Package ‘StratifiedBalancing’

July 22, 2016

Type Package
Title Performs Stratified Covariate Balancing for Data with Discrete and Continuous Outcome Variables
Version 0.2.0
Author Farrokh Alemi Phd, Amr ElRafey
Maintainer Amr ElRafey <ae2504@columbia.edu>
Description Stratified covariate balancing through naturally occurring strata to adjust for confounding and interaction effects. Contains 4 primary functions which perform stratification, sensitivity analysis and return adjusted odds along with naturally occurring strata.
License GPL-2
LazyData TRUE
Depends plyr
NeedsCompilation no
Repository CRAN
Date/Publication 2016-07-22 14:31:47

R topics documented:
clean ................................................................. 2
dispcont ............................................................ 2
dispdisc ............................................................. 3
gerom ............................................................... 3
makeit ............................................................... 3
oddscont ........................................................... 4
oddsdisc ........................................................... 4
senscont ........................................................... 4
sensdisc ........................................................... 5
stratacont .......................................................... 6
stratadisc .......................................................... 7
stratifycont ......................................................... 9
stratifydisc ......................................................... 9
summarycont ....................................................... 10
Description

Function for internal use please refer to functions stratadisc(), stratacont(), sensdisc() and senscont().

Usage

clean(g)

Arguments

g

dispcont

dispcont

Description

Function for internal use please refer to functions stratadisc(), stratacont(), sensdisc() and senscont().

Usage

dispcont(mat)

Arguments

mat
**dispdisc**

---

**Description**

Function for internal use please refer to functions `stratadisc()`, `stratacont()`, `sensdisc()` and `senscont()`.

**Usage**

```
dispdisc(mat)
```

**Arguments**

- `mat`

---

**gerom**

---

**Description**

Function for internal use please refer to functions `stratadisc()`, `stratacont()`, `sensdisc()` and `senscont()`.

**Usage**

```
gerom(d,rer)
```

**Arguments**

- `d`, `rer`

---

**makeit**

---

**Description**

Function for internal use please refer to functions `stratadisc()`, `stratacont()`, `sensdisc()` and `senscont()`.

**Usage**

```
makeit(d)
```

**Arguments**

- `d`
**oddscont**

Description

Function for internal use please refer to functions `stratadisc()`, `stratacont()`, `sensdisc()` and `senscont()`.

Usage

`oddscont(mat)`

Arguments

- **mat**

**oddsdisc**

Description

Function for internal use please refer to functions `stratadisc()`, `stratacont()`, `sensdisc()` and `senscont()`.

Usage

`oddsdisc(mat)`

Arguments

- **mat**

**senscont**

Description

This function performs sensitivity analysis on continuous variable outcomes by dropping variables one at a time, in-order to determine which variables can be removed without altering the results of stratified covariate balancing.

Usage

`senscont(Treatment,Outcome,Matrix)`
Arguments

- **Treatment**: Column number of variable to be used as treatment.
- **Outcome**: Column number of variable to be used as outcome.
- **Matrix**: Name of matrix or data.frame where data is stored.

Details

This function performs sensitivity analysis by dropping variables one at a time, in-order to determine which variables can be removed without altering the results of stratified covariate balancing.

Examples

```r
## In this example we will generate a matrix with a large number of
## covariates and a small number of observations. No model will be
## built into the data, our goal here is to demonstrate how sensitivity
## analysis would be performed.

## Firstly a matrix with 10 columns and 1000 observations will be created
m.matrix(nrow=1000,ncol=10)
for(i in 1:ncol(m)){
m[,i]=rbinom(1000,1,0.5)
}

## We will populate the 10th column randomly from the normal distribution
m[,10]=rnorm(1000,0,1)
## Next we will run function sensdisc() on the data.
g=senscont(9,10,m)
```

Description

This function performs sensitivity analysis on discrete outcome variables by dropping variables one at a time, in-order to determine which variables can be removed without altering the results of stratified covariate balancing.

Usage

`sensdisc(Treatment,Outcome,Matrix)`

Arguments

- **Treatment**: Column number of variable to be used as treatment.
- **Outcome**: Column number of variable to be used as outcome.
- **Matrix**: Name of matrix or data.frame where data is stored.
Details

This function performs sensitivity analysis by dropping variables one at a time, in-order to determine which variables can be removed without altering the results of stratified covariate balancing.

