Package 'CovSel'

April 9, 2025

Version 1.2.2

Depends np, MASS

Suggests bindata

Title Model-Free Covariate Selection

Description Model-free selection of covariates under unconfoundedness for situations where the parameter of interest is an average causal effect. This package is based on model-free backward elimination algorithms proposed in de Luna, Waernbaum and Richardson (