQ, cR)
```

Arguments

num	number of observations
y	data matrix, vector or time series
A	time-invariant observation matrix
mu0	initial state mean vector
Sigma0	initial state covariance matrix
Phi	state transition matrix
cQ	Cholesky-type decomposition of state error covariance matrix Q – see details below
cR	Cholesky-type decomposition of observation error covariance matrix R – see details below

Details

Practically, the script only requires that Q or R may be reconstructed as `t(cQ)%*%(cQ)` or `t(cR)%*%(cR)`, respectively, which allows more flexibility.

Value

xs	state smoothers
Ps	smoother mean square error
x0n	initial mean smoother
P0n	initial smoother covariance

`J0`	initial value of the J matrix
`J`	the J matrices
`xp`	one-step-ahead prediction of the state
`Pp`	mean square prediction error
`xf`	filter value of the state
`Pf`	mean square filter error
`like`	the negative of the log likelihood
`Kn`	last value of the gain

Author(s)

D.S. Stoffer

References

http://www.stat.pitt.edu/stoffer/tsa3/

See also http://www.stat.pitt.edu/stoffer/tsa3/chap6.htm for an explanation of the difference between levels 0, 1, and 2.

`Ksmooth1` *Kalman Filter and Smoother - General model*

Description

Returns both the filtered values in Property 6.1 on page 326 and the smoothed values in Property 6.2 on page 330 for the state-space model, (6.3) – (6.4). For demonstrations, see Example 6.7 on page 338 and Example 6.9 on page 348.

Usage

```
Ksmooth1(num, y, A, mu0, Sigma0, Phi, Ups, Gam, cQ, cR, input)
```

Arguments

`num`	number of observations
`y`	data matrix, vector or time series
`A`	time-varying observation matrix, an array with `dim=c(q,p,n)`
`mu0`	initial state mean
`Sigma0`	initial state covariance matrix
`Phi`	state transition matrix
`Ups`	state input matrix; use `Ups = 0` if not needed
`Gam`	observation input matrix; use `Gam = 0` if not needed

`cQ`	Cholesky-type decomposition of state error covariance matrix Q – see details below
`cR`	Cholesky-type decomposition of observation error covariance matrix R – see details below
`input`	matrix or vector of inputs having the same row dimension as y; use `input = 0` if not needed

Details

Practically, the script only requires that Q or R may be reconstructed as `t(cQ)%*%(cQ)` or `t(cR)%*%(cR)`, respectively, which allows more flexibility.

Value

`xs`	state smoothers
`Ps`	smoother mean square error
`x0n`	initial mean smoother
`P0n`	initial smoother covariance
`J0`	initial value of the J matrix
`J`	the J matrices
`xp`	one-step-ahead prediction of the state
`Pp`	mean square prediction error
`xf`	filter value of the state
`Pf`	mean square filter error
`like`	the negative of the log likelihood
`Kn`	last value of the gain

Author(s)

D.S. Stoffer

References

http://www.stat.pitt.edu/stoffer/tsa3/

See also http://www.stat.pitt.edu/stoffer/tsa3/chap6.htm for an explanation of the difference between levels 0, 1, and 2.

`Ksmooth2` *Kalman Filter and Smoother - General model, may have correlated errors*

Description

Returns the filtered and smoothed values in Property 6.5 on page 354 for the state-space model, (6.97) – (6.99). This is the smoother companion to Kfilter2.

Usage

```
Ksmooth2(num, y, A, mu0, Sigma0, Phi, Ups, Gam, Theta, cQ, cR,
          S, input)
```

Arguments

`num`	number of observations
`y`	data matrix, vector or time series
`A`	time-varying observation matrix, an array with `dim=c(q,p,n)`
`mu0`	initial state mean
`Sigma0`	initial state covariance matrix
`Phi`	state transition matrix
`Ups`	state input matrix; use `Ups = 0` if not needed
`Gam`	observation input matrix; use `Gam = 0` if not needed
`Theta`	state error pre-matrix
`cQ`	Cholesky-type decomposition of state error covariance matrix Q – see details below
`cR`	Cholesky-type decomposition of observation error covariance matrix R – see details below
`S`	covariance matrix of state and observation errors
`input`	matrix or vector of inputs having the same row dimension as y; use `input = 0` if not needed

Details

Practically, the script only requires that Q or R may be reconstructed as `t(cQ)%*%(cQ)` or `t(cR)%*%(cR)`, respectively, which allows more flexibility.

Value

`xs`	state smoothers
`Ps`	smoother mean square error
`J`	the J matrices
`xp`	one-step-ahead prediction of the state
`Pp`	mean square prediction error
`xf`	filter value of the state
`Pf`	mean square filter error
`like`	the negative of the log likelihood
`Kn`	last value of the gain

Author(s)

D.S. Stoffer

References

http://www.stat.pitt.edu/stoffer/tsa3/

See also http://www.stat.pitt.edu/stoffer/tsa3/chap6.htm for an explanation of the difference between levels 0, 1, and 2.

`lag1.plot` *Lag Plot - one time series*

Description

Produces a grid of scatterplots of a series versus lagged values of the series.

Usage

