Package: multitaper (via r-universe)

October 28, 2026

Version 1.0-17

Title Spectral Analysis Tools using the Multitaper Method

Author Karim Rahim <karim.rahim@queensu.ca>, Wesley S. Burr <wesleyburr@trentu.ca>

Maintainer Karim Rahim <karim.rahim@queensu.ca>

Depends R (>= 3.0), methods

Suggests psd, fftwtools, slp

Description Implements multitaper spectral analysis using discrete prolate spheroidal sequences (Slepians) and sine tapers. It includes an adaptive weighted multitaper spectral estimate, a coherence estimate, Thomson's Harmonic F-test, and complex demodulation. The Slepians sequences are generated efficiently using a tridiagonal matrix solution, and jackknifed confidence intervals are available for most estimates. This package is an implementation of the method described in D.J. Thomson (1982) ``Spectrum estimation and harmonic analysis'' <doi:10.1109/PROC.1982.12433>.

License GPL (>= 2)

ByteCompile true

LazyData true

URL https://github.com/krahim/multitaper/

NeedsCompilation yes

Repository https://cran.r-universe.dev

Date/Publication 2023-07-20 12:49:50 UTC

RemoteUrl https://github.com/cran/multitaper

RemoteRef HEAD

RemoteSha 493d7683d7e1f3d5eefdad76aabc2a7b9cb55a85

Contents

centre . . . 2
CETdaily . . . 3
CETmonthly . . . 3
demod.dpss . . . 4
dpss . . . 5
dpssToEigenvalues . . . 6
dropFreqs . . . 7
HadCRUTnh . . . 8
mlco2 . . . 8
mtm.coh . . . 9
multitaperTrend . . . 10
percivalAR4 . . . 10
plot.mtm . . . 11
plot.mtm.coh . . . 12
sineTaper . . . 13
spec.mtm . . . 14
willamette . . . 16

Index **18**

`centre` *Centres (converts to zero-mean) the time series.*

Description

Centres the data using an expansion on the Slepian sequences if the bandwidth parameter (nw) and number of tapers (k) is specified, otherwise subtracts the mean or robust trimmed mean.

Usage

```
centre(x, nw = NULL, k = NULL, deltaT = NULL, trim = 0)
```

Arguments

`x`	The data as a vector or as a time series.
`nw`	The Slepian bandwidth parameter, typically between 2.0 and 6.0.
`k`	The number of Slepian tapers used, often 2*nw.
`deltaT`	Parameter required if the data is a vector and not a time series, and only for the Slepian case.
`trim`	[only used if nw and k are not specified] The fraction (0 to 0.5) of observations to be trimmed from each end of 'x' before the mean is computed. Values of trim outside that range are taken as the nearest endpoint.

References

Thomson, D.J (1982) Spectrum estimation and harmonic analysis. *Proceedings of the IEEE* Volume **70**, number 9, pp. 1055–1096.

Slepian, D. (1978) Prolate spheroidal wave functions, Fourier analysis, and uncertainty. V–The discrete case. *Bell System Technical Journal* Volume **57**, pp. 1371–1430.

Examples

```
data(willamette)
cent.Slepian <- centre(willamette, nw=4, k=8, deltaT=1)
cent.Trim <- centre(willamette, trim=0.2)
```

CETdaily *Central England Temperature daily time series*

Description

Central England Temperature daily time series from the Hadley Centre United Kingdom Meteorological Office, http://www.metoffice.gov.uk/hadobs/hadcet/. The data set represents daily CET values recorded to the tenth of a degree Celsius, and contains temperatures for the years 1772 through September 30, 2011. This dataset was retrieved from the Met office at Hadley on Oct 1, 2011.

Usage

```
CETdaily
```

Format

A data frame indicating the year, month, day, and temperature observed in Central England. This data-set contains 87566 observations.

