Package ‘CMatching’

September 19, 2017

Title Matching Algorithms for Causal Inference with Clustered Data
Version 1.0
Date 2017-08-31
Author Massimo Cannas [aut, cre], Elena Colicino [ctb]
Maintainer Massimo Cannas <massimo.cannas@unica.it>
Description Provides functions to perform matching algorithms for causal inference with clustered data, as described in B. Arpino and M. Cannas (2016) <doi:10.1002/sim.6880>. Pure within-cluster and preferential-within cluster matching are implemented. Both algorithms provide causal estimates with cluster-adjusted estimates of standard errors.
Depends R (>= 2.6.0), Matching
Imports stats, lmtest, multiwayvcov, lme4
Suggests MASS
LazyData false
License GPL-2
Encoding UTF-8
NeedsCompilation no
Repository CRAN
Date/Publication 2017-09-19 17:30:32 UTC

R topics documented:

CMatching-package .................................................. 2
MatchPW ................................................................. 3
MatchW ................................................................. 6
schools ................................................................. 10
summary.Match ...................................................... 13

Index 15
Description

Provides functions to perform matching algorithms for causal inference with clustered data, as described in B. Arpino and M. Cannas (2016) <doi:10.1002/sim.6880>. Pure within-cluster and preferential-within cluster matching are implemented. Both algorithms provide causal estimates with cluster-adjusted estimates of standard errors.

Details

Arpino and Cannas (2016) described several strategies to handle unobserved cluster characteristics in causal inference estimation with clustered data. Depending on researcher’s belief about the strength of unobserved cluster level covariates it is possible to take into account clustering either in the estimation of the propensity score model (through the inclusion of fixed or random effects) and/or in the implementation of the matching algorithm. The package contains functions `matchw` and `matchpw` to adapt classic matching algorithms for causal inference to clustered data and a customized `summary` function to analyze the output. `matchw` implements a pure within-cluster matching while `matchpw` implements an approach which can be called "preferential" within-cluster matching. The preferential approach first searches for matchable units within the same cluster. If no match was found the algorithm searches in other clusters. The functions also provide causal estimands with cluster-adjusted standard errors from fitting a multilevel model on matched data. Both functions are wrappers of the `match` function and return an object of class "match" which can be be used as input of the `MatchBalance` function to examine how much the procedure resulted in improved covariate balance.

Author(s)

Massimo Cannas [aut, cre], Elena Colicino [ctb]

Maintainer: Massimo Cannas <massimo.cannas@unica.it>

References


See Also

`match`, `MatchBalance`
Description

This function implements "preferential" - within clusters matching. In other words, units that do not match within clusters (as defined by the Group variable) can match between cluster in the second step.

Usage

\[
\text{MatchPW}(Y = \text{NULL}, \text{Tr}, X, \text{Group} = \text{NULL}, \text{estimand} = "\text{ATT}", M = 1, \\
\text{exact} = \text{NULL}, \text{caliper} = 0.25, \text{replace} = \text{TRUE}, \text{ties} = \text{TRUE}, \text{weights} = \text{NULL}, \ldots)
\]

Arguments

- **Y**: A vector containing the outcome of interest.
- **Tr**: A vector indicating the treated and control units.
- **X**: A matrix of covariates we wish to match on. This matrix should contain all confounders or the propensity score or a combination of both.
- **Group**: A vector describing the clustering structure (typically the cluster ID). This can be any numeric vector of the same length of Tr and X containing integer numbers in ascending order otherwise an error message will be returned. Default is NULL, however if Group is missing or contains only one integer the output of the Match function is returned with a warning.
- **estimand**: The causal estimand desired, one of "ATE", "ATT" and "ATC", which stand for Average Treatment Effect, Average Treatment effect on the Treated and on the Controls, respectively. Default is "ATT".
- **M**: The number of matches which are sought for each unit. Default is 1 ("one-to-one matching").
- **exact**: An indicator for whether exact matching on the variables contained in X is desired. Default is FALSE. This option has precedence over the caliper option.
- **caliper**: A maximum allowed distance for matching units. Units for which no match was found within caliper distance are discarded. Default is 0.25. The caliper is interpreted in standard deviation units of the unclustered data for each variable. For example, if caliper=0.25 all matches at distance bigger than 0.25 times the standard deviation for any of the variables in X are discarded. The caliper is used for both within and between clusters matching.
- **replace**: Default is TRUE. Note that setting the parameter to FALSE would give a warning since only the within-matching part can be performed without replacement (see Details).
- **ties**: An indicator for dealing with multiple matches. If more than M matches are found for each unit the additional matches are a) wholly retained with equal weights if ties=TRUE; b) a random one is chosen if ties=FALSE. Default is TRUE.
weights A vector of observation specific weights.

