Package ‘tseriesChaos’

January 7, 2019

Title Analysis of Nonlinear Time Series
Date 2013-04-29
Version 0.1-13.1
Author Antonio, Fabio Di Narzo
Depends R (>= 2.2.0)
Imports deSolve
Suggests scatterplot3d
LazyData yes
LazyLoad yes
Description Routines for the analysis of nonlinear time series. This work is largely inspired by the TISEAN project, by Rainer Hegger, Holger Kantz and Thomas Schreiber: <http://www.mpipks-dresden.mpg.de/~tisean/>.
Maintainer Antonio Fabio Di Narzo <antonio.fabio@gmail.com>
License GPL-2
Repository CRAN
Date/Publication 2019-01-07 12:21:28 UTC
NeedsCompilation yes

R topics documented:

C2 ................................................................. 2
d2 ............................................................... 3
duffing.syst .................................................. 4
embed ......................................................... 4
false.nearest ................................................ 5
lorenz.syst .................................................. 6
lorenz.ts ..................................................... 6
Lyapunov exponent ....................................... 7
mutual ......................................................... 8
plot.ami ...................................................... 9
Sample correlation integral

Description
Sample correlation integral for the specified length scale

Usage
C2(series, m, d, t, eps)

Arguments
- series: time series
- m: embedding dimension
- d: time delay
- t: Theiler window
- eps: length scale

Details
Computes the sample correlation integral on the provided time series for the specified length scale, and considering a time window t (see references). It uses a naive algorithm: simply returns the fraction of points pairs nearer than eps. Normally, you would use d2, which takes roughly the same time, but computes the correlation sum for multiple length scales and embedding dimensions at once.

Value
The sample correlation integral at eps length scale.

Author(s)
Antonio, Fabio Di Narzo
References

Hegger, R., Kantz, H., Schreiber, T., Practical implementation of nonlinear time series methods: The TISEAN package; CHAOS 9, 413-435 (1999)

See Also
d2

d2 Sample correlation integral (at multiple length scales)

Description

Computes the sample correlation integral over a grid of neps length scales starting from eps.min, and for multiple embedding dimensions

Usage
d2(series, m, d, t, eps.min, neps=100)

Arguments

series time series
m max embedding dimension
d time delay
t Theiler window
eps.min min length scale
neps number of length scales to evaluate

Details

Computes the sample correlation integral over neps length scales starting from eps.min, for embedding dimension 1,...,m, considering a t time window (see references). The slope of the linear segment in the log-log plot gives an estimate of the correlation dimension (see the example).

Value

Matrix. Column 1: length scales. Column i=2,..., m+1: sample correlation integral for embedding dimension i-1.

Author(s)

Antonio, Fabio Di Narzo
References
Hegger, R., Kantz, H., Schreiber, T., Practical implementation of nonlinear time series methods: The TISEAN package; CHAOS 9, 413-435 (1999)

Examples
d2(lorenz.ts, m=6, d=2, t=4, eps.min=2)

<table>
<thead>
<tr>
<th>duffing.syst</th>
<th>Duffing oscillator</th>
</tr>
</thead>
</table>

Description
Duffing oscillator system, to be used with sim.cont

Details
To be used with sim.cont

Author(s)
Antonio, Fabio Di Narzo

<table>
<thead>
<tr>
<th>embedd</th>
<th>Embedding of a time series</th>
</tr>
</thead>
</table>

Description
Embedding of a time series with provided time delay and embedding dimension parameters.

Usage
embedd(x, m, d, lags)

Arguments
x time series
m embedding dimension (if lags missed)
d time delay (if lags missed)
lags vector of lags (if m and d are missed)

Details
Embedding of a time series with provided delay and dimension parameters.
false.nearest

Value

Matrix with columns corresponding to lagged time series.

Author(s)


Examples

```r
library(scatterplot3d)
x <- window(rossler.ts, start=90)
xyz <- embed(x, m=3, d=8)
scatterplot3d(xyz, type="l")

## embedding multivariate time series
series <- cbind(seq(1,50), seq(101,150))
head(embed(series, m=6, d=1))
```

false.nearest  Method of false nearest neighbours

Description

Method of false nearest neighbours to help deciding the optimal embedding dimension

Usage

```r
false.nearest(series, m, d, t, rt=10, eps=sd(series)/10)
```

Arguments

- `series`: time series
- `m`: maximum embedding dimension
- `d`: delay parameter
- `t`: Theiler window
- `rt`: escape factor
- `eps`: neighborhood diameter

Details

Method of false nearest neighbours to help deciding the optimal embedding dimension.

Value

Fraction of false neighbors (first row) and total number of neighbors (second row) for each specified embedding dimension (columns)
Author(s)
Antonio, Fabio Di Narzo

References
Hegger, R., Kantz, H., Schreiber, T., Practical implementation of nonlinear time series methods: The TISEAN package; CHAOS 9, 413-435 (1999)

Examples
(fn.out <- false.nearest(rossler.ts, m=6, d=8, t=180, eps=1, rt=3))
plot(fn.out)

Description
Lorenz system, to be used with sim.cont

Details
To be used with sim.cont

Author(s)
Antonio, Fabio Di Narzo

Description
Lorenz simulated time series, without noise. Of each state of the system, we observe the euclidean norm.