CETmonthly *Central England Temperature monthly time series*

Description

Central England Temperature monthly time series from the Hadley Centre United Kingdom Meteorological Office, http://www.metoffice.gov.uk/hadobs/hadcet/. The data set represents monthly CET temperature values recorded to the nearest tenth of a degree Celsius, and containing records for January 1659 through December 2011. This dataset was retrieved from the Met office at Hadley on Mar 10, 2012.

Usage

```
CETmonthly
```

Format

A data frame indicating the year, month and temperature observed in Central England. This data-set contains 4237 observations.

References

Parker DE, Horton EB (2005). Uncertainties in central England temperature 1878-2003 and some improvements to the maximum and minimum series. *Int. J. Climatol*. 25: 1173–1188.

`demod.dpss` *Computes complex demodulates using multiple taper techniques*

Description

Computes complex demodulate of a given series around a given central frequency using multiple taper techniques. Returns amplitude, phase, and complex demodulate.

Usage

```
demod.dpss(x,centreFreq,nw,blockLen,stepSize=1,wrapphase=TRUE,...)
```

Arguments

`x`	Time series, required to be contiguous.
`centreFreq`	Frequency around which to demodulate.
`nw`	Parameter controlling time-bandwidth.
`blockLen`	Length of sub-block to use; demodulate is computed on each block in turn.
`stepSize`	This is a proposed option that sets the index step size between blocks. Currently this must be set to 1 and changes in step size have not been implemented.
`wrapphase`	If true, routine wraps phases around +/-360 degree boundaries.
`...`	Additional arguments. Currently only includes depreciated arguments

References

Thomson, D.J. (1995). The Seasons, Global Temperature, and Precession. *Science*, Volume 268, pp. 59–68.

Bloomfield P. (2000). Fourier Analysis of Time Series. 2nd edition. Wiley New York, pp. 97–130.

Examples

```
data(CETmonthly)
nJulOff <- 1175
xd <- ts(CETmonthly[,"temp"],deltat=1/12)
demodYr <- demod.dpss(xd,centreFreq=1,nw=3,blockLen=120,stepSize=1)
phase <- demodYr$phase
offsJul <- 3*360/365
phaseAdj <- phase
phaseAdj[1:nJulOff] <- phase[1:nJulOff] + offsJul
yr <- (time(xd)+1658)[1:length(phase)]
plot(yr, phaseAdj, type="l", lwd=2,
     ylab="Phase of the Year in Degrees",
     xlab="Gegorian calender date")
lines((1:nJulOff)/12+1659, phase[1:nJulOff], col="red")
fit <- lm( phaseAdj ~ yr)
abline(fit, lty=2, col="blue")
cat(paste("Precession Estimate: ",fit$coef[2]*60*60,digits=6," (arcseconds/yr)\n",sep=""))
```

---

dpss *Compute Discrete Prolate Spheroidal Sequences*

Description

Compute Discrete Prolate Spheroidal (Slepian) Sequences for use as tapers or other applications. This function uses the tridiagonal method and exploits symmetry. Note the odd order tapers are normalized so that the slope at the centre is positive in accordance with Slepian (1978) and Thomson (1982). This differs from Percival and Walden (1993). This code follows section (8.3) of Percival and Walden (1993) using LAPACK function calls Anderson (1999).

Usage

```
dpss(n,k,nw, returnEigenvalues=TRUE)
```

Arguments

n	A positive integer, typically the non-zero-padded length of the time series.
k	A positive integer, the number of tapers, often 2*nw for spectrum estimation purposes.
nw	A positive double-precision number, the time-bandwidth parameter.
returnEigenvalues	If true the appropriate eigenvalues are calculated and returned using the function dpssToEigenvalues. If FALSE, the eigenvalues returned are from the LAPACK function DSTEBZ using the tridiagonal. See section 8.3 of Percival and Walden (1993), or equation (13) in Slepian (1978).

Value

v	A n by k matrix of Slepian Sequences. Each column represents the Slepian sequence of order k-1.
eigen	A length k vector of eigenvalues corresponding to equation (13) in Slepian (1978), or the eigenvalues of the input tridiagonal matrix returned from the internal call to the LAPACK function DSTEBZ.