Please note that all additional arguments of the Match function are not used.

Details

The function performs preferential-within matching in the clusters defined by the variable Group. In the first phase matching within clusters is performed (see `matchw`) and in the second the unmatched treated (or controls if estimand="ATC") are matched with all controls (treated) units. This can be helpful to avoid dropping many units in small clusters.

Value

`index.control` The index of control observations in the matched dataset.

`index.treated` The index of control observations in the matched dataset.

`index.dropped` The index of dropped observations due to the exact or caliper option. Note that these observations are treated if estimand is "ATT", controls if "ATC".

`est` The causal estimate. This is provided only if Y is not null. If estimand is "ATT" it is the (weighted) mean of Y in matched treated minus the (weighted) mean of Y in matched controls. Equivalently it is the weighted average of the within-cluster ATT’s, with weights given by cluster sizes.

`se` A model-based standard error for the causal estimand. This is a cluster robust estimator of the standard error for the linear model: y ~ constant+Tr, run on the matched dataset (see `cluster.vcov` for details on how this estimator is obtained).

`mdata` The matched datasets. These datasets can also be recovered using `index.treated` and `index.controls`.

`orig.treated.nobs.by.group` The original number of treated observations by group in the dataset.

`orig.control.nobs.by.group` The original number of control observations by group in the dataset.

`orig.dropped.nobs.by.group` The number of dropped observations by group after within cluster matching.

`orig.dropped.nobs.by.group.after.prefwithin` The number of dropped observations by group after preferential within group matching.

`orig.nobs` The original number of observations in the dataset.

`orig.wnobs` The original number of weighted observations in the dataset.

`orig.treated.nobs` The original number of treated observations in the dataset.

`orig.control.nobs` The original number of control observations in the dataset.

`wnobs` the number of weighted observations in the matched dataset.

`caliper` The caliper used.

`intcaliper` The internal caliper used.
exact The value of the exact argument.
nndrops.matches The number of matches dropped either because of the caliper or exact option.
estimand The estimand required.

Note

The function returns an object of class Match. This allows compatibility with the MatchBalance function which can be used to examine the covariate balance before and after matching. See the examples below.

Author(s)

Massimo Cannas [aut, cre], Elena Colicino [ctb]

References


See Also

See also Match, MatchBalance

Examples

data(schools)

# The data set is the subsample of NELS-88 data consisting
# of 10 handpicked schools from the 1003 schools in the full data set.

# Let us consider the following variables:

X<-schools$ses # X may contain more covariates (all confounders)
Y<-schools$math
Tr<-ifelse(schools$homework>1,1,0)
Group<-schools$schid

# Let us assume that the effect of homeworks (Tr) on math score (Y)
# is unconfounded conditional on X and other unobserved schools features.
# Several strategies to handle unobserved group characteristics
# are described in Arpino & Cannas, 2016 (see References).

#### Multivariate Matching on covariates in X (default parameters:
# one-to-one matching on X with replacement with a caliper of 0.25).