Details
Lorenz simulated time series, without noise, obtained with the call: lorenz.ts <- sim.cont(lorenz.syst, 0, 100, 0.05)

Author(s)
Antonio, Fabio Di Narzo
Lyapunov exponent

Tools to evaluate the maximal Lyapunov exponent of a dynamic system

Description

Tools to evaluate the maximal Lyapunov exponent of a dynamic system from a univariate time series

Usage

\texttt{lyap}_k(\text{series}, \text{m}, \text{d}, \text{t}, \text{k=1}, \text{ref}, \text{s}, \text{eps})

\texttt{lyap}\texttt{(dists, start, end)}

Arguments

- \texttt{series} \quad \text{time series}
- \texttt{m} \quad \text{embedding dimension}
- \texttt{d} \quad \text{time delay}
- \texttt{k} \quad \text{number of considered neighbours}
- \texttt{eps} \quad \text{radius where to find nearest neighbours}
- \texttt{s} \quad \text{iterations along which follow the neighbours of each point}
- \texttt{ref} \quad \text{number of points to take into account}
- \texttt{t} \quad \text{Theiler window}
- \texttt{dists} \quad \text{Should be the output of a call to \texttt{lyap}_k (see the example)}
- \texttt{start} \quad \text{Starting time of the linear bite of \texttt{dists}}
- \texttt{end} \quad \text{Ending time of the linear bite of \texttt{dists}}

Details

The function \texttt{lyap}_k estimates the largest Lyapunov exponent of a given scalar time series using the algorithm of Kantz.

The function \texttt{lyap} computes the regression coefficients of a user specified segment of the sequence given as input.

Value

\texttt{lyap}_k\text{ gives the logarithm of the stretching factor in time.}

\texttt{lyap}\text{ gives the regression coefficients of the specified input sequence.}

Author(s)

Antonio, Fabio Di Narzo
References

Hegger, R., Kantz, H., Schreiber, T., Practical implementation of nonlinear time series methods: The TISEAN package; CHAOS 9, 413-435 (1999)


See Also

`mutual`, `false.nearest` for the choice of optimal embedding parameters. `embed` to perform embedding.

Examples

```r
output <- lyap_k(lorenz.ts[, 1], m=3, d=2, s=200, t=40, ref=1700, k=2, eps=4)
plot(output)
lyap(output, 0.73, 2.47)
```

---

**Description**

Estimates the average mutual information index (ami) of a given time series for a specified number of lags

**Usage**

`mutual(series, partitions = 16, lag.max = 20, plot=TRUE, ...)`

**Arguments**

- `series`: time series
- `partitions`: number of bins
- `lag.max`: largest lag
- `plot`: logical. If 'TRUE' (the default) the ami is plotted
- `...`: further arguments to be passed to the plot method

**Details**

Estimates the mutual information index for a specified number of lags. The joint probability distribution function is estimated with a simple bi-dimensional density histogram.

**Value**

An object of class "ami", which is a vector containing the estimated mutual information index for each lag between 0 and `lag.max`. 
Author(s)
Antonio, Fabio Di Narzo

References
Hegger, R., Kantz, H., Schreiber, T., Practical implementation of nonlinear time series methods: The TISEAN package; CHAOS 9, 413-435 (1999)

Examples
mutual(lorenz.ts)

plot.ami  
Plotting average mutual information index

Description
Plotting method for objects inheriting from class "ami".

Usage
## S3 method for class 'ami'
plot(x, main = NULL, ...)

Arguments
x  
"ami" object

main, ...  
additional graphical arguments

Details
Plots the ami for each lag in x.

Author(s)
Antonio, Fabio Di Narzo

See Also
mutual
plot.d2  

Plotting sample correlation integrals

Description
Plotting method for objects inheriting from class "d2".

Usage
```r
## S3 method for class 'd2'
plot(x, ...)
```

Arguments
- `x` "d2" object
- `...` additional graphical arguments

Details
Plots the sample correlation integrals in `x` in log-log scale, as a line for each considered embedding dimension.

Author(s)
Antonio, Fabio Di Narzo

See Also
d2

plot.false.nearest  

Plotting false nearest neighbours results

Description
Plotting method for objects inheriting from class "false.nearest".

Usage
```r
## S3 method for class 'false.nearest'
plot(x, ...)
```

Arguments
- `x` "false.nearest" object
- `...` additional graphical arguments
Details

Plots the results of `false.nearest`.

Author(s)

Antonio, Fabio Di Narzo

See Also

`false.nearest`

Description

Printing method for objects inheriting from class "d2".

Usage

```r
## S3 method for class 'd2'
print(x, ...)
```

Arguments

- `x` : "d2" object
- `...` : additional arguments to 'print'

Details

Simply calls `plot.d2`.

