Package 'BalanceCheck'

March 18, 2019

Version 0.2

Date 2019-3-17

Title Balance Check for Multiple Covariates in Matched Observational
Studies

Author Hao Chen and Dylan Small

Maintainer Hao Chen <hxchen@ucdavis.edu>

Depends R (>= 3.0.1)

Description Two practical tests are provided for assessing whether multiple covariates in a treatment group and a matched control group are balanced in observational studies.

License GPL (>= 2)

Imports ade4, mvtnorm

NeedsCompilation no

Repository CRAN

Date/Publication 2019-03-18 05:14:14 UTC

R topics documented:

BalanceCheck	*Balance Check for Multiple Covariates in Matched Observational Studies*

Description

This package includes two tests for assessing whether multiple covariates in a treatment group and a matched control group are balanced in observational studies.

Author(s)

Hao Chen and Dylan Small

Maintainer: Hao Chen (hxchen@ucdavis.edu)

References

Chen, H. and Small, D. (2019) New multivariate tests for assessing covariate balance in matched observational studies.

See Also

CrossMST, CrossNN

CrossMST	*Covariate balance checking through the minimum spanning tree*

Description

This function tests whether covariates in a treatment group and a matched control group are balanced in observational studies through the minimum spanning tree constructed on the subjects.

Usage

```
CrossMST(distM,treated.index,perm=0,k=1,discrete.correction=TRUE)
```

Arguments

distM   The distance matrix for the pooled observations (pooled over the treated subjects and the matched controls). If there are n treated subjects and n matched controls, then this distance matrix is a 2n by 2n matrix with the [i,j] element the distance between observation i and observation j. What distance to use is decided by users. Some simple choices are the Euclidean distance, L1 distance, and mahalanobis distance.

treated.index   The subject indices of the treated subjects. The subjects are ordered in the same way as for calculating the distance matrix, distM.

perm	The number of permutations performed to calculate the p-value of the test. The default value is 0, which means the permutation is not performed and only approximate p-value based on asymptotic theory is provided. Doing permutation could be time consuming, so be cautious if you want to set this value to be larger than 10,000.
k	Set as positive integer values, indicates k-MST is used.
discrete.correction	
	When this is set as TRUE (recommended), a continuation correction is done for computing the asymptotic p-value to account for the discrete nature of the statistic.

Value

test.stat.Z	The standardized test statistic (ZR in the reference paper.
pval.appr	The approximated p-value based on asymptotic theory.
pval.perm	The permutation p-value when argument 'perm' is positive.

References

Chen, H. and Small, D. (2019) New multivariate tests for assessing covariate balance in matched observational studies.

See Also

CrossNN

Examples

```
## A snippet of the smoking example in the reference paper.
## smoking.rda contains a 300 by 300 distance matrix, smokingDist.
## The indices of the treated subjects are 1:150.
data(smoking)
CrossMST(smokingDist, 1:150)

## Uncomment the following line to get permutation p-value with 1,000 permutations.
# CrossMST(smokingDist, 1:150, perm=1000)
```

CrossNN	*Covariate balance checking through the nearest neighbor graph*

Description

This function tests whether covariates in a treatment group and a matched control group are balanced in observational studies through the nearest neighbor graph constructed on the subjects.

Usage

```
CrossNN(distM,treated.index,perm=0,k=1,discrete.correction=TRUE)
```

Arguments

distM
: The distance matrix for the pooled observations (pooled over the treated subjects and the matched controls). If there are n treated subjects and n matched controls, then this distance matrix is a 2n by 2n matrix with the [i,j] element the distance between observation i and observation j. What distance to use is decided by users. Some simple choices are the Euclidean distance, L1 distance, and mahalanobis distance.

treated.index
: The subject indices of the treated subjects. The subjects are ordered in the same way as for calculating the distance matrix, distM.

perm
: The number of permutations performed to calculate the p-value of the test. The default value is 0, which means the permutation is not performed and only approximate p-value based on asymptotic theory is provided. Doing permutation could be time consuming, so be cautious if you want to set this value to be larger than 10,000.

k
: Set as positive integer values, indicates k-NN is used.

discrete.correction
: When this is set as TRUE (recommended), a continuation correction is done for computing the asymptotic p-value to account for the discrete nature of the statistic.

Value

test.stat.Z
: The standardized test statistic (ZD in the reference paper.

pval.appr
: The approximated p-value based on asymptotic theory.

pval.perm
: The permutation p-value when argument 'perm' is positive.

See Also

CrossMST

Examples

```
## A snippet of the smoking example in the reference paper.
## smoking.rda contains a 300 by 300 distance matrix, smokingDist.
## The indices of the treated subjects are 1:150.
data(smoking)
CrossNN(smokingDist, 1:150)

## Uncomment the following line to get permutation p-value with 1,000 permutations.
# CrossNN(smokingDist, 1:150, perm=1000)
```

---

getD *Get intermediate results for CrossNN*

Description

This function calculates D11 and D22 for the CrossNN test.

Usage

```
getD(distM,treated.index)
```

Arguments

distM
: The distance matrix for the pooled observations (pooled over the treated subjects and the matched controls). If there are n treated subjects and n matched controls, then this distance matrix is a 2n by 2n matrix with the [i,j] element the distance between observation i and observation j. What distance to use is decided by users. Some simple choices are the Euclidean distance, L1 distance, and mahalanobis distance.

treated.index
: The subject indices of the treated subjects. The subjects are ordered in the same way as for calculating the distance matrix, distM.

See Also

CrossNN

---

getDk *Get intermediate results for CrossNN under k-NN*

Description

This function calculates D11 and D22 for the CrossNN test when k-NN (k>1) is used.

Usage

```
getDk(distM,treated.index,k)
```

Arguments

distM
: The distance matrix for the pooled observations (pooled over the treated subjects and the matched controls). If there are n treated subjects and n matched controls, then this distance matrix is a 2n by 2n matrix with the [i,j] element the distance between observation i and observation j. What distance to use is decided by users. Some simple choices are the Euclidean distance, L1 distance, and mahalanobis distance.

treated.index	The subject indices of the treated subjects. The subjects are ordered in the same way as for calculating the distance matrix, distM.
k	An integer larger than 1.

See Also

[CrossNN](#)

---

getR1R2	*Get intermediate results for CrossMST*

Description

This function calculates R1 and R2 for the CrossMST test.

Usage

```
getR1R2(E,treated.index)
```

Arguments

E	A matrix with the number of rows the number of edges in the MST and 2 columns. Each row records the subject indices of the two ends of an edge in the MST.
treated.index	The subject indices of the treated subjects.

See Also

[CrossMST](#)

---

smokingDist	*The distance matrix of the smoking data*

Description

This is a 300 by 300 distance matrix with the indices of the treated subjects 1:150 and the indices of the matched controls 151:300. These treated subjects and matched controls are a subset of the original dataset used in the reference paper with 679 treated subjects and 679 matched controls.

Index