References

Anderson, E. (1999). *LAPACK Users' guide (Vol. 9).* Siam.

Percival, D.B. and Walden, A.T. (1993) *Spectral analysis for physical applications.* Cambridge University Press.

Slepian, D. (1978) Prolate spheroidal wave functions, Fourier analysis, and uncertainty. V–The discrete case. *Bell System Technical Journal* Volume **57**, pp. 1371–1430

Thomson, D.J (1982) Spectrum estimation and harmonic analysis. *Proceedings of the IEEE* Volume **70**, number 9, pp. 1055–1096.

Examples

```
dpss(10,4,4.0)
dpss(100,8,5.0)
```

---

dpssToEigenvalues *Compute eigenvalues for the Discrete Prolate Spheroidal Sequences (dpss)*

Description

Compute eigenvalues for the Discrete Prolate Spheroidal Sequences. The method used here is described in Chapter 8 of Percival and Walden (1993).

Usage

```
dpssToEigenvalues(v, nw)
```

Arguments

v	A matrix of dpss's, with each column representing a sequence of a different order, 1 to k.
nw	A positive double-precision number, the time-bandwidth parameter.

References

Percival, D.B. and Walden, A.T. (1993) *Spectral analysis for physical applications.* Cambridge University Press.

Examples

```
dpss1 <- dpss(10,4,4.0, returnEigenvalues=FALSE)$v
dpssToEigenvalues(dpss1,4.0)
```

---

dropFreqs *Truncate mtm or mtm.coh Objects in Frequency*

Description

A utility function to truncate the frequencies in a spectral estimate. This utility is used before calling plot(), to increase the visual frequency resolution of a plot by truncating frequencies outside a particular band of interest. This function is not a filter, but rather a utility to allow R to 'zoom' a spectrum plot to a certain frequency band.

Usage

```
dropFreqs(spec, minFreq, maxFreq)
```

Arguments

spec	A spectrum object 'obj', of class spec, mtm, or mtm.coh.
minFreq	The lower bound for the frequency band to be retained, in the same units as the obj$freq array.
maxFreq	The upper bound for the frequency band to be retained, also in the same units as the obj$freq array.

Examples

```
data(willamette)
mtm1 <- spec.mtm(willamette, nw=4.0, k=8, plot=FALSE, deltat=1.0, dtUnits="month")
mtm2 <- dropFreqs(mtm1, 0.1, 0.4)
plot(mtm2)

# another option
plot(dropFreqs(mtm1, 0.1, 0.4))

# using sine tapers
mtm.sine <- spec.mtm(willamette, k=10, plot=FALSE, deltat=1.0, dtUnits="month",
                     taper="sine", sineAdaptive=FALSE, sineSmoothFact=0.05)
plot(dropFreqs(mtm.sine, 0.1, 0.4))
```

HadCRUTnh *HadCRUT Land Temperature Anomaly (Northern Hemisphere) Series*

Description

Hadley Climate Research Unit Temperature anomaly (Northern Hemisphere) time series. Consists of monthly observations, truncated to start at March 1958 and extend to September 2009, to match mlco2 dataset. This dataset was retrieved from the Hadley CRU on Oct 1, 2011.

Usage

```
HadCRUTnh
```

Format

A data frame indicating the year, month and temperature anomaly for the Northern Hemisphere, between 1958 and 2009.

mlco2 *Mauna Loa Observatory CO2 Monthly Averages*

Description

Observations of monthly CO2 atmospheric concentration averages from the Mauna Loa Observatory, Mauna Loa, Hawaii, USA. Obtained from the ESRL Global Monitoring Division of the National Oceanic and Atmospheric Administration at http://www.esrl.noaa.gov/gmd/dv/data/index.php?parameter_name=Carbo
Dataset downloaded Oct 1, 2011.

