

# Package ‘HKprocess’

October 27, 2022

**Version** 0.1-1

**Date** 2022-10-26

**Title** Hurst-Kolmogorov Process

**Description** Methods to make inference about the Hurst-Kolmogorov (fractional Gaussian noise, fGn) and the AR(1) process. Related time series trend tests are also included.

**Imports** MCMCpack (>= 1.3-3), gtools(>= 3.5.0)

**Depends** R (>= 3.2.3)

**Suggests** ltsa (>= 1.4.6)

**License** GPL-3

**Author** Hristos Tyralis [aut, cre] (<<https://orcid.org/0000-0002-8932-4997>>)

**Maintainer** Hristos Tyralis <montchrister@gmail.com>

**Repository** CRAN

**NeedsCompilation** yes

**Date/Publication** 2022-10-26 22:17:59 UTC

## R topics documented:

acfHKp . . . . .	2
inferf . . . . .	3
inferfmetrop . . . . .	4
inferH . . . . .	5
inferHmetrop . . . . .	6
infermsfmetrop . . . . .	8
infermsmetrop . . . . .	9
lssd . . . . .	10
lsv . . . . .	11
ltza . . . . .	12
ltzb . . . . .	13
ltzc . . . . .	14
ltzd . . . . .	15
MannKendallLTP . . . . .	16

mlear1 . . . . .	17
mleHK . . . . .	18

<b>Index</b>	<b>20</b>
--------------	-----------

acfHKp *Autocorrelation of HKp*

## Description

The Hurst-Kolmogorov stochastic process (HKp) is a persistent process. The term HKp is an alternative of the term Fractional Gaussian Noise (FGN, see Koutsoyiannis 2010). Its autocorrelation function (ACF) is given by eq.16 (Koutsoyiannis 2002).

## Usage

```
acfHKp(H, maxlag)
```

## Arguments

H	Hurst parameter
maxlag	ACF computed at lags 0, 1, ..., maxlag

## Value

Vector of autocorrelations at lags 0, 1, ..., maxlag.

## Note

The parameter H should be in (0, 1), see Koutsoyiannis (2002)

## Author(s)

Hristos Tyralis

## References

Koutsoyiannis D (2002) The Hurst phenomenon and fractional Gaussian noise made easy. *Hydrological Sciences Journal* **47**(4):573–595. doi:10.1080/02626660209492961.

Koutsoyiannis D (2010) A random walk on water. *Hydrology and Earth System Sciences* **14**:585–601. doi:10.5194/hess145852010.

## Examples

```
# Compute the ACF at lags 0, 1, ..., 10 when H = 0.8.
```

```
acfHKp(0.8, 10)
```

---

infern	<i>Posterior distribution of the <math>\phi</math> parameter of the AR(1) process, using an Accept-Reject algorithm</i>
--------	---

---

### Description

The function `infern` is used to create a sample from the posterior distribution of  $\phi$ . The function uses the eq.10 (Tyrallis and Koutsoyiannis 2014) to make inference on  $\phi$  and an Accept-Reject algorithm (see Robert and Casella 2004, Algorithm A.4).

### Usage

```
infern(data, n, add = 0.001, minu = -0.999, maxu = 0.999)
```

### Arguments

<code>data</code>	time series data
<code>n</code>	The size of the simulated sample
<code>add</code>	A number added to the maximum value of the natural logarithm of eq.10, to avoid bugs of the Accept-Reject algorithm due to computation errors
<code>minu</code>	A lower bound to the parameter $\phi$
<code>maxu</code>	An upper bound to the parameter $\phi$

### Value

Vector with the simulated sample.

### Author(s)

Hristos Tyrallis

### References

Robert CP, Casella G (2004) *Monte Carlo Statistical Methods*. New York: Springer. doi:10.1007/9781475741452.

Tyrallis H, Koutsoyiannis D (2014) A Bayesian statistical model for deriving the predictive distribution of hydroclimatic variables. *Climate Dynamics* **42(11-12)**:2867–2883. doi:10.1007/s00382-0131804y.