# Match within schools
mwc<-MatchW(Y=schools$math, Tr=Tr, X=schools$ses, Group=schools$schid, caliper=0.1)
# Match preferentially within school (first match within schools
# and then (try to) match remaining units between schools)
mpw <- MatchPW(Y=schools$math, Tr=Tr, X=schools$ses, Group=schools$schid, caliper=0.1)
summary(mpw)

# compare balance before and after matching
bmw <- MatchBalance(Tr~ses, data=schools, match.out=mw)
bmpw <- MatchBalance(Tr~ses, data=schools, match.out=mpw)

# see drops by school
mpw$orig.dropped.nobs.by.group.after.within    # after within
mpw$orig.dropped.nobs.by.group.after.prefwithin # after preferential within

### Propensity score matching

# estimate the ps model
mod <- glm(Tr~ses+parented+public+sex+race+urban,
family=binomial(link="logit"), data=schools)
eps <- fitted(mod)

# propensity score matching within schools
psmw <- MatchW(Y=schools$math, Tr=Tr, X=eps, Group=schools$schid, caliper=0.1)

# can combine MatchW and MatchPW with several specifications of ps:
# eg1: match within using ps estimated from dummy logit model
mod <- glm(Tr~ses+parented+public+sex+race+urban
+schid - 1, family=binomial(link="logit"), data=schools)
eps <- fitted(mod)
dpsm <- MatchW(Y=schools$math, Tr=Tr, X=eps, Group=schools$schid, caliper=0.1)

# eg2: classic matching using ps estimated from multilevel logit model
require(lme4)
mod<- glmer(Tr~ses+parented+public+sex+race+urban + (1|schid),
family=binomial(link="logit"), data=schools)
eps <- fitted(mod)

mpsm <- MatchW(Y=schools$math, Tr=Tr, X=eps, Group=NULL, caliper=0.1)
# or simply run Match with X=eps

---

Within-cluster Matching
**Description**

This function implements multivariate and propensity score matching within clusters defined by the Group variable.

**Usage**

```r
MatchW(y = NULL, Tr, X, Group = NULL, estimand = "ATT", M = 1,
exact = NULL, caliper = 0.25, weights = NULL, replace = TRUE, ties = TRUE, ...)
```

**Arguments**

- `y`: A vector containing the outcome of interest.
- `Tr`: A vector indicating the treated and control units.
- `X`: A matrix of covariates we wish to match on. This matrix should contain all confounders or the propensity score or a combination of both.
- `Group`: A vector describing the clustering structure (typically the cluster ID). This can be any numeric vector of the same length of `Tr` and `X` containing integer numbers in ascending order otherwise an error message will be returned. Default is NULL, however if `Group` is missing or contains only one integer the output of the `Match` function is returned with a warning.
- `estimand`: The causal estimand desired, one of "ATE", "ATT" and "ATC", which stand for Average Treatment Effect, Average Treatment effect on the Treated and on the Controls, respectively. Default is "ATT".
- `M`: The number of matches which are sought for each unit. Default is 1 ("one-to-one matching").
- `exact`: An indicator for whether exact matching on the variables contained in `X` is desired. Default is FALSE. This option has precedence over the caliper option.
- `caliper`: A maximum allowed distance for matching units. Units for which no match was found within caliper distance are discarded. Default is 0.25. The caliper is interpreted in standard deviation units of the unclustered data for each variable. For example, if caliper=0.25 all matches at distance bigger than 0.25 times the standard deviation for any of the variables in `X` are discarded.
- `weights`: A vector of specific observation weights.
- `replace`: Matching can be with or without replacement depending on whether matches can be re-used or not. Default is TRUE.
- `ties`: An indicator for dealing with multiple matches. If more than `M` matches are found for each unit the additional matches are a) wholly retained with equal weights if ties=TRUE; b) a random one is chosen if ties=FALSE. Default is TRUE.

**Details**

This function is a wrapper of the `Match` function meant to be a natural extension of the `Match` function to clustered data.
The function retains main arguments of `Match` and returns the same type of object ("Match") with some additional output showing the matching cluster by cluster. It differs from wrapper `matchby` because model-based standard errors are given instead of classic standard errors and because the caliper is the same for each cluster. Moreover, observation weights are available.