Usage

```
mlco2
```

Format

A data frame indicating the year, month and atmospheric concentration of CO2 in PPM.

mtm.coh *Compute and plot the multitaper magnitude-squared coherence.*

Description

Computes and plots the adaptive multitaper spectrum estimate.

Usage

```
mtm.coh(mtm1, mtm2, fr=NULL, tau=0, phcorr = TRUE, plot=TRUE, ...)
```

Arguments

mtm1	An object created with spec.mtm(... ,returnInternals=TRUE).
mtm2	An object created with spec.mtm(... ,returnInternals=TRUE). Note mtm1 and mtm2 must be created with the same frequency resolution. They both must have the same values for nFFT and returnZeroFreq.
fr	The frequency values for the mtm object. This can be null by default (which results in computation for the full frequency range) or it can be a subset of frequency values.
tau	Phase-correction factor, if known.
phcorr	Correct phase (unwrap). By default, set to TRUE; set to FALSE if you would prefer the phase to be untouched.
plot	Boolean value indicating if a plot should be drawn.
...	Additional parameters, such as xaxs="i" which are passed through to the plotting function.

References

Thomson, DJ (1991) Jackknifed error estimates for spectra, coherences, and transfer functions, *Advances in Spectrum Estimation* 58–113.

Thomson, D.J (1982) Spectrum estimation and harmonic analysis. *Proceedings of the IEEE* Volume **70**, number 9, pp. 1055–1096.

Percival, D.B. and Walden, A.T. (1993) *Spectral analysis for physical applications* Cambridge University Press.

Examples

```
data(HadCRUTnh)
data(mlco2)
spec1 <- spec.mtm(HadCRUTnh, nw=5.0, k=8, plot=FALSE,
    returnInternals=TRUE, dtUnits="month", deltat=1.0)
spec2 <- spec.mtm(mlco2, nw=5.0, k=8, plot=FALSE, returnInternals=TRUE,
    dtUnits="month", deltat=1.0)
resCoh <- mtm.coh(spec1, spec2, plot=FALSE)
plot(resCoh)
plot(resCoh, cdfQuantilesTicks=1-10^(-(6:12)))
```

multitaperTrend *Estimate Linear Trend using Multitaper Techniques*

Description

Estimate linear trend using inverse spectrum estimation, with the spectrum being computed via multitaper. This technique has improved spectral properties when compared to the least-squares approach. Returned values from this function include the intercept, slope, and centered time array.

Usage

```
multitaperTrend(xd, B, deltat, t.in)
```

Arguments

xd	Contiguous time series to be detrended.
B	Bandwidth to use in estimating trend in physical units; corresponds to NW via equation NW=BT, where N and W are the usual Slepian definitions, and T is the total time elapsed, i.e. T = N*deltat.
deltat	Time step for series xd, also used in computing T.
t.in	Time array, used in accurately estimating the slope.

Examples

```
x <- 1:101
y <- 1.0 + 0.5*(x) + rnorm(n=101,mean=0,sd=2)
vars <- multitaperTrend(xd=y, B=0.05, deltat=1.0, t.in=x)
plot(x,y,type="l")
lines(x,vars[[1]]+vars[[2]]*vars[[3]],type="l",col="red")
```

percivalAR4 *Auto Regressive Series generated by Don Percival at Applied Physics Laboratory*

Description

This is a simulated AR(4) time series (page 45 of Percival and Walden 1993). The source of this series is: Applied Physics Laboratory (Don Percival). The value for delta T is 1, and the sample size is 1024. Another realization of this series based on the same autoregressive coefficients can be generated in R using the code in the example section of the documentation.

Usage

```
percivalAR4
```

Format

A time series object containing 1024 simulated values.

Source

Presented on page 45 of Percival, D.B. and Walden, A.T. (1993). See: http://faculty.washington.edu/dbp/DATA/ar4.dat

References

Percival, D.B. and Walden, A.T. (1993) *Spectral analysis for physical applications.* Cambridge University Press.