```
lag1.plot(series, max.lag = 1, corr = TRUE, smooth = TRUE)
```

Arguments

`series`	the data
`max.lag`	maximum lag
`corr`	if TRUE, shows the autocorrelation value in a legend
`smooth`	if TRUE, adds a lowess fit to each scatterplot

Note

In `tsa3` this is called `lag.plot1`.

Author(s)

D.S. Stoffer

References

http://www.stat.pitt.edu/stoffer/tsa3/

Examples

```
data(soi)
lag1.plot(soi, 9)
```

`lag2.plot` *Lag Plot - two time series*

Description

Produces a grid of scatterplots of one series versus another. The first named series is the one that gets lagged.

Usage

```
lag2.plot(series1, series2, max.lag = 0, corr = TRUE, smooth = TRUE)
```

Arguments

`series1`	first series (the one that gets lagged)
`series2`	second series
`max.lag`	maximum number of lags
`corr`	if TRUE, shows the cross-correlation value in a legend
`smooth`	if TRUE, adds a lowess fit to each scatterplot

Note

In `tsa3` this is called `lag.plot2`.

Author(s)

D.S. Stoffer

References

http://www.stat.pitt.edu/stoffer/tsa3/

Examples

```
data(soi, rec)
lag2.plot(soi, rec, 8)
```

LagReg *Lagged Regression*

Description

Performs lagged regression as discussed in Chapter 4, Section 4.10. See Example 4.24 on page 244 for a demonstration.

Usage

```
LagReg(input, output, L = c(3, 3), M = 40, threshold = 0,
        inverse = FALSE)
```

Arguments

`input`	input series
`output`	output series
`L`	degree of smoothing; see `spans` in the help file for `spec.pgram`.
`M`	must be even; number of terms used in the lagged regression
`threshold`	the cut-off used to set small (in absolute value) regression coeffcients equal to zero
`inverse`	if TRUE, will fit a forward-lagged regression

Details

For a bivariate series, `input` is the input series and `output` is the output series. The degree of smoothing for the spectral estimate is given by L; see `spans` in the help file for `spec.pgram`. The number of terms used in the lagged regression approximation is given by M, which must be even. The threshold value is the cut-off used to set small (in absolute value) regression coeffcients equal to zero (it is easiest to run LagReg twice, once with the default threshold of zero, and then again after inspecting the resulting coeffcients and the corresponding values of the CCF). Setting inverse=TRUE will fit a forward-lagged regression; the default is to run a backward-lagged regression. The script is based on code that was contributed by Professor Doug Wiens, Department of Mathematical and Statistical Sciences, University of Alberta.

Value

The estimated impulse response function; see Example 4.24.

Author(s)

D.S. Stoffer

References

http://www.stat.pitt.edu/stoffer/tsa3/

`lap` *LA Pollution-Mortality Study*

Description

LA Pollution-Mortality Study (1970-1979, weekly data).

Usage

```
data(lap)
```

Format

The format is: mts [1:508, 1:11]

Details

columns are time series	with names
(1) Total Mortality	`tmort`
(2) Respiratory Mortality	`rmort`
(3) Cardiovascular Mortality	`cmort`
(4) Temperature	`tempr`
(5) Relative Humidity	`rh`
(6) Carbon Monoxide	`co`
(7) Sulfur Dioxide	`so2`
(8) Nitrogen Dioxide	`no2`
(9) Hydrocarbons	`hycarb`
(10) Ozone	`o3`
(11) Particulates	`part`

References

http://www.stat.pitt.edu/stoffer/tsa3/

`lead` *Leading Indicator*

Description

Leading indicator, 150 months; taken from Box and Jenkins (1970).

Usage

