Package: knnmi (via r-universe)

September 1, 2024

Title k-Nearest Neighbor Mutual Information Estimator

Version 1.0

Description This is a 'C++' mutual information (MI) library based on the k-nearest neighbor (KNN) algorithm. There are three functions provided for computing MI for continuous values, mixed continuous and discrete values, and conditional MI for continuous values. They are based on algorithms by A. Kraskov, et. al. (2004) <doi:10.1103/PhysRevE.69.066138>, BC Ross (2014)<doi:10.1371/journal.pone.0087357>, and A. Tsimpiris (2012) <doi:10.1016/j.eswa.2012.05.014>, respectively.

License GPL (>= 3)

Depends R (>= 4.1.0)

Suggests spelling, testthat (>= 3.0.0)

Config/testthat/edition 3

Encoding UTF-8

LazyData true

RoxygenNote 7.3.1

Language en-US

NeedsCompilation yes

Author Brian Gregor [aut, cre] (<https://orcid.org/0009-0004-9640-0574>), Katia Bulekova [aut] (<https://orcid.org/0000-0003-1560-2146>), Reina Chau [aut] (<https://orcid.org/0000-0003-3012-1404>), Stefano Monti [aut] (<https://orcid.org/0000-0002-9376-0660>), Benoit Jacob [cph] (Author of included Eigen library), Gael Guennebaud [cph] (Author of included Eigen library), Jose Luis Blanco [cph] (Author of included nanoflann library), Pranjal Kumar Rai [cph] (Author of included nanoflann library)

Maintainer Brian Gregor <bgregor@bu.edu>

Date/Publication 2024-04-02 12:32:06 UTC

Repository https://bu-bgregor.r-universe.dev

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

RemoteRef HEAD

RemoteSha 8e572ccbea79fff77ad827b897c53968df00d26c

Contents

`cond_mutual_inf` *Conditional mutual information estimation*

Description

Conditional mutual information estimation CMI(X;Y|Z) where X is a continuous vector. The input Y and conditional input Z can be vectors or matrices. If Y and Z are discrete then they must be numeric or integer valued.

Usage

```
cond_mutual_inf(X, Y, Z, k = 3L)
```

Arguments

`X`	input vector.
`Y`	input vector or matrix.
`Z`	conditional input vector or matrix.
`k`	number of nearest neighbors.

Details

Argument Y is a vector of the same size as vector X, or a matrix whose column dimension matches the size of X. Argument Z is also a vector of the same size as vector X, or a matrix whose column dimension matches the size of X. If Y and Z are both matrices they must additionally have the same number of rows. If Y and/or Z are discrete values they must have a numeric or integer type.

Value

Returns the estimated conditional mutual information. The return value is a vector of size 1 if both Y and Z are vectors. If either Y or Z are matrices the return value is a vector whose size is the number of rows in the matrix.

References

Alkiviadis Tsimpiris, Ioannis Vlachos, Dimitris Kugiumtzis, Nearest neighbor estimate of conditional mutual information in feature selection, Expert Systems with Applications, Volume 39, Issue 16, 2012, Pages 12697-12708 doi:10.1016/j.eswa.2012.05.014

Examples

```
data(mutual_info_df)
set.seed(654321)
cond_mutual_inf(mutual_info_df$Zc_XcYc,
                       mutual_info_df$Xc, t(mutual_info_df$Yc))

M <- cbind(mutual_info_df$Xc, mutual_info_df$Yc)
ZM <- cbind(mutual_info_df$Yc, mutual_info_df$Wc)
cond_mutual_inf(mutual_info_df$Zc_XcYcWc, t(M), t(ZM))
```

`mutual_info_df` *Toy Dataset for knnmi package*

Description

Toy Dataset for knnmi package

Usage

```
data(mutual_info_df)
```

Format

A data frame with 100 rows and 10 columns

`mutual_inf_cc` *Mutual information estimation*

Description

Estimate the mutual information MI(X;Y) of the target `X` and features `Y` where `X` and `Y` are both continuous using k-nearest neighbor distances.

Usage

```
mutual_inf_cc(target, features, k = 3L)
```

Arguments

`target`	input vector.
`features`	input vector or matrix.
`k`	Integer number of nearest neighbors. The default value is 3.

Details

The features argument is a vector of the same size as the target vector, or a matrix whose column dimension matches the size of the target vector.

Value

Returns the estimated mutual information. The return value is a vector of size 1 if the features argument is a vector. If the features argument is a matrix then the return value is a vector whose size matches the number of rows in the matrix.

References

Alexander Kraskov, Harald Stögbauer, and Peter Grassberger. Phys. Rev. E **69**, 066138 (2004). doi:10.1103/PhysRevE.69.066138

Examples

```
data(mutual_info_df)
set.seed(654321)
mutual_inf_cc(mutual_info_df$Yc, t(mutual_info_df$Zc_XcYc))
mutual_inf_cc(mutual_info_df$Xc, t(mutual_info_df$Zc_XcYc), k=5)
```

`mutual_inf_cd` *Mutual information estimation*

Description

Estimate the mutual information MI(X;Y) of the target `X` and features `Y` where `X` is continuous or discrete and `Y` is discrete using k-nearest neighbor distances.

Usage

```
mutual_inf_cd(target, features, k = 3L)
```

Arguments

`target`	input vector.
`features`	input vector or matrix.
`k`	Integer number of nearest neighbors. The default value is 3.

Details

The features argument is a vector of the same size as the target vector, or a matrix whose column dimension matches the size of the target vector. Discrete values for the features or targets must be numeric or integer types.

Value

Returns the estimated mutual information. The return value is a vector of size 1 if the features argument is a vector. If the features argument is a matrix then the return value is a vector whose size matches the number of rows in the matrix.

References

Ross BC (2014) Mutual Information between Discrete and Continuous Data Sets. PLoS ONE 9(2): e87357. doi:10.1371/journal.pone.0087357

Examples

```
data(mutual_info_df)
set.seed(654321)
mutual_inf_cd(mutual_info_df$Zc_XdYd, t(mutual_info_df$Xd))

M <- cbind(mutual_info_df$Xd, mutual_info_df$Yd)
mutual_inf_cd(mutual_info_df$Zc_XdYdWd, t(M))
```

Index