**Value**

- `index.control`: The index of control observations in the matched dataset.
- `index.treated`: The index of control observations in the matched dataset.
- `index.dropped`: The index of dropped observations due to the exact or caliper option. Note that these observations are treated if estimand is "ATT", controls if "ATC".
- `est`: The causal estimate. This is provided only if \( Y \) is not null. If estimand is "ATT" it is the (weighted) mean of \( Y \) in matched treated units minus the (weighted) mean of \( Y \) in matched controls. Equivalently, it is the weighted average of the within-cluster ATT’s, with weights given by cluster sizes.
- `se`: A model-based standard error for the causal estimand. This is a cluster robust estimator of the standard error for the linear model: \( Y \sim constant + Tr \), run on the matched dataset (see `cluster.vcov` for details on how this estimator is obtained). Note that these standard errors differ from a weighted average of cluster specific standard errors provided by the `matchby` function, which are generally larger. Estimating standard errors for causal parameters with clustered data is an active field of research and there is no perfect solution to date so the choice of standard errors should be considered carefully.
- `mdata`: The matched datasets. These datasets can also be recovered using `index.treated` and `index.controls`.
- `orig.treated.nobs.by.group`: The original number of treated observations by group in the dataset.
- `orig.control.nobs.by.group`: The original number of control observations by group in the dataset.
- `orig.dropped.nobs.by.group`: The number of dropped observations by group after within cluster matching.
- `orig.nobs`: The original number of observations in the dataset.
- `orig.wnobs`: The original number of weighted observations in the dataset.
- `orig.treated.nobs`: The original number of treated observations in the dataset.
- `orig.control.nobs`: The original number of control observations in the dataset.
- `wnobs`: The number of weighted observations in the matched dataset.
- `caliper`: The caliper used.
- `intcaliper`: The internal caliper used.
- `exact`: The value of the exact argument.
- `ndrops.matches`: The number of matches dropped either because of the caliper or exact option.
- `estimand`: The estimand required.
Note

The function returns an object of class `Match`. This allows compatibility with the `MatchBalance` function which can be used to examine the covariate balance before and after matching (see the examples below).

Author(s)

Massimo Cannas [aut, cre], Elena Colicino [ctb]

References


See Also

See also `Match`, `MatchBalance`

Examples

data(schools)

# The data set is the subsample of NELS-88 data consisting
# of 10 handpicked schools from the 1003 schools in the full data set.

# Let us consider the following variables:
X<-schools$ses # X may contain more covariates (all confounders)
Y<-schools$math
Tr<-ifelse(schools$homework>1,1,0)
Group<-schools$schid

# Let us assume that the effect of homeworks (Tr) on math score (Y)
# is unconfounded conditional on X and other unobserved schools features.
# Several strategies to handle unobserved group characteristics
# are described in Arpino & Cannas, 2016 (see References).

### Multivariate Matching on covariates in X (default parameters:
# one-to-one matching on X with replacement with a caliper of 0.25).

# Match within schools
mw<-MatchW(Y=schools$math, Tr=Tr, X=schools$ses, Group=schools$schid, caliper=0.1)
summary(mw)

# Match preferentially within school (first match within schools
# and then (try to) match remaining units between schools)
mpw <- MatchPW(Y=schools$math, Tr=Tr, X=schools$ses, Group=schools$schid, caliper=0.1)
summary(mpw)

# compare balance before and after matching
bmw <- MatchBalance(Tr=ses, data=schools, match.out=mw)
mpw <- MatchBalance(Tr=ses, data=schools, match.out=mpw)

# see drops by school
mpw$orig.dropped.nobs.by.group.after.within # after within
mpw$orig.dropped.nobs.by.group.after.prefwithin # after preferential within

### Propensity score matching

# estimate the ps model
mod <- glm(Tr~ses+parented+public+sex+race+urban,
          family=binomial(link="logit"), data=schools)
eps <- fitted(mod)

# propensity score matching within schools
psmw <- MatchW(Y=schools$math, Tr=Tr, X=eps, Group=schools$schid, caliper=0.1)