```
data(lead)
```

Format

The format is: Time-Series [1:150] from 1 to 150: 10.01 10.07 10.32 9.75 10.33 ...

Details

This is also the R time series `BJsales.lead`: The sales time series `BJsales` and leading indicator `BJsales.lead` each contain 150 observations. The objects are of class "ts".

See Also

sales

---

mvspec *Univariate and Multivariate Spectral Estimation*

Description

This is `spec.pgram` with a few changes in the defaults and written so you can easily extract the estimate of the multivariate spectral matrix as `fxx`. The bandwidth calculation has been changed to the more practical definition given in the text. Can be used to replace `spec.pgram` for univariate series.

Usage

```
mvspec(x, spans = NULL, kernel = NULL, taper = 0, pad = 0,
         fast = TRUE, demean = FALSE, detrend = TRUE,
         plot = TRUE, na.action = na.fail, ...)
```

Arguments

`x`	univariate or multivariate time series (i.e., the p columns of x are time series)
`spans`	specify smoothing; same as `spec.pgram`
`kernel`	specify kernel; same as `spec.pgram`
`taper`	specify taper; same as `spec.pgram` with different default
`pad`	specify padding; same as `spec.pgram`
`fast`	specify use of FFT; same as `spec.pgram`
`demean`	if TRUE, series is demeaned first; same as `spec.pgram`
`detrend`	if TRUE, series is detrended first; same as `spec.pgram`
`plot`	plot the estimate; same as `spec.pgram`
`na.action`	same as `spec.pgram`
`...`	additional arguments; same as `spec.pgram`

Details

This is `spec.pgram` with a few changes in the defaults and written so you can easily extract the estimate of the multivariate spectral matrix as `fxx`. The bandwidth calculation has been changed to the more practical definition given in the text, $(L_h/n.used) * frequency(x)$. Although meant to be used to easily obtain multivariate spectral estimates, this script can be used in Chapter 4 for the spectral analysis of a univariate time series. Note that the script does not taper by default (`taper=0`); this forces the user to do "conscious tapering". See Example 7.12 on page 461 for a demonstration.

Value

An object of class "spec", which is a list containing at least the following components:

`fxx`	spectral matrix estimates; an array of dimensions `dim = c(p,p,nfreq)`
`freq`	vector of frequencies at which the spectral density is estimated.
`spec`	vector (for univariate series) or matrix (for multivariate series) of estimates of the spectral density at frequencies corresponding to freq.
`coh`	NULL for univariate series. For multivariate time series, a matrix containing the squared coherency between different series. Column i + (j - 1) * (j - 2)/2 of coh contains the squared coherency between columns i and j of x, where i < j.
`phase`	NULL for univariate series. For multivariate time series a matrix containing the cross-spectrum phase between different series. The format is the same as coh.
`Lh`	Number of frequencies (approximate) used in the band, as defined in (4.57) on page 204.
`n.used`	Sample length used for the FFT
`series`	The name of the time series.
`snames`	For multivariate input, the names of the component series.
`method`	The method used to calculate the spectrum.

The results are returned invisibly if plot is true.

References

http://www.stat.pitt.edu/stoffer/tsa3/

Examples

```
# univariate example
plot(co2)   # co2 is an R data set
mvspec(co2, spans=c(5,5), taper=.5)

# multivariate example
ts.plot(mdeaths, fdeaths, col=1:2)   # an R data set, male/female monthly deaths ...
dog = mvspec(cbind(mdeaths,fdeaths), spans=c(3,3), taper=.1)
dog$fxx        # look a spectral matrix estimates
dog$bandwidth  # bandwidth with time unit = year
dog$bandwidth/frequency(mdeaths)  # ... with time unit = month
plot(dog, plot.type="coherency")  # plot of squared coherency
```

`nyse` *Returns of the New York Stock Exchange*

Description

Returns of the New York Stock Exchange (NYSE) from February 2, 1984 to December 31, 1991.

Usage

```
data(nyse)
```

Format

The format is: Time-Series [1:2000] from 1 to 2000: 0.00335 -0.01418 -0.01673 0.00229 -0.01692 ...

Source

S+GARCH module - Version 1.1 Release 2: 1998

`oil` *Crude oil, WTI spot price FOB*

Description

Crude oil, WTI spot price FOB (in dollars per barrel), weekly data from 2000 to mid-2010.

Usage

