Package ‘TransferEntropy’

April 26, 2016

Type Package

Title The Transfer Entropy Package

Version 1.4

Date 2016-01-01

Author ANN Library: David Mount, Sunil Arya (see src/ann_1.1.2/Copyright.txt).
Transfer Entropy Packge: Ghazaleh Haratinezhad Torbati, Glenn Lawyer.

Maintainer Ghazaleh Haratinezhad Torbati <ghazale.hnt@gmail.com>

Description Estimates the transfer entropy from one time series to another, where each time series consists of continuous random variables. The transfer entropy is an extension of mutual information which takes into account the direction of information flow, under the assumption that the underlying processes can be described by a Markov model. Two estimation methods are provided. The first calculates transfer entropy as the difference of mutual information. Mutual information is estimated using the Kraskov method, which builds on a nearest-neighbor framework (see package references). The second estimation method estimate transfer entropy via the a generalized correlation sum.

License GPL (>= 2) | file LICENSE

Depends R (>= 3.0.0)

Imports Rcpp (>= 0.11.6)

LinkingTo Rcpp, BH

URL https://github.com/Healthcast/TransEnt

NeedsCompilation yes

Repository CRAN

Date/Publication 2016-04-26 16:50:27

R topics documented:

TransferEntropy-package *The Transfer Entropy Package*

Description

Estimates the transfer entropy from one time series to another, where each time series consists of continuous random variables. The transfer entropy is an extension of mutual information which takes into account the direction of information flow, under the assumption that the underlying processes can be described by a Markov model. Two estimation methods are provided. The first calculates transfer entropy as the difference of mutual information. Mutual information is estimated using the Kraskov method, which builds on a nearest-neighbor framework (see package references). The second estimation method estimate transfer entropy via the a generalized correlation sum. The main function `computeTE` calculates the transfer entropy from one continuous-valued random process to another. Two calculaton methods are offered.

Trans Entrop is an information theoric measure that quantifies the statistical coherence between systems evolving in time. Unlike mutual information, transfer entropy is more adequate for indicating the direction of the information flow.

$$T_{X->Y} = \Sigma p(Y_{n+1}, Y_n^{(k)}, X_n^{(l)}) \log \frac{p(Y_{n+1}|Y_n^{(k)}, X_n^{(l)})}{p(Y_{n+1}|Y_n^{(k)})}$$

TE can be computed in a straightforward way for discrete valued random processes by counting. This does not directly generalize to continuous valued variables as continuous random processes seldom revisit prior values. Naive binning may distort the estimate of the underlying probability distribution. Hence the development of more statistically robust estimation methods such as the generalized correlation sum or the Kraskov estimator, which are used here.

Please see the references for additional background on the derivation and intepretation of the transfer entropy and the issues of estimating ratios of probability distributions from observations of continuously distributed random variables.

Author(s)

ANN Library: David Mount, Sunil Arya (see src/ann_1.1.2/Copyright.txt). Transfer Entropy Packge: Ghazaleh Haratinezhad Torbati, Glenn Lawyer. Maintainer: Ghazaleh Haratinezhad Torbati <ghazale.hnt@gmail.com>

References

Measuring Information Transfer. Thomas Schreiber. 2000.

Information Transfer in Continuous Processes. A. Kaiser, T. Schreiber. 2002.

Estimating Mutual Information. Alexander Kraskov, Harald Stoegbauer, Peter Grassberger. 2004.

A Recipe for the Estimation of Information Flow in a Dynamical System. Deniz Gencaga, Kevin H. Knuth, William B. Rossow. 2015.

computeTE *Estimate Transfer Entropy.*

Description

`ComputeTE` Estimate the Transfer Entropy (TE) from one continuous-valued random process to a second process.

Usage

```
computeTE(X, Y, embedding, k, method = "MI_diff", epsDistace = -1,
  safetyCheck = FALSE)
```

Arguments

`X`	Numeric vector, Transfer Entropy is calculated to random process X
`Y`	Numeric vector, Transfer Entropy is calculated from random process Y
`embedding`	Numeric, The embedding dimension. Must be positive integer
`k`	Numeric, The k'th neighbor used by the Kraskov estimator. Must be positive integer. Kraskov suggests a value in (1,3)
`method`	String, The method to be used to estimate TE from ("MI_dif","Correlation")
`epsDistace`	Numeric, The distance used for measuring TE in Correlation method, by default it is the average distance calculated in XKY
`safetyCheck`	Logical, For computing TE using "mi_diff" method the data need to be noisy otherwise a crach might happen. This parameter can check if there are any idetical points in the spaces made for this use

Details

A function to calculate Transfer Entropy from random process Y to random process X. The TE, introduced by Schreiber in 2000, extends the concept of mutual information to provide a direction-sensitive measure of information flow between two time series. Formally, the transfer entropy from time series Y to X is given by $T_{Y \to X} = \sum p(x_{n+1}, x_n^{(k)}, y_n^{(l)}) log \frac{p(x_{n+1}|x_n^{(k)}, y_n^{(l)})}{p(x_{n+1}|x_n^{(k)})}$ where x_{n+1} is the value of X at time $n+1$, and $x_n^{(k)}$ ($y_n^{(l)}$) is the k (l) lagged values of X (Y) at time n. The definition of TE assumes X is an Markov process. The `embedding` dimension should be chosen to match the delay of the Markov process. The TE measures the additional amount of information Y contains about X over the information contained in the Markov embedding. Two methods for estimating TE are provided. The first is based on the mutual information distance $MI(X_{i+1}|X^{(e)}, Y_i) - MI(X_{i+1}|X^{(e)})$, where e is the embedding dimension. This approach follows directly from the definition of the TE. Mutual information is estimated using the k-nearest neighbor approach suggested by Krasvok. The second method is based on the generalized correlation sum.

Things can go wrong in several ways. First, the random processes must meet the assumption of the TE. That is, X must represent some form of Markov process whose probability distribution may also be influenced by Y. A more subtle error can occur when multiple points in $X^(k)$ (or some of its subspaces) have identical coordinates. This can lead to several points which have identical distance to a query point, which violates the assumptions of the Kraskov estimator, causing it to throw an error. The solution in this case is to add some small noise to the measurements X prior to computing TE.

Value

Numeric, The estimated transfer entropy

Examples

```
## Intitialize two vectors of length 10001
X <- rep(0,10000+1)
Y <- rep(0,10000+1)
## Create two linked random processes. Y is independent of X,
## while X is determined in part by the previous values of Y.
for(i in 1:10000){
 Y[i+1] <- 0.6*Y[i] + rnorm(1)
 X[i+1] <- 0.4*X[i] + 0.6*Y[i] + rnorm(1)
}
X <- X[101:10000]
Y <- Y[101:10000]
## Compute the TE from Y to X
computeTE(X,Y,3,1,"MI_diff")  ## should be circa 0.16
## and from X to Y
computeTE(Y,X,3,1,"MI_diff")  ## should be circa zero
computeTE(X,Y,3,1,"Correlation",0.4)
computeTE(Y,X,3,1,"Correlation",0.4)
```

Index