Examples

```
## get the Percival realization of the series saved as data.
data(percivalAR4)
## generate another realization of this series using the same AR(4)
## coefficients.
ar4Coef <- c(2.7607, -3.8106, 2.6535, -0.9238)
ar4.ts <- arima.sim(list(order = c(4, 0, 0), ar=ar4Coef), n=1024)
```

---

plot.mtm *Compute and plot the multitaper spectrum estimate*

Description

Plots the multitaper spectral estimate and the multitaper F-test.

Usage

```
## S3 method for class 'mtm'
plot(x, jackknife = FALSE, Ftest = FALSE, ftbase = 1.01, siglines = NULL, ...)
```

Arguments

x	An object of the class mtm generated by spec.mtm.
jackknife	Boolean variable indicating if jackknife confidence intervals should be plotted, only applies if Ftest=FALSE.
Ftest	Boolean variable indicating if the multitaper harmonic F-test should be plotted instead of the spectrum.
ftbase	Lowest value to be plotted when the F-test is plotted. When Ftest = TRUE, max(ftestvalue, ftbase) is plotted.
siglines	Vector of significance values (as probabilities, 0.0 to 1.0) to plot as horizontal significance lines on the F-test plot.
...	Arguments to be passed to methods, such as graphical parameters (see 'par').

Details

The value log can be set to "yes" (default), "no", or "dB" as in the function plot.spec.

References

Thomson, D.J (1982) Spectrum estimation and harmonic analysis. *Proceedings of the IEEE* Volume **70**, number 9, pp. 1055–1096.

Percival, D.B. and Walden, A.T. (1993) *Spectral analysis for physical applications* Cambridge University Press.

See Also

`spec.mtm` and `plot.spec`

Examples

```
data(willamette)
resSpec <- spec.mtm(willamette, nw=4.0, k=8, Ftest=TRUE, plot=FALSE, deltat=1.0, dtUnits="month")
plot(resSpec)
plot(resSpec, Ftest=TRUE)
plot(resSpec, Ftest=TRUE, siglines=c(0.90, 0.99))

# with jackknife estimate
resSpec2 <- spec.mtm(willamette, nw=4.0, k=8, Ftest=TRUE, jackknife=TRUE, plot=FALSE,
                     deltat=1.0, dtUnits="month")
plot(resSpec2,jackknife=TRUE)
```

---

plot.mtm.coh *Compute and plot the multitaper magnitude-squared coherence.*

Description

Plots the magnitude-squared coherence for a mtm.coh object computed from two equal-parameter mtm objects.

Usage

```
## S3 method for class 'mtm.coh'
plot(x,percentGreater=NULL,nehlim=10,nehc=4, cdfQuantilesTicks=NULL,
                       drawPercentLines=TRUE, percentG=c(.1,.2,.5,.8,.9), ...)
```

Arguments

x	An object of the class mtm.coh generated by spec.mtm.coh.
percentGreater	Prints the percent of the coherence function greater than the given values in the lower left hand corner. The values are expected in a vector representing the percentages. For example c(.5, .7) will print the percent of the msc that is greater than 50% and 70%.
nehlim	A smoothing parameter used in smoothing the variance in the final plot. nehlim is the number of points to smooth by on each side.
nehc	A smoothing parameter used in smoothing the MSC in the final plot. nehc is the number of points to smooth by on each side.
cdfQuantilesTicks	Percent lines to place the tick marks on the right axis (CDF). See the example in mtm.coh for use.
drawPercentLines	Boolean variable indicating if significance lines are to be drawn.
percentG	A vector of values for which to print dashed lines indicating the percent greater than a particular value. If drawpPercentLines is FALSE then this will not be used.
...	Parameters passed to plotting function. Currently only tested with xaxs="i".

Details

Returns an object containing the user-specified significance levels (percentG), along with their normal transforms. This allows the user to examine the mtm.coh object and clip the NTmsc variable to different significances.