```
data(oil)
```

Format

The format is: Time-Series [1:545] from 2000 to 2010: 26.2 26.1 26.3 24.9 26.3 ...

Details

pairs with the series gas

Source

http://tonto.eia.doe.gov/dnav/pet/pet_pri_spt_s1_w.htm

References

http://www.stat.pitt.edu/stoffer/tsa3/

See Also

`gas`

`part` *Particulate levels from the LA pollution study*

Description

Particulate series corresponding to `cmort` from the LA pollution study.

Usage

```
data(part)
```

Format

The format is: Time-Series [1:508] from 1970 to 1980: 72.7 49.6 55.7 55.2 66 ...

See Also

`lap`

`PLT` *Platelet Levels*

Description

PLT: Measurements made for 91 days on the three variables, log(white blood count) [WBC], log(platelet) [PLT] and hematocrit [HCT]. Missing data code is 0 (zero).

Usage

```
data(PLT)
```

Format

The format is: Time-Series [1:91] from 1 to 91: 4.47 4.33 4.09 4.6 4.41 ...

Details

See Examples 6.1 and 6.9 for more details.

Source

Jones, R.H. (1984). Fitting multivariate models to unequally spaced data. In *Time Series Analysis of Irregularly Observed Data*, pp. 158-188. E. Parzen, ed. Lecture Notes in Statistics, 25, New York: Springer-Verlag.

References

http://www.stat.pitt.edu/stoffer/tsa3/

See Also

blood, HCT, WBC

prodn *Monthly Federal Reserve Board Production Index*

Description

Monthly Federal Reserve Board Production Index (1948-1978, n = 372 months).

Usage

```
data(prodn)
```

Format

The format is: Time-Series [1:372] from 1948 to 1979: 40.6 41.1 40.5 40.1 40.4 41.2 39.3 41.6 42.3 43.2 ...

References

http://www.stat.pitt.edu/stoffer/tsa3/

qinfl *Quarterly Inflation*

Description

Quarterly inflation rate in the Consumer Price Index from 1953-Ito 1980-II, n = 110 observations.

Usage

```
data(qinfl)
```

Format

The format is: Time-Series [1:110] from 1953 to 1980: 1.673 3.173 0.492 -0.327 -0.333 ...

Details

pairs with qintr (interest rate)

Source

Newbold, P. and T. Bos (1985). *Stochastic Parameter Regression Models*. Beverly Hills: Sage.

References

http://www.stat.pitt.edu/stoffer/tsa3/

See Also

qintr

---

qintr *Quarterly Interest Rate*

Description

Quarterly interest rate recorded for Treasury bills from 1953-Ito 1980-II, n = 110 observations.

Usage

```
data(qintr)
```

Format

The format is: Time-Series [1:110] from 1953 to 1980: 1.98 2.15 1.96 1.47 1.06 ...

Details

pairs with qinfl (inflation)

Source

Newbold, P. and T. Bos (1985). *Stochastic Parameter Regression Models*. Beverly Hills: Sage.

References

See http://www.stat.pitt.edu/stoffer/tsa3/

See Also

qinfl

rec *Recruitment (number of new fish)*

Description

Recruitment (number of new fish) for a period of 453 months ranging over the years 1950-1987.

Usage

```
data(rec)
```

Format

The format is: Time-Series [1:453] from 1950 to 1988: 68.6 68.6 68.6 68.6 68.6 ...

Details

can pair with `soi` (Southern Oscillation Index)

Source

Data furnished by Dr. Roy Mendelssohn of the Pacific Fisheries Environmental Laboratory, NOAA (personal communication).

References

http://www.stat.pitt.edu/stoffer/tsa3/

See Also

soi

sales *Sales*

Description

Sales, 150 months; taken from Box and Jenkins (1970).

Usage

```
data(sales)
```

Format

The format is: Time-Series [1:150] from 1 to 150: 200 200 199 199 199 ...

Details

This is also the R data set `BJsales`: The sales time series `BJsales` and leading indicator `BJsales.lead` each contain 150 observations. The objects are of class "ts".

See Also

`lead`

---

salt *Salt Profiles*

Description

Salt profiles taken over a spatial grid set out on an agricultural field, 64 rows at 17-ft spacing.

Usage