### Examples

```
# Posterior distribution of the phi parameter of the AR(1) process for the Nile
# time series.

set.seed(12345)

samp.sim <- inferf(Nile,500)
```

```
hist(samp.sim, breaks = 20, main = expression(paste("Histogram of ",phi)),
     xlab = expression(phi))
```

---

inferfmetrop	<i>Posterior distribution of the <math>\phi</math> parameter of the AR(1) process, using a Metropolis algorithm.</i>
--------------	--

---

### Description

The function `inferfmetrop` is used to create a sample from the posterior distribution of  $\phi$ . The function uses the eq.10 in Tyrakis and Koutsoyiannis (2014) and a Metropolis algorithm to make inference on  $\phi$ .

### Usage

```
inferfmetrop(data, theta.init = 0.7, burnin = 500, mcmc = 20000, thin = 1,
             tune = 1, verbose = 0, seed = NA)
```

### Arguments

<code>data</code>	time series data
<code>theta.init</code>	Starting values for the sampling. Must be of the appropriate dimension. It must also be the case that <code>fun(theta.init, ...)</code> is greater than <code>-Inf</code> .
<code>burnin</code>	The number of burn-in iterations for the sampler.
<code>mcmc</code>	The number of MCMC iterations after burnin.
<code>thin</code>	The thinning interval used in the simulation. The number of MCMC iterations must be divisible by this value.
<code>tune</code>	The tuning parameter for the Metropolis sampling. Can be either a positive scalar or a $k$ -vector, where $k$ is the length of $\theta$ .
<code>verbose</code>	A switch which determines whether or not the progress of the sampler is printed to the screen. If <code>verbose</code> is greater than 0 the iteration number, the $\theta$ vector, the function value, and the Metropolis acceptance rate are sent to the screen every <code>verbose</code> iteration.
<code>seed</code>	The seed for the random number generator. If NA, the Mersenne Twister generator is used with default seed 12345; if an integer is passed it is used to seed the Mersenne twister. The user can also pass a list of length two to use the L'Ecuyer random number generator, which is suitable for parallel computation. The first element of the list is the L'Ecuyer seed, which is a vector of length six or NA (if NA a default seed of <code>rep(12345, 6)</code> is used). The second element of list is a positive substream number. See the MCMCpack specification for more details.

### Value

An `mcmc` object that contains the posterior sample. This object can be summarized by functions provided by the `coda` package.

**Note**

The Metropolis algorithm uses the function `MCMCmetrop1R` from the package `MCMCpack` (Martin et al. 2011).

**Author(s)**

Hristos Tyralis

**References**

Martin AD, Quinn KM, Park JH (2011) MCMCpack: Markov chain Monte Carlo in R. *Journal of Statistical Software* **42(9)**:1–21. doi:10.18637/jss.v042.i09.

Tyralis H, Koutsoyiannis D (2014) A Bayesian statistical model for deriving the predictive distribution of hydroclimatic variables. *Climate Dynamics* **42(11-12)**:2867–2883. doi:10.1007/s00382-0131804y.

**Examples**

```
# Posterior distribution of the phi parameter of the AR(1) process for the Nile
# time series.

samp.sim <- inferfmetrop(Nile, theta.init = 0.7, burnin = 500, mcmc = 500,
  thin = 1, tune = 1, seed = 12345)

hist(samp.sim, breaks = 20, main = expression(paste("Histogram of ",phi)),
  xlab = expression(phi))
```

---

inferH

*Posterior distribution of the H parameter of the HKp using an Accept-Reject algorithm.*

---

**Description**

The function `inferH` is used to create a sample from the posterior distribution of  $H$ . The function uses the eq.10 (Tyralis and Koutsoyiannis 2014) to make inference on  $H$  and an Accept-Reject algorithm (see Robert and Casella 2004, Algorithm A.4).

**Usage**

```
inferH(data, n, add = 0.001, minu = 0.001, maxu = 0.999)
```

**Arguments**

<code>data</code>	time series data
<code>n</code>	The size of the simulated sample
<code>add</code>	A number added to the maximum value of the natural logarithm of eq.10, to avoid bugs of the Accept-Reject algorithm due to computation errors
<code>minu</code>	A lower bound to the parameter $H$
<code>maxu</code>	An upper bound to the parameter $H$

**Value**

Vector with the simulated sample.