References

Thomson, D.J (1982) Spectrum estimation and harmonic analysis. *Proceedings of the IEEE* Volume **70**, number 9, pp. 1055–1096.

Percival, D.B. and Walden, A.T. (1993) *Spectral analysis for physical applications* Cambridge University Press.

Examples

```
# examples here
```

---

sineTaper *Computes sine tapers*

Description

Computes sine tapers for use in transfer function estimation and plotting. Not called from within spec.mtm.

Usage

```
sineTaper(n, k)
```

Arguments

n	The data as a vector or as a time series.
k	The Slepian bandwidth parameter, typically between 2.0 and 6.0.

References

Riedel, K.S. and Sidorenko, A. (1995) Minimum bias multiple taper spectral estimation. *IEEE Transactions on Signal Processing*, Volume **43**, Number **1**, pp. 188–195.

---

spec.mtm *Compute and plot multitaper spectrum estimates*

Description

Computes and plots adaptive or nonadaptive multitaper spectrum estimates from contiguous time series objects.

Usage

```
spec.mtm(timeSeries, nw=4.0, k=7, nFFT="default", taper=c("dpss"),
         centre=c("Slepian"), dpssIN=NULL, returnZeroFreq=TRUE,
         Ftest=FALSE, jackknife=FALSE, jkCIProb=.95, adaptiveWeighting=TRUE,
         maxAdaptiveIterations=100, plot=TRUE, na.action=na.fail,
         returnInternals=FALSE, sineAdaptive=FALSE, sineSmoothFact=0.2,
         dtUnits=c("default"), deltat=NULL, ...)
```

Arguments

timeSeries	A time series of equally spaced data, this can be created by the ts() function where deltat is specified.
nw	nw a positive double precision number, the time-bandwidth parameter.
k	k a positive integer, the number of tapers, often 2*nw.
nFFT	This function pads the data before computing the fft. nFFT indicates the total length of the data after padding.
taper	Choose between dpss-based multitaper (the default,'dpss') or sine taper method. In the case of the sine taper, parameter nw is useless, and both Ftest and jackknife are forced to FALSE. The sine taper also has two specific parameters below.
centre	The time series is centred using one of three methods: expansion onto discrete prolate spheroidal sequences ('Slepian'), arithmetic mean ('arithMean'), trimmed mean ('trimMean'), or not at all ('none').

`dpssIN` Allows the user to enter a dpss object which has already been created. This can save computation time when Slepians with the same bandwidth parameter and same number of tapers are used repeatedly.

`returnZeroFreq` Boolean variable indicating if the zeroth frequency (DC component) should be returned for all applicable arrays.

`Ftest` Boolean variable indicating if the Ftest result should be computed and returned.

`jackknife` Boolean variable indicating if jackknifed confidence intervals should be computed and returned.

`jkCIProb` Decimal value indicating the jackknife probability for calculating jackknife confidence intervals. The default returns a 95% confidence interval.

`adaptiveWeighting` Boolean flag for enabling/disabling adaptively weighted spectrum estimates. Defaults to `TRUE`. The `FALSE` case gives complex Fourier transforms equivalent to direct estimates with Slepian sequences as tapers.

`maxAdaptiveIterations` Maximum number of iterations in the adaptive multitaper calculation. Generally convergence is quick, and should require less than 100 iterations.

`plot` Boolean variable indicating if the spectrum should be plotted.

`na.action` Action to take if NAs exist in the data, the default is to fail.

`returnInternals` Return the weighted eigencoefficients, complex mean values, and so on. These are necessary for extensions to the multitaper, including magnitude-squared coherence (function mtm.coh in this package). Note: The internal ($mtm) variables eigenCoefs and eigenCoefWt correspond to the multitaper eigencoefficients. The eigencoefficients correspond to equation (3.4) and weights, eigenCoefWt, correspond to sqrt(ld_k(f)|^2) from equation (5.4) in Thomson's 1982 paper. This is because the square root values contained in eigenCoefWt are commonly used in additional calculations (example: eigenCoefWt * eigenCoefs). The values returned in mtm$cmv correspond to the the estimate of the coefficients hat(mu)(f) in equation (13.5) in Thomson (1982), or to the estimate of hat(C)_1 at frequency 1 in equation (499) form Percival and Walden (1993)

`sineAdaptive` In the case of using the sine taper method, choose between non-adaptive and adaptive taper choice.