```
data(salt)
```

Format

The format is: Time-Series [1:64] from 1 to 64: 6 6 6 3 3 3 4 4 4 1.5 ...

Details

pairs with `saltemp`, temperature profiles on the same grid

References

http://www.stat.pitt.edu/stoffer/tsa3/

See Also

`saltemp`

saltemp *Temperature Profiles*

Description

Temperature profiles over a spatial grid set out on an agricultural field, 64 rows at 17-ft spacing.

Usage

```
data(saltemp)
```

Format

The format is: Time-Series [1:64] from 1 to 64: 5.98 6.54 6.78 6.34 6.96 6.51 6.72 7.44 7.74 6.85 ...

Details

pairs with `salt`, salt profiles on the same grid

References

http://www.stat.pitt.edu/stoffer/tsa3/

See Also

`salt`

sarima *Fit ARIMA Models*

Description

Fits ARIMA models (including improved diagnostics) in a short command. It can also be used to perform regression with autocorrelated errors. This is a front end to R's `arima()` with a different back door.

Usage

```
sarima(xdata, p, d, q, P = 0, D = 0, Q = 0, S = -1,
       details = TRUE, xreg=NULL,
       tol = sqrt(.Machine$double.eps),
       no.constant = FALSE)
```

Arguments

`xdata`	univariate time series
`p`	AR order
`d`	difference order
`q`	MA order
`P`	SAR order; use only for seasonal models
`D`	seasonal difference; use only for seasonal models
`Q`	SMA order; use only for seasonal models
`S`	seasonal period; use only for seasonal models
`xreg`	Optionally, a vector or matrix of external regressors, which must have the same number of rows as xdata.
`details`	turns on or off the output from the nonlinear optimization routine, which is `optim`. The default is TRUE, use `details=FALSE` to turn off the output.
`tol`	controls the relative tolerance (reltol in `optim`) used to assess convergence. The default is `sqrt(.Machine$double.eps)`, the R default.
`no.constant`	controls whether or not sarima includes a constant in the model. In particular, if there is no differencing (d = 0 and D = 0) you get the mean estimate. If there is differencing of order one (either d = 1 or D = 1, but not both), a constant term is included in the model. These two conditions may be overridden (i.e., no constant will be included in the model) by setting this to TRUE; e.g., `sarima(x,1,1,0,no.constant=TRUE)`. Otherwise, no constant or mean term is included in the model. If regressors are included (via `xreg`), this is ignored.

Details

If your time series is in x and you want to fit an ARIMA(p,d,q) model to the data, the basic call is `sarima(x,p,d,q)`. The results are the parameter estimates, standard errors, AIC, AICc, BIC (as defined in Chapter 2) and diagnostics. To fit a seasonal ARIMA model, the basic call is `sarima(x,p,d,q,P,D,Q,S)`. For example, `sarima(x,2,1,0)` will fit an ARIMA(2,1,0) model to the series in x, and `sarima(x,2,1,0,0,1,1,12)` will fit a seasonal ARIMA$(2,1,0) * (0,1,1)_{12}$ model to the series in x.

Value

`fit`	the `arima` object
`AIC`	value of the AIC
`AICc`	value of the AICc
`BIC`	value of the BIC

References

http://www.stat.pitt.edu/stoffer/tsa3/

See Also

sarima.for

Examples

```
sarima(log(AirPassengers),0,1,1,0,1,1,12)
(dog <- sarima(log(AirPassengers),0,1,1,0,1,1,12))
summary(dog$fit)  # fit has all the returned arima() values
plot(resid(dog$fit))  # plot the innovations (residuals)
```

---

sarima.for *ARIMA Forecasting*

Description

ARIMA forecasting - this is a front end to R's `predict.Arima`.

Usage