# can combine MatchW and MatchPW with several specifications of ps:
# eg 1: match within using ps estimated from dummy logit model
mod <- glm(Tr ~ ses + parented + public + sex + race + urban + schid ~ 1,
          family=binomial(link="logit"), data=schools)
eps <- fitted(mod)

dpsm <- MatchW(Y=schools$math, Tr=Tr, X=eps, Group=schools$schid, caliper=0.1)

# eg 2: classic matching using ps estimated from multilevel logit model
require(lme4)
mod<glmer(Tr ~ ses + parented + public + sex + race + urban + (1|schid),
          family=binomial(link="logit"), data=schools)
eps <- fitted(mod)

mpsmc<-MatchW(Y=schools$math, Tr=Tr, X=eps, Group=NULL, caliper=0.1)
# or simply run Match with X=eps

---

**Description**

Data set used by Kreft and De Leeuw in their book *Introducing Multilevel Modeling, Sage (1988)* to analyse the relationship between math score and time spent by students to do math homework. The data set is a subsample of NELS-88 data consisting of 10 handpicked schools from the 1003 schools in the full data set. Students are nested within schools and information is available both at the school and student level.
**Usage**

data("schools")

**Format**

A data frame with 260 observations on the following 19 variables.

- **schid**: School ID: a numeric vector identifying each school.
- **stuid**: The student ID.
- **ses**: Socioeconomic status.
- **meanses**: Mean ses for the school.
- **homework**: The number of hours spent weekly doing homeworks.
- **white**: A dummy for white race (=1) versus non-white (=0).
- **parented**: Parents highest education level.
- **public**: Public school: 1=public, 0=non public.
- **ratio**: Student-teacher ratio.
- **percmin**: Percent minority in school.
- **math**: Math score
- **sex**: Sex: 1=male, 2=female.
- **race**: Race of student, 1=asian, 2=Hispanic, 3=Black, 4=White, 5=Native American.
- **sctype**: Type of school: 1=public, 2=catholic, 3=Private other religion, 4=Private non-r.
- **cstr**: Classroom environment structure: ordinal from 1=not accurate to 5=very much accurate.
- **scsize**: School size: ordinal from 1=[1,199) to 7=[1200+).
- **urban**: Urbanicity: 1=Urban, 2=Suburban, 3=Rural.
- **region**: Geographic region of the school: NE=1, NC=2, South=3, West=4.
- **schnum**: Standardized school ID.

**Source**


**Examples**

data(schools)

# The data set is the subsample of NELS-88 data consisting
# of 10 handpicked schools from the 1003 schools in the full data set.

# Let us consider the following variables:

X<-schools$ses # X may contain more covariates (all confounders)
Y<-schools$math
# Let us assume that the effect of homeworks (\(T_r\)) on math score (\(Y\))
# is unconfounded conditional on \(X\) and other unobserved schools features.
# Several strategies to handle unobserved group characteristics
# are described in Arpino & Cannas, 2016 (see References).

### Multivariate Matching on covariates in \(X\) (default parameters: # one-to-one matching on \(X\) with replacement with a caliper of 0.25).

# Match within schools
\[ mw < - \text{MatchW}(Y = \text{schools}$\text{math}$, \(T_r = T_r\), \(X = \text{schools}$\text{ses}\), \(\text{Group} = \text{schools}$\text{schid}\), caliper = 0.1) \]
\[ \text{summary}(mw) \]

# Match preferentially within school (first match within schools
# and then (try to) match remaining units between schools)
\[ mpw < - \text{MatchPW}(Y = \text{schools}$\text{math}$, \(T_r = T_r\), \(X = \text{schools}$\text{ses}\), \(\text{Group} = \text{schools}$\text{schid}\), caliper = 0.1) \]
\[ \text{summary}(mpw) \]

# compare balance before and after matching
\[ \text{bmw} < - \text{MatchBalance}(Tr = \text{ses}, data = \text{schools}, match.out = mw) \]
\[ \text{bmpw} < - \text{MatchBalance}(Tr = \text{ses}, data = \text{schools}, match.out = mpw) \]