Examples

```r
## In this example we will generate a matrix with a large number of
covariates and a small number of observations. No model will be
built into the data, our goal here is to demonstrate how sensitivity
analysis would be performed.

## Firstly a matrix with 10 columns and 1000 observations will be created
m=matrix(nrow=1000,ncol=10)
for(i in 1:ncol(m)) {
m[,i]=rbinom(1000,1,0.5)
}

## Next we will run function sensdisc() on the data.
g=sensdisc(9,10,m)
```

stratacont

**Stratacont()**

Description

This is the primary stratification function for continuous outcome variables. It locates naturally occurring strata in the data, weights them and returns an improved regression coefficient for the treatment variable which is adjusted for both confounding and interaction terms. Returns a matrix containing naturally occurring strata.

Usage

`stratacont(Treatment,Outcome,Matrix)`

Arguments

- **Treatment**: Column number of variable to be used as treatment.
- **Outcome**: Column number of variable to be used as outcome.
- **Matrix**: Name of matrix or data.frame where data is stored.

Details

This is the primary stratification function for continuous outcome variables. It locates naturally occurring strata in the data, weights them and returns an improved regression coefficient for the treatment variable which is adjusted for both confounding and interaction terms.
Examples

## We will first begin by simulating data in 5 covariates and a continuous outcome
## with significant interaction terms and correlations amongst covariates (to simulate an
## experiment with a strongly non-linear underlying model).
## First, we will create a matrix with the input variables. The input variables will all be
## categorical variables.

```r
m = matrix(nrow=5000, ncol=6)
for (i in 1:ncol(m)) {
  m[,i] = rbinom(5000, 1, 0.5)
}
```

## Next, we will simulate the output variable and include interaction terms

```r
a = (2*m[,5] + 0.5*m[,1] + 4*m[,2] + 2.3*m[,3] + 5*m[,4] +
    2.3*m[,1]*m[,2] + 3.5*m[,1]*m[,3] + 2.1*m[,1]*m[,4] +
    5*m[,1]*m[,2]*m[,3] + 6*m[,1]*m[,4] + 3*m[,1]*m[,2]*m[,4] +
    2*m[,1]*m[,3] + 3.4*m[,1]*m[,2]*m[,3] + m[,1]*m[,4] +
    5*m[,1]*m[,2]*m[,4] + 4*m[,2]*m[,3]*m[,4])

m[,6] = rnorm(1, a, 1)
```

## We are interested in determining the coefficient of covariate 5 which is 2.
## Most straightforward
## way of doing this is to use simple linear regression as follows

```r
k = lm(m[,6] ~ ., data = m[, 1:5])
```

## The value of the coefficient of variable 5 found by the regression can be retrieved using

```r
k$coeff[6]
```

## We can now use the `stratacont()` function to find a more accurate estimation of the coefficient

```r
g = stratacont(5, 6, m)
```

## Note that as the model includes more covariates, the accuracy of the stratification
## techniques is far superior.

---

**Description**

This is the primary stratification function for discrete outcome variables. It locates naturally occurring strata in the data, weights them and returns a common odds ratio for the treatment variable which is adjusted for both confounding and interaction terms. Returns a matrix containing naturally occurring strata.

**Usage**

```r
stratadisc(Treatment, Outcome, Matrix)
```
Arguments

- **Treatment**: Column number of variable to be used as treatment.
- **Outcome**: Column number of variable to be used as outcome.
- **Matrix**: Name of matrix or data.frame where data is stored.

Details

This is the primary stratification function for discrete outcome variables. It locates naturally occurring strata in the data, weights them and returns a common odds ratio for the treatment variable which is adjusted for both confounding and interaction terms.