```
sarima.for(xdata, n.ahead, p, d, q, P = 0, D = 0, Q = 0, S = -1,
           tol = sqrt(.Machine$double.eps), no.constant = FALSE)
```

Arguments

`xdata`	univariate time series
`n.ahead`	forecast horizon (number of periods)
`p`	AR order
`d`	difference order
`q`	MA order
`P`	SAR order; use only for seasonal models
`D`	seasonal difference; use only for seasonal models
`Q`	SMA order; use only for seasonal models
`S`	seasonal period; use only for seasonal models
`tol`	controls the relative tolerance (reltol) used to assess convergence. The default is `sqrt(.Machine$double.eps)`, the R default.
`no.constant`	controls whether or not a constant is included in the model. If `no.constant=TRUE`, no constant is included in the model. See `sarima` for more details.

Details

For example, `sarima.for(x,5,1,0,1)` will forecast five time points ahead for an ARMA(1,1) fit to x. The output prints the forecasts and the standard errors of the forecasts, and supplies a graphic of the forecast with +/- 2 prediction error bounds.

Value

`pred`	the forecasts
`se`	the prediction (standard) errors

References

http://www.stat.pitt.edu/stoffer/tsa3/

See Also

sarima

Examples

```
sarima.for(log(AirPassengers),12,0,1,1,0,1,1,12)
```

`SigExtract` *Signal Extraction And Optimal Filtering*

Description

Performs signal extraction and optimal filtering as discussed in Chapter 4, Section 4.11. See Example 4.25 on page 249 for a demonstration.

Usage

```
SigExtract(series, L = c(3, 3), M = 50, max.freq = 0.05)
```

Arguments

`series`	univariate time series to be filtered
`L`	degree of smoothing (may be a vector); see `spans` in `spec.pgram` for more details
`M`	number of terms used in the lagged regression approximation
`max.freq`	truncation frequency, which must be larger than 1/M.

Details

The basic function of the script, and the default setting, is to remove frequencies above 1/20 (and, in particular, the seasonal frequency of 1 cycle every 12 time points). The sampling frequency of the time series is set to unity prior to the analysis.

Value

Returns plots of (1) the original and filtered series, (2) the estiamted spectra of each series, (3) the filter coefficients and the desired and attained frequency response function. The filtered series is returned invisibly.

Note

The script is based on code that was contributed by Professor Doug Wiens, Department of Mathematical and Statistical Sciences, University of Alberta.

Author(s)

D.S. Stoffer

References

http://www.stat.pitt.edu/stoffer/tsa3/

`so2` *SO2 levels from the LA pollution study*

Description

Sulfur dioxide levels from the LA pollution study

Usage

```
data(so2)
```

Format

The format is: Time-Series [1:508] from 1970 to 1980: 3.37 2.59 3.29 3.04 3.39 2.57 2.35 3.38 1.5 2.56 ...

See Also

`lap`

`soi` *Southern Oscillation Index*

Description

Southern Oscillation Index (SOI) for a period of 453 months ranging over the years 1950-1987.

Usage

```
data(soi)
```

Format

The format is: Time-Series [1:453] from 1950 to 1988: 0.377 0.246 0.311 0.104 -0.016 0.235 0.137 0.191 -0.016 0.29 ...

Details

pairs with `rec` (Recruitment)

Source

Data furnished by Dr. Roy Mendelssohn of the Pacific Fisheries Environmental Laboratory, NOAA (personal communication).

References

http://www.stat.pitt.edu/stoffer/tsa3/

See Also

`rec`

`soiltemp` *Spatial Grid of Surface Soil Temperatures*

Description

A 64 by 36 matrix of surface soil temperatures.

Usage

```
data(soiltemp)
```

Format

The format is: num [1:64, 1:36] 6.7 8.9 5 6.6 6.1 7 6.5 8.2 6.7 6.6 ...

References

http://www.stat.pitt.edu/stoffer/tsa3/

speech *Speech Recording*

Description

A small .1 second (1000 points) sample of recorded speech for the phrase "aaa...hhh".

Usage

```
data(speech)
```

Format

The format is: Time-Series [1:1020] from 1 to 1020: 1814 1556 1442 1416 1352 ...

References

http://www.stat.pitt.edu/stoffer/tsa3/

star *Variable Star*

Description

The magnitude of a star taken at midnight for 600 consecutive days. The data are taken from the classic text, The Calculus of Observations, a Treatise on Numerical Mathematics, by E.T. Whittaker and G. Robinson, (1923, Blackie and Son, Ltd.).

Usage

```
data(star)
```

Format

The format is: Time-Series [1:600] from 1 to 600: 25 28 31 32 33 33 32 ...

References

http://www.stat.pitt.edu/stoffer/tsa3/

`stoch.reg` *Frequency Domain Stochastic Regression*

Description

Performs frequency domain stochastic regression discussed in Chapter 7, Section 7.3.

Usage