**Author(s)**

Hristos Tyralis

**References**

Robert CP, Casella G (2004) *Monte Carlo Statistical Methods*. New York: Springer. doi:10.1007/9781475741452.

Tyralis H, Koutsoyiannis D (2014) A Bayesian statistical model for deriving the predictive distribution of hydroclimatic variables. *Climate Dynamics* **42(11-12)**:2867–2883. doi:10.1007/s00382-0131804y.

**Examples**

```
# Posterior distribution of the H parameter of the HKp for the Nile time series.

set.seed(12345)

samp.sim <- inferH(Nile,500)

hist(samp.sim, breaks = 20, main = "Histogram of H", xlab = "H")
```

---

inferHmetrop	<i>Posterior distribution of the H parameter of the HKp, using a Metropolis algorithm.</i>
--------------	--

---

**Description**

The function inferHmetrop is used to create a sample from the posterior distribution of H. The function uses the eq.10 in Tyralis and Koutsoyiannis (2014) and a Metropolis algorithm to make inference on H.

**Usage**

```
inferHmetrop(data, theta.init=0.7, burnin = 500, mcmc = 20000, thin = 1,
             tune = 1, verbose = 0, seed = NA)
```

**Arguments**

data	time series data
theta.init	Starting values for the sampling. Must be of the appropriate dimension. It must also be the case that fun(theta.init, ...) is greater than -Inf.
burnin	The number of burn-in iterations for the sampler.
mcmc	The number of MCMC iterations after burnin.

thin	The thinning interval used in the simulation. The number of MCMC iterations must be divisible by this value.
tune	The tuning parameter for the Metropolis sampling. Can be either a positive scalar or a $k$ -vector, where $k$ is the length of $\theta$ .
verbose	A switch which determines whether or not the progress of the sampler is printed to the screen. If verbose is greater than 0 the iteration number, the $\theta$ vector, the function value, and the Metropolis acceptance rate are sent to the screen every verbose iteration.
seed	The seed for the random number generator. If NA, the Mersenne Twister generator is used with default seed 12345; if an integer is passed it is used to seed the Mersenne twister. The user can also pass a list of length two to use the L'Ecuyer random number generator, which is suitable for parallel computation. The first element of the list is the L'Ecuyer seed, which is a vector of length six or NA (if NA a default seed of rep(12345, 6) is used). The second element of list is a positive substream number. See the MCMCpack specification for more details.

### Value

An mcmc object that contains the posterior sample. This object can be summarized by functions provided by the coda package.

### Note

The Metropolis algorithm uses the function MCMCmetrop1R from the package MCMCpack (Martin et al. 2011).

### Author(s)

Hristos Tyralis

### References

Martin AD, Quinn KM, Park JH (2011) MCMCpack: Markov chain Monte Carlo in R. *Journal of Statistical Software* **42(9)**:1–21. doi:10.18637/jss.v042.i09.

Tyralis H, Koutsoyiannis D (2014) A Bayesian statistical model for deriving the predictive distribution of hydroclimatic variables. *Climate Dynamics* **42(11-12)**:2867–2883. doi:10.1007/s00382-0131804y.

### Examples

```
# Posterior distribution of the H parameter of the HKp for the Nile time series.

samp.sim <- inferHmetrop(Nile, theta.init = 0.7, burnin = 500, mcmc = 500,
  thin = 1, tune = 1, seed = 12345)

hist(samp.sim, breaks = 20, main = "Histogram of H", xlab = "H")
```

---

infermsfmetrop      *Bayesian inference for  $\mu$  and  $\sigma^2$  for the AR(1) process.*

---

### Description

The function `infermsfmetrop` is used to create a sample from the posterior distribution of  $\mu$  and  $\sigma^2$ , using eqs. 8,9 in Tyralis and Koutsoyiannis (2014) for the AR(1) process.

### Usage

```
infermsfmetrop(fbayes, data)
```

### Arguments

<code>data</code>	time series data
<code>fbayes</code>	phi parameter simulation sample, should be a vector

### Value

A matrix with two columns, the first corresponding to the sample of  $\mu$  and the second corresponding to the sample of  $\sigma^2$ .