`sineSmoothFact` The sine taper option has an inherent smoothing parameter that can be set between 0.01 and 0.5. Lower values indicate smaller amounts of smoothing.

`dtUnits` Allows indication of the units of delta-t for accurate frequency axis labels.

`deltat` Time step for observations. If not in seconds, dtUnits should be set to indicate the proper units for plot labels.

`...` Additional parameters, such as xaxs="i" which are passed to the plotting function. Not all parameters are supported.

Details

The value log can be set to "yes" (default), "no", or "dB" as in the function plot.spec.

References

Thomson, D.J (1982) Spectrum estimation and harmonic analysis. *Proceedings of the IEEE* Volume **70**, Number 9, pp. 1055–1096.

Percival, D.B. and Walden, A.T. (1993) *Spectral analysis for physical applications* Cambridge University Press.

Riedel, K.S. and Sidorenko, A. (1995) Minimum bias multiple taper spectral estimation. *IEEE Transactions on Signal Processing* Volume **43**, Number 1, pp. 188–195.

See Also

`plot.mtm` and `plot.spec`

Examples

```
## default behaviour, dpss tapers; deltat and dtUnits set to ensure axis accuracy
data(willamette)
spec.mtm(willamette, nw=4.0, k=8, deltat=1/12, dtUnits="year")
spec.mtm(willamette, nw=4.0, k=8, nFFT=2048, deltat=1/12, dtUnits="year")

## if you have a ts object, you can skip the deltat and dtUnits parameters
will.ts <- ts(data=willamette, start=1950.75, freq=12)
spec.mtm(will.ts, nw=4.0, k=8)

## using Sine Tapers
spec.mtm(will.ts, k=10, taper="sine", sineAdaptive=FALSE)
spec.mtm(will.ts, k=10, taper="sine", sineAdaptive=TRUE,
         maxAdaptiveIterations=100, sineSmoothFact=0.05)
```

---

willamette *Willamette River time series*

Description

Willamette River time series. Each point represents the log of the average daily flow over a one month period from October, 1950, to August 1983. The sampling time is 1/12 year and the Nyquist frequency is 6 cycles per year. The data is from the companion code to "Spectral Analysis for the Physical Applications" (1993) and was originally compiled by the US Geological Survey.

Usage

```
willamette
```

Format

A vector containing 395 observations

References

Percival, D.B. and Walden, A.T. (1993) *Spectral analysis for physical applications.* Cambridge University Press.

Examples

```
data(willamette)
# time series object, January = year.0, December = year.917
will.ts <- ts(data=willamette, start=(1950+9/12), freq=12)
```

Index

∗ **datasets**
  CETdaily, 3
  CETmonthly, 3
  HadCRUTnh, 8
  mlco2, 8
  percivalAR4, 10
  willamette, 16
∗ **math**
  centre, 2
  demod.dpss, 4
  dpss, 5
  dpssToEigenvalues, 6
  multitaperTrend, 10
  sineTaper, 13
∗ **multitaper**
  dropFreqs, 7
  mtm.coh, 9
  plot.mtm, 11
  plot.mtm.coh, 12
  spec.mtm, 14

centre, 2
CETdaily, 3
CETmonthly, 3

demod.dpss, 4
dpss, 5
dpssToEigenvalues, 6
dropFreqs, 7

HadCRUTnh, 8

mlco2, 8
mtm.coh, 9
multitaperTrend, 10

percivalAR4, 10
plot.mtm, 11, *16*
plot.mtm.coh, 12
plot.spec, *12*, *16*

sineTaper, 13
spec.mtm, *12*, 14

willamette, 16