Package: fastmit (via r-universe)

October 27, 2024

Type Package

Title Fast Mutual Information Based Independence Test

Version 0.1.1

Description A mutual information estimator based on k-nearest neighbor
method proposed by A. Kraskov, et al. (2004)
<doi:10.1103/PhysRevE.69.066138> to measure general dependence
and the time complexity for our estimator is only squared to
the sample size, which is faster than other statistics.
Besides, an implementation of mutual information based
independence test is provided for analyzing multivariate data
in Euclidean space (T B. Berrett, et al. (2019)
<doi:10.1093/biomet/asz024>); furthermore, we extend it to
tackle datasets in metric spaces.

License GPL (>= 2)

Encoding UTF-8

LazyData true

LinkingTo Rcpp, RcppArmadillo

Imports Rcpp, stats

Suggests testthat

NeedsCompilation yes

Author Shiyun Lin [aut, cre], Jin Zhu [aut], Wenliang Pan [aut],
Xueqin Wang [aut], SC2S2 [cph]

Maintainer Shiyun Lin <linshy27@mail2.sysu.edu.cn>

RoxygenNote 6.1.1

Repository CRAN

Date/Publication 2019-12-30 14:00:07 UTC

Contents

---

mi   *kNN Mutual Information Estimators*

Description

Estimate mutual information based on the distribution of nearest neighborhood distances. The kNN method is described by Kraskov, et. al (2004).

Usage

```
mi(x, y, k = 5, distance = FALSE)
```

Arguments

x	A numeric vector, matrix, data.frame or [dist](#) object.
y	A numeric vector, matrix, data.frame or [dist](#) object.
k	Order of neighborhood to be used in the kNN method.
distance	Bool flag for considering x and y as distance matrices or not. If distance = TRUE, x and y would be considered as distance matrices, otherwise, these arguments are treated as data and Euclidean distance would be implemented for the samples in x and y. Default: distance = FALSE.

Details

If two samples are passed to arguments x and y, the sample sizes (i.e. number of rows of the matrix or length of the vector) must agree. Moreover, data being passed to x and y must not contain missing or infinite values.

Value

mi	The estimated mutual information.

References

Kraskov, A., Stögbauer, H., & Grassberger, P. (2004). Estimating mutual information. Physical review E 69(6): 066138.

Examples

```
library(fastmit)
set.seed(1)
x <- rnorm(100)
y <- x + rnorm(100)
mi(x, y, k = 5, distance = FALSE)

set.seed(1)
x <- rnorm(100)
```

```
y <- 100 * x + rnorm(100)
distx <- dist(x)
disty <- dist(y)
mi(distx, disty, k = 5, distance = TRUE)
```

---

mi.test	*Mutual Information Test*

Description

Mutual Information test of independence. Mutual Information are generic dependence measures in Banach spaces.

Usage

```
mi.test(x, y, k = 5, distance = FALSE, num.permutations = 99,
  seed = 1)
```

Arguments

x   A numeric vector, matrix, data.frame or [dist](#) object.

y   A numeric vector, matrix, data.frame or [dist](#) object.

k   Order of neighborhood to be used in the kNN method.

distance   Bool flag for considering x and y as distance matrices or not. If distance = TRUE, x and y would be considered as distance matrices, otherwise, these arguments are treated as data and Euclidean distance would be implemented for the samples in x and y. Default: distance = FALSE.

num.permutations

            The number of permutation replications. If num.permutations = 0, the function just returns the Mutual Information statistic. Default: num.permutations = 99.

seed   The random seed. Default: seed = 1.

Details

If two samples are passed to arguments x and y, the sample sizes (i.e. number of rows of the matrix or length of the vector) must agree. Moreover, data being passed to x and y must not contain missing or infinite values.

mi.test utilizes the Mutual Information statistics (see [mi](#)) to measure dependence and derive a p-value via replicating the random permutation num.permutations times.

Value

If `num.permutations > 0`, `mi.test` returns a `htest` class object containing the following components:

`statistic`	Mutual Information statistic.
`p.value`	The p-value for the test.
`replicates`	Permutation replications of the test statistic.
`size`	Sample size.
`alternative`	A character string describes the alternative hypothesis.
`method`	A character string indicates what type of test was performed.
`data.name`	Description of data.

If `num.permutations = 0`, `mi.test` returns a statistic value.

Examples

```
library(fastmit)
set.seed(1)
error <- runif(50, min = -0.3, max = 0.3)
x <- runif(50, 0, 4*pi)
y <- cos(x) + error
# plot(x, y)
res <- mi.test(x, y)
```

Index