### Author(s)

Hristos Tyralis

### References

Tyralis H, Koutsoyiannis D (2014) A Bayesian statistical model for deriving the predictive distribution of hydroclimatic variables. *Climate Dynamics* **42(11-12)**:2867–2883. doi:10.1007/s00382-0131804y.

### Examples

```
# Posterior distribution of the mu and sigma parameters of the AR(1) process for
# the Nile time series.

set.seed(12345)

samp.sim1 <- inferf(Nile, 500)

samp.sim2 <- infermsfmetrop(samp.sim1, Nile)

hist(samp.sim2[,1], breaks = 20, main = expression(paste("Histogram of ",mu)),
     xlab = expression(paste(mu)))

hist(sqrt(samp.sim2[,2]), breaks = 20,
     main = expression(paste("Histogram of ",sigma)),
     xlab = expression(paste(sigma)))
```



---

infermsmetrop	<i>Bayesian inference for <math>\mu</math> and <math>\sigma^2</math> for the HKp.</i>
---------------	---

---

**Description**

The function infermsmetrop is used to create a sample from the posterior distribution of  $\mu$  and  $\sigma^2$ , using eqs. 8,9 in Tyralis and Koutsoyiannis (2014) for the HKp.

**Usage**

```
infermsmetrop(Hbayes, data)
```

**Arguments**

data	time series data
Hbayes	Hurst parameter simulation sample, should be a vector

**Value**

A matrix with two columns, the first corresponding to the sample of  $\mu$  and the second corresponding to the sample of  $\sigma^2$ .

**Author(s)**

Hristos Tyralis

**References**

Tyralis H, Koutsoyiannis D (2014) A Bayesian statistical model for deriving the predictive distribution of hydroclimatic variables. *Climate Dynamics* **42(11-12)**:2867–2883. doi:[10.1007/s00382-0131804y](https://doi.org/10.1007/s00382-0131804y).

**Examples**

```
# Posterior distributions of the mu and sigma parameters of the HKp for the Nile
# time series.

set.seed(12345)

samp.sim1 <- inferH(Nile, 500)

samp.sim2 <- infermsmetrop(samp.sim1, Nile)

hist(samp.sim2[,1], breaks = 20, main = expression(paste("Histogram of ",mu)),
     xlab = expression(paste(mu)))

hist(sqrt(samp.sim2[,2]), breaks = 20,
     main = expression(paste("Histogram of ",sigma)),
     xlab = expression(paste(sigma)))
```

---

 lssd

*LSSD estimation for the HKp parameters.*


---

### Description

The function `lssd` is used to estimate the  $\sigma$  and  $H$  parameters of the HKp, using the LSSD (Least Squares based on Standard Deviation) method as described in Koutsoyiannis (2003) and Tyralis and Koutsoyiannis (2011, Section 2.3).

### Usage

```
lssd(data, k1, p = 2, q = 0, interval = c(0.001, 0.999))
```

### Arguments

<code>data</code>	time series data
<code>k1</code>	maximum aggregation scale
<code>p</code>	Parameter used to determine the weights
<code>q</code>	Parameter used to determine the penalty factor
<code>interval</code>	H interval estimation

### Value

Vector of LSSD estimates of  $\sigma$  and  $H$ .

### Author(s)

Hristos Tyralis

### References

Koutsoyiannis D (2003) Climate change, the Hurst phenomenon, and hydrological statistics. *Hydrological Sciences Journal* **48**(1):3–24. doi:10.1623/hysj.48.1.3.43481.

Tyralis H, Koutsoyiannis D (2011) Simultaneous estimation of the parameters of the Hurst-Kolmogorov stochastic process. *Stochastic Environmental Research & Risk Assessment* **25**(1):21–33. doi:10.1007/s004770100408x.

### Examples

```
# Estimate the parameters for the Nile time series.

lssd(data = Nile, k1 = floor(length(Nile)/10), p = 2)
```

---

`lsv`*LSV estimation for the HKp parameters.*

---

**Description**

The function `lsv` is used to estimate the  $\sigma$  and  $H$  parameters of the HKp, using the LSV (Least Squares based on Variance) method as described in Tyralis and Koutsoyiannis (2011, Section 2.4).