```
stoch.reg(data, cols.full, cols.red, alpha, L, M, plot.which)
```

Arguments

`data`	data matrix
`cols.full`	specify columns of data matrix that are in the full model
`cols.red`	specify columns of data matrix that are in the reduced model (use NULL if there are no inputs in the reduced model)
`alpha`	test size
`L`	smoothing - see `spans` in `spec.pgram`
`M`	number of points in the discretization of the integral
`plot.which`	`coh` or `F.stat`, to plot either the squared-coherencies or the F-statistics, respectively

Value

`power.full`	spectrum under the full model
`power.red`	spectrum under the reduced model
`Betahat`	regression parameter estimates
`eF`	pointwise (by frequency) F-tests
`coh`	coherency

Note

The script is based on code that was contributed by Professor Doug Wiens, Department of Mathematical and Statistical Sciences, University of Alberta. See Example 7.1 on page 417 for a demonstration.

Author(s)

D.S. Stoffer

References

http://www.stat.pitt.edu/stoffer/tsa3/

sunspotz *Biannual Sunspot Numbers*

Description

Biannual smoothed (12-month moving average) number of sunspots from June 1749 to December 1978; n = 459. The "z" on the end is to distinguish this series from the one included with R (called `sunspots`).

Usage

```
data(sunspotz)
```

Format

The format is: Time Series: Start = c(1749, 1) End = c(1978, 1) Frequency = 2

References

http://www.stat.pitt.edu/stoffer/tsa3/

SVfilter *Switching Filter (for Stochastic Volatility Models)*

Description

Performs a special case switching filter when the observational noise is a certain mixture of normals. Used to fit a stochastic volatility model. See Example 6.18 page 380 and Example 6.19 page 383.

Usage

```
SVfilter(num, y, phi0, phi1, sQ, alpha, sR0, mu1, sR1)
```

Arguments

`num`	number of observations
`y`	time series of returns
`phi0`	state constant
`phi1`	state transition parameter
`sQ`	state standard deviation
`alpha`	observation constant
`sR0`	observation error standard deviation for mixture component zero
`mu1`	observation error mean for mixture component one
`sR1`	observation error standard deviation for mixture component one

Value

xp	one-step-ahead prediction of the volatility
Pp	mean square prediction error of the volatility
like	the negative of the log likelihood at the given parameter values

Author(s)

D.S. Stoffer

References

http://www.stat.pitt.edu/stoffer/tsa3/

---

tempr — *Temperatures from the LA pollution study*

Description

Temperature series corresponding to `cmort` from the LA pollution study.

Usage

```
data(tempr)
```

Format

The format is: Time-Series [1:508] from 1970 to 1980: 72.4 67.2 62.9 72.5 74.2 ...

See Also

`lap`

---

unemp — *U.S. Unemployment*

Description

Monthly U.S. Unemployment series (1948-1978, n = 372)

Usage

```
data(unemp)
```

Format

The format is: Time-Series [1:372] from 1948 to 1979: 235 281 265 241 201 ...

References

http://www.stat.pitt.edu/stoffer/tsa3/

---

varve *Annual Varve Series*

Description

Sedimentary deposits from one location in Massachusetts for 634 years, beginning nearly 12,000 years ago.

Usage

```
data(varve)
```

Format

The format is: Time-Series [1:634] from 1 to 634: 26.3 27.4 42.3 58.3 20.6 ...

References

http://www.stat.pitt.edu/stoffer/tsa3/

WBC *White Blood Cell Levels*

Description

WBC: Measurements made for 91 days on the three variables, log(white blood count) [WBC], log(platelet) [PLT] and hematocrit [HCT]. Missing data code is 0 (zero).

Usage

```
data(WBC)
```

Format

The format is: Time-Series [1:91] from 1 to 91: 2.33 1.89 2.08 1.82 1.82 ...

Details

See Examples 6.1 amd 6.9 for more details.

Source

Jones, R.H. (1984). Fitting multivariate models to unequally spaced data. In *Time Series Analysis of Irregularly Observed Data*, pp. 158-188. E. Parzen, ed. Lecture Notes in Statistics, 25, New York: Springer-Verlag.

References

http://www.stat.pitt.edu/stoffer/tsa3/

See Also

blood, HCT, PLT

Index