# see drops by school
\[ \text{mpw}$\text{orig}$, \text{dropped.nobs.by.group.after.within} \]
\[ \text{mpw}$\text{orig}$, \text{dropped.nobs.by.group.after.within} \]

### Propensity score matching

# estimate the ps model
\[ \text{mod} < - \text{glm}(T_r = \text{ses} + \text{parented} + \text{public} + \text{sex} + \text{race} + \text{urban}, \]
\[ \text{family} = \text{binomial}(\text{link} = \text{"logit"}), \text{data} = \text{schools} \]
\[ \text{eps} < - \text{fitted(mod)} \]

# propensity score matching within schools
\[ \text{psmw} < - \text{MatchW}(Y = \text{schools}$\text{math}$, \(T_r = T_r\), \(X = \text{eps}\), \(\text{Group} = \text{schools}$\text{schid}\), caliper = 0.1) \]

# can combine MatchW and MatchPW with several specifications of ps:
# eg 1: match within using ps estimated from dummy logit model
\[ \text{mod} < - \text{glm}(T_r - \text{ses} - \text{parented} + \text{public} + \text{sex} + \text{race} + \text{urban} \]
\[ + \text{schid} - 1, \text{family} = \text{binomial}(\text{link} = \text{"logit"}), \text{data} = \text{schools} \]
\[ \text{eps} < - \text{fitted(mod)} \]
\[ \text{dpsm} < - \text{MatchW}(Y = \text{schools}$\text{math}$, \(T_r = T_r\), \(X = \text{eps}\), \(\text{Group} = \text{schools}$\text{schid}\), caliper = 0.1) \]

# eg2: classic matching using ps estimated from multilevel logit model
\[ \text{require(lme4)} \]
\[ \text{mod} < - \text{glmer}(T_r - \text{ses} - \text{parented} + \text{public} + \text{sex} + \text{race} + \text{urban} + (1|\text{schid}), \]
\[ \text{family} = \text{binomial}(\text{link} = \text{"logit"}), \text{data} = \text{schools} \]
\[ \text{eps} < - \text{fitted(mod)} \]
\[ \text{dpsm} < - \text{MatchW}(Y = \text{schools}$\text{math}$, \(T_r = T_r\), \(X = \text{eps}\), \(\text{Group} = \text{schools}$\text{schid}\), caliper = 0.1) \]
family=binomial(link="logit"), data=schools)
eps <- fitted(mod)

mpsm<-MatchW(Y=schools$math, Tr=Tr, X=eps, Group=NULL, caliper=0.1)
# or simply run Match with X=eps

summary.Match  

Summarizing output from MatchW and MatchPW

Description

summary method for MatchW and MatchPW

Usage

## S3 method for class 'Match'
summary(object, ..., full = FALSE, digits = 5)

Arguments

object  
An object of class "Match".

...  
Other options for the generic summary function.

full  
A flag for whether the unadjusted estimates and naive standard errors should also be summarized.

digits  
The number of significant digits that should be displayed.

Details

A summary of most important output from a "Match" object. If Group contains only one value the output is the same of summary(Match()). Otherwise the output contains also the distribution of treated (control) observations by group and the distribution of dropped (because od 'caliper' or 'exact' option) by group.

Note

Naive standard errors are not available when there is more than one group so the full parameter is ineffective in that case.

Author(s)

Massimo Cannas <massimo.cannas@unica.it>

References


See Also

See also `Match, MatchW, MatchPW, MatchBalance`
Index

+ Topic causal inference
  CMatching-package, 2
+ Topic clustered data
  MatchPW, 3
  MatchW, 6
+ Topic cluster
  CMatching-package, 2
+ Topic matching
  MatchPW, 3
  MatchW, 6
+ Topic school dataset (NELS-88)
  schools, 10

cluster.vcov, 4, 8
CMatching (CMatching-package), 2
CMatching-package, 2

Match, 2, 5, 9, 14
MatchBalance, 2, 5, 9, 14
MatchPW, 3, 13, 14
MatchW, 6, 13, 14

print.summary.Match (summary.Match), 13

schools, 10
summary.Match, 13