**Usage**

```
lsv(data, k1, p = 6, q = 0, interval = c(0.001, 0.999))
```

**Arguments**

<code>data</code>	time series data
<code>k1</code>	maximum aggregation scale
<code>p</code>	Parameter used to determine the weights
<code>q</code>	Parameter used to determine the penalty factor
<code>interval</code>	H interval estimation

**Value**

Vector of LSV estimates of  $\sigma$  and  $H$ .

**Author(s)**

Hristos Tyralis

**References**

Tyralis H, Koutsoyiannis D (2011) Simultaneous estimation of the parameters of the Hurst-Kolmogorov stochastic process. *Stochastic Environmental Research & Risk Assessment* **25(1)**:21–33. doi:[10.1007/s004770100408x](https://doi.org/10.1007/s004770100408x).

**Examples**

```
# Estimate the parameters for the Nile time series.  
  
lsv(data = Nile, k1 = floor(length(Nile)/10), p = 6, q = 50)
```

---

ltza	<i>Value of quadratic forms for the inverse of a symmetric positive definite autocorrelation matrix.</i>
------	--

---

### Description

The function `ltza` is used to calculate the value of quadratic forms for the inverse of a symmetric positive definite autocorrelation matrix, using the Levinson algorithm (Golub and Van Loan 1996, Algorithm 4.7.2).

### Usage

```
ltza(r, x)
```

### Arguments

<code>r</code>	autocorelation vector
<code>x</code>	time series data

### Value

Vector with values  $t(x) * \text{solve}(R) * x$ ,  $t(en) * \text{solve}(R) * x$ ,  $t(en) * \text{solve}(R) * en$  and the natural logarithm of the determinant of  $R$ .  $t(.)$  denotes the transpose of a vector,  $en = (1,1,\dots,1)$  and  $R$  is the autocorrelation matrix.

### Author(s)

Hristos Tyrallis

### References

Golub GH, Van Loan CF (1996) *Matrix Computations*. Baltimore: John Hopkins University Press.

### Examples

```
# Estimate the parameters for the Nile time series.

r <- acfHKp(H = 0.8, maxlag = length(Nile)-1)

examp <- ltza(r, Nile)

# Comparison of the algorithm with typical approaches

examp[1] - as.numeric(t(Nile) %*% solve(toeplitz(r)) %*% Nile)

examp[2] - as.numeric(t(rep(1, length(r))) %*% solve(toeplitz(r)) %*% Nile)

examp[3] - as.numeric(t(rep(1, length(r))) %*% solve(toeplitz(r)) %*%
  rep(1,length(r)))
```

```
examp[4] - log(det(toeplitz(r)))
```

---

ltzb	<i>Value of quadratic forms for the inverse of a symmetric positive definite autocorrelation matrix.</i>
------	--

---

### Description

The function `ltzb` is used to calculate the value of quadratic forms for the inverse of a symmetric positive definite autocorrelation matrix, using the Levinson algorithm (Golub and Van Loan 1996, Algorithm 4.7.2).

### Usage

```
ltzb(r, x)
```

### Arguments

<code>r</code>	autocorelation vector
<code>x</code>	time series data

### Value

Vector with values  $t(en) * solve(R) * x$  and  $t(en) * solve(R) * en$ .  $t(.)$  denotes the transpose of a vector,  $en = (1,1,\dots,1)$  and  $R$  is the autocorrelation matrix.

### Author(s)

Hristos Tyrallis

### References

Golub GH, Van Loan CF (1996) *Matrix Computations*. Baltimore: John Hopkins University Press.

### Examples

```
# Estimate the parameters for the Nile time series.

r <- acfHKp(H = 0.8,maxlag = length(Nile)-1)

examp <- ltzb(r,Nile)
# Comparison of the algorithm with typical approaches

examp[1] - as.numeric(t(rep(1, length(r))) %*% solve(toeplitz(r)) %*% Nile)

examp[2] - as.numeric(t(rep(1, length(r))) %*% solve(toeplitz(r)) %*%
  rep(1,length(r)))
```

---

ltzc	<i>Value of quadratic forms for the inverse of a symmetric positive definite autocorrelation matrix.</i>
------	--

---

### Description

The function `ltzc` is used to calculate the value of quadratic forms for the inverse of a symmetric positive definite autocorrelation matrix, using the Levinson algorithm (Golub and Van Loan 1996, Algorithm 4.7.2).