Examples

```r
## We will first begin by simulating data in 4 covariates and a discrete outcome, with
## significant interaction terms and correlations amongst covariates (to simulate a
## non-randomized experiment with a strongly non-linear underlying model).
## First, we will create a matrix with the input variables
m <- matrix(nrow = 1000, ncol = 5)
for (i in 1:ncol(m)) {
m[, i] <- rbinom(1000, 1, 0.5)
}
## Next, we will create correlations amongst covariates 2, 1 and 3
for (i in 1:nrow(m)) {
  if (m[i, 3] == 1) m[i, 2] <- rbinom(1, 0.8)
  if (m[i, 3] == 0) m[i, 2] <- rbinom(1, 0.2)
  if (m[i, 3] == 1) m[i, 1] <- rbinom(1, 0.8)
  if (m[i, 3] == 0) m[i, 1] <- rbinom(1, 0.2)
}
## Next, we will simulate the output variable and include interaction terms
for (i in 1:nrow(m)) {
a <- exp(2*m[i, 4] + 0.5*m[i, 1] - 4*m[i, 2] + 2.3*m[i, 3] + 2.3*m[i, 3]*m[i, 2] + 8*m[i, 1]*m[i, 2] + 2.1*m[i, 2]*m[i, 3] + 9*m[i, 3]*m[i, 4])
  (1 + exp(2*m[i, 4] + 0.5*m[i, 1] - 4*m[i, 2] + 2.3*m[i, 3] + 2.3*m[i, 3]*m[i, 2] + 8*m[i, 1]*m[i, 2] + 2.1*m[i, 2]*m[i, 3] + 9*m[i, 3]*m[i, 4]))
  m[i, 5] <- rbinom(1, 1, a)
}
## We are interested in determining the coefficient of covariate 4 which is 2.
## The most straightforward way of doing this
## is to use logistic regression as follows
m <- as.data.frame(m)
k <- glm(m[, 5] ~ ., data = m[, (1:4)], family = binomial)
## The value of the coefficient of variable 1 is found by the
## logistic regression can be retrieved using
k$coeff[5]
## We can now use the stratadisc() function to find a
## more accurate estimation of the coefficient
g <- stratadisc(4, 5, m)
## We need to take the log() of the first
## number returned "Odds Ratio of Impact Of Treatment On Outcome"
## and the estimated value of the parameter should be more accurate
```
stratifycont

Description
Function for internal use please refer to functions stratadisc(), stratacont(), sensdisc() and senscont().

Usage
stratifycont(Treatment, Outcome, Matrix)

Arguments
Treatment
Outcome
Matrix

stratifydisc

Description
Function for internal use please refer to functions stratadisc(), stratacont(), sensdisc() and senscont().

Usage
stratifydisc(Treatment, Outcome, Matrix)

Arguments
Treatment
Outcome
Matrix
**summarycont**

**Description**

Function for internal use please refer to functions `stratadisc()`, `stratacont()`, `sensdisc()` and `senscont()`.

**Usage**

```r
summarycont(mat)
```

**Arguments**

- `mat`

---

**summarydisc**

**Description**

Function for internal use please refer to functions `stratadisc()`, `stratacont()`, `sensdisc()` and `senscont()`.

**Usage**

```r
summarydisc(mat)
```

**Arguments**

- `mat`

---

**un1**

**Description**

Function for internal use please refer to functions `stratadisc()`, `stratacont()`, `sensdisc()` and `senscont()`.

**Usage**

```r
un1(d)
```

**Arguments**

- `d`
Description

Function for internal use please refer to functions `stratadisc()`, `stratacont()`, `sensdisc()` and `senscont()`.

Usage

`un3(a,d)`

Arguments

- `a`
- `d`

Description

Function for internal use please refer to functions `stratadisc()`, `stratacont()`, `sensdisc()` and `senscont()`.

Usage

`weightcont(matrix)`

Arguments

- `Matrix`
weightdisc

Description

Function for internal use please refer to functions stratadisc(), stratacont(), sensdisc() and senscont().

Usage

weightdisc(mat)

Arguments

mat
Index

*Topic \textasciitilde kwd1
  clean, 2
  dispcont, 2
dispdisc, 3
gerom, 3
makeit, 3
oddscont, 4
oddsdisc, 4
senscont, 4
sensdisc, 5
stratacont, 6
stratadisc, 7
stratifycont, 9
stratifydisc, 9
summarycont, 10
summarydisc, 10
un1, 10
un3, 11
weightcont, 11
weightdisc, 12

*Topic \textasciitilde kwd2
  clean, 2
dispcont, 2
dispdisc, 3
gerom, 3
makeit, 3
oddscont, 4
oddsdisc, 4
senscont, 4
sensdisc, 5
stratacont, 6
stratadisc, 7
stratifycont, 9
stratifydisc, 9
summarycont, 10
summarydisc, 10
un1, 10
un3, 11
weightcont, 11
weightdisc, 12