Author(s)

Antonio, Fabio Di Narzo

See Also

`plot.d2, d2`
print.false.nearest  Printing false nearest neighbours results

Description

Printing method for objects inheriting from class "false.nearest".

Usage

## S3 method for class 'false.nearest'
print(x, ...)

Arguments

x  "false.nearest" object

...  additional arguments to 'print'

Details

Prints the table of results of false.nearest.

Author(s)

Antonio, Fabio Di Narzo

See Also

plot.false.nearest, false.nearest

recurr  Recurrence plot

Description

Recurrence plot

Usage

recurr(series, m, d, start.time=start(series), end.time=end(series), ...)

recurr
Arguments

series          time series
m              embedding dimension
d              time delay
start.time      starting time window (in time units)
end.time        ending time window (in time units)
...             further parameters to be passed to filled.contour

Details

Produces the recurrence plot, as proposed by Eckmann et al. (1987). White is maximum distance, black is minimum.

warning

Be aware that number of distances to store goes as $n^2$, where $n = \text{length}(\text{window}(\text{series}, \text{start}=\text{start.time}, \text{end}=\text{end.time}))$.

Author(s)

Antonio, Fabio Di Narzo

References


Examples

recurr(lorenz.ts, m=3, d=2, start.time=15, end.time=20)

Description

Roessler system of equations

Details

To be used with sim.cont.

Author(s)

Antonio, Fabio Di Narzo
Description

Roessler simulated time series, without noise. Of each state of the system, we observe the first component.

Details

Roessler simulated time series, without noise, obtained with the call:
```r
rossler.ts <- sim.cont(rossler.syst, start=0, end=650, dt=0.1, start.x=c(0,0,0), parms=c(0.15, 0.2, 10))
```

Author(s)

Antonio, Fabio Di Narzo

---

Description

Simulates a continuous dynamic system

Usage

```r
sim.cont(syst, start.time, end.time, dt, start.x, params=NULL, obs.fun=function(x) x[1])
```

Arguments

- `syst` : ODE system
- `start.time` : starting time
- `end.time` : ending time
- `dt` : time between observations
- `start.x` : initial conditions
- `params` : parameters for the system
- `obs.fun` : observed function of the state

Details

Simulates a dynamic system of provided ODEs. Uses `lsoda` in `odesolve` for numerical integration of the system.
**Value**

The time series of the observed function of the system’s state.

**Author(s)**

Antonio, Fabio Di Narzo.

**See Also**

lorenz.syst, rossler.syst, duffing.syst

**Examples**

```r
rossler.ts <- sim.cont(rossler.syst, start=0, end=650, dt=0.1,
start.x=c(0,0,0), parms=c(0.15, 0.2, 10))
```

**Description**

Space-time separation plot

**Usage**

```r
stplot(series, m, d, idt=1, mdt)
```

**Arguments**

- `series`: time series
- `m`: embedding dimension
- `d`: time delay
- `idt`: observation steps in each iteration
- `mdt`: number of iterations

**Details**

Produces the space-time separation plot, as introduced by Provenzale et al. (1992), which can be used to decide the Theiler time window \( t \), which is required in many other algorithms in this package.

It plots the probability that two points in the reconstructed phase-space have distance smaller than epsilon in function of epsilon and of the time \( t \) between the points, as iso-lines at levels 10%, 20%, ..., 100%.

**Value**

lines of constant probability at 10%, 20%, ..., 100%.
Author(s)
Antonio, Fabio Di Narzo

References

See Also
false.nearest, d2, lyap_k

Examples
stplot(rossler.ts, m=3, d=8, idt=1, mdt=250)
Index

*Topic datagen
  duffing.syst, 4
  lorenz.syst, 6
  rossler.syst, 13
  sim.cont, 14
*Topic datasets
  lorenz.ts, 6
  rossler.ts, 14
*Topic hplot
  plot.ami, 9
  plot.d2, 10
  plot.false.nearest, 10
*Topic manip
  embedd, 4
*Topic ts
  C2, 2
  d2, 3
  false.nearest, 5
  lyapunov exponent, 7
  mutual, 8
  print.d2, 11
  print.false.nearest, 12
  recurr, 12
  stplot, 15

C2, 2

d2, 2, 3, 3, 10, 11, 16
duffing.syst, 4, 15

embedd, 4, 8

false.nearest, 5, 8, 11, 12, 16

lorenz.syst, 6, 15
lorenz.ts, 6
lyap(Lyapunov exponent), 7
lyap_k, 16
lyap_k(Lyapunov exponent), 7
Lyapunov exponent, 7

mutual, 8, 8, 9

plot.ami, 9
plot.d2, 10, 11
plot.false.nearest, 10, 12
print.d2, 11
print.false.nearest, 12

recurr, 12
rossler.syst, 13, 15
rossler.ts, 14

sim.cont, 4, 6, 13, 14
stplot, 15