### Usage

```
ltzc(r, x)
```

### Arguments

<code>r</code>	autocorelation vector
<code>x</code>	time series data

### Value

Vector with values  $t(x) * \text{solve}(R) * x$  and the natural logarithm of the determinant of  $R$ .  $t(.)$  denotes the transpose of a vector and  $R$  is the autocorrelation matrix.

### Author(s)

Hristos Tyrallis

### References

Golub GH, Van Loan CF (1996) *Matrix Computations*. Baltimore: John Hopkins University Press.

### Examples

```
# Estimate the parameters for the Nile time series.

r <- acfHKp(H = 0.8, maxlag = length(Nile)-1)

examp <- ltzc(r, Nile)

# Comparison of the algorithm with typical approaches

examp[1] - as.numeric(t(Nile) %*% solve(toeplitz(r)) %*% Nile)

examp[2] - log(det(toeplitz(r)))
```

---

ltzd	<i>Value of quadratic forms for the inverse of a symmetric positive definite autocorrelation matrix.</i>
------	--

---

**Description**

The function `ltzd` is used to calculate the value of quadratic forms for the inverse of a symmetric positive definite autocorrelation matrix, using the Levinson algorithm (Golub and Van Loan 1996, Algorithm 4.7.2).

**Usage**

```
ltzd(r, x)
```

**Arguments**

<code>r</code>	autocorelation vector
<code>x</code>	time series data

**Value**

A numeric value  $t(x) * \text{solve}(R) * x$ .  $t(\cdot)$  denotes the transpose of a vector and  $R$  is the autocorrelation matrix.

**Author(s)**

Hristos Tyrallis

**References**

Golub GH, Van Loan CF (1996) *Matrix Computations*. Baltimore: John Hopkins University Press.

**Examples**

```
# Estimate the parameters for the Nile time series.  
  
r <- acfHKp(H = 0.8, maxlag = length(Nile) - 1)  
  
examp <- ltzd(r, Nile)  
  
# Comparison of the algorithm with typical approaches  
  
examp - as.numeric(t(Nile) %*% solve(toeplitz(r)) %*% Nile)
```

---

MannKendallLTP      *Mann-Kendall trend test under the scaling hypothesis.*

---

### Description

The function MannKendallLTP applies the Mann-Kendall test under the scaling hypothesis for the data (Hamed 2008).

### Usage

MannKendallLTP(data)

### Arguments

data                  time series data

### Value

A list with three components.

Mann\_Kendall      Kendall's tau statistic, score, variance of score, Sen's slope, denominator D where  $\tau=S/D$  and p-value for the Mann-Kendall test

Significance\_of\_H      H estimate (eq.21, Hamed 2008) of the modified variables and p-value

Mann\_Kendall\_LTP      Variance of score (p.356, Hamed 2008) and p-value for the Mann-Kendall test under the scaling hypothesis

### Note

The functions score.c, score0.c and VstarSfunction.c are called from the C library of the package. The estimator of H for the stochastic process in eq(18) (Hamed 2008) is the ML estimator in Tyrallis and Koutsoyiannis (2011). The denominator for the Mann-Kendall test is calculated according to eq(23.3.4) in Hipel and McLeod (1994). The Mann-Kendall and modified Mann-Kendall test's hypotheses are  $H_0$ : no trend vs  $H_1$ : trend is present. The H test's hypotheses are  $H_0$ : H is not significant vs  $H_1$ : H is significant.

### Author(s)

Hristos Tyrallis

### References

Hamed KH (2008) Trend detection in hydrologic data: The Mann-Kendall trend test under the scaling hypothesis. *Journal of Hydrology* **349(3–4)**:350–363. doi:10.1016/j.jhydrol.2007.11.009.

Hipel KW, McLeod AI (1994) *Time series modelling of water resources and environmental systems*. Amsterdam: Elsevier.



Tyralis H, Koutsoyiannis D (2011) Simultaneous estimation of the parameters of the Hurst-Kolmogorov stochastic process. *Stochastic Environmental Research & Risk Assessment* **25(1)**:21–33. doi:10.1007/s004770100408x.

## Examples

```
# Modified Mann-Kendall test for the Nile time series.  
  
MannKendallLTP(Nile)
```

---

mlear1	<i>Maximum likelihood estimation for the AR(1) parameters.</i>
--------	--

---

## Description

The function `mlear1` is used to estimate the  $\mu$ ,  $\sigma$  and  $\phi$  parameters of the AR(1) process as defined in Tyralis and Koutsoyiannis (2014). The method for their estimation is described in eqs.8-9 (Tyralis and Koutsoyiannis 2011).

## Usage

```
mlear1(data, interval = c(-0.9999, 0.9999), tol = .Machine$double.eps^0.25)
```

## Arguments

<code>data</code>	time series data
<code>interval</code>	$\phi$ interval estimation
<code>tol</code>	estimation error tolerance

## Value

A vector whose values are the maximum likelihood estimates of  $\mu$ ,  $\sigma$  and  $\phi$ .

## Note

The function `likelihoodfunction.c` is called from the C library of the package. Ideas for creating this function came from McLeod et al. (2007).

## Author(s)

Hristos Tyralis

## References

- McLeod AI, Yu H, Krougly ZL (2007) Algorithms for linear time series analysis: With R package. *Journal of Statistical Software* **23(5)**:1–26. doi:10.18637/jss.v023.i05.
- Tyralis H, Koutsoyiannis D (2011) Simultaneous estimation of the parameters of the Hurst-Kolmogorov stochastic process. *Stochastic Environmental Research & Risk Assessment* **25(1)**:21–33. doi:10.1007/s004770100408x.
- Tyralis H, Koutsoyiannis D (2014) A Bayesian statistical model for deriving the predictive distribution of hydroclimatic variables. *Climate Dynamics* **42(11-12)**:2867–2883. doi:10.1007/s00382-0131804y.

## Examples

```
# Estimate the parameters for the Nile time series.

mlear1(Nile)
```

---

mleHK

*Maximum likelihood estimation for the HKp parameters.*


---

## Description

The function mleHK is used to estimate the  $\mu$ ,  $\sigma$  and H parameters of the HKp. The method for their estimation is described in eqs.8-9 (Tyralis and Koutsoyiannis 2011).

## Usage

```
mleHK(data, interval = c(0.0001, 0.9999), tol = .Machine$double.eps^0.25)
```

## Arguments

data	time series data
interval	H interval estimation
tol	estimation error tolerance

## Value

Vector of maximum likelihood estimates of  $\mu$ ,  $\sigma$  and H.

## Note

The function likelihoodfunction.c is called from the C library of the package. Ideas for creating this function came from McLeod et al. (2007).

## Author(s)

Hristos Tyralis

**References**

McLeod AI, Yu H, Krougly ZL (2007) Algorithms for linear time series analysis: With R package. *Journal of Statistical Software* **23(5)**:1–26. doi:[10.18637/jss.v023.i05](https://doi.org/10.18637/jss.v023.i05).

Tyralis H, Koutsoyiannis D (2011) Simultaneous estimation of the parameters of the Hurst-Kolmogorov stochastic process. *Stochastic Environmental Research & Risk Assessment* **25(1)**:21–33. doi:[10.1007/s004770100408x](https://doi.org/10.1007/s004770100408x).

**Examples**

```
# Estimate the parameters for the Nile time series.
```

```
mleHK(Nile)
```

# Index

## \* **array**

ltza, 12  
ltzb, 13  
ltzc, 14  
ltzd, 15

## \* **htest**

MannKendallLTP, 16

## \* **models**

inferf, 3  
inferfmetrop, 4  
inferH, 5  
inferHmetrop, 6  
infermsfmetrop, 8  
infermsmetrop, 9

## \* **ts**

acfHKp, 2  
lssd, 10  
lsv, 11  
MannKendallLTP, 16  
mlear1, 17  
mleHK, 18

acfHKp, 2

inferf, 3  
inferfmetrop, 4  
inferH, 5  
inferHmetrop, 6  
infermsfmetrop, 8  
infermsmetrop, 9

lssd, 10  
lsv, 11  
ltza, 12  
ltzb, 13  
ltzc, 14  
ltzd, 15

MannKendallLTP, 16  
mlear1, 17  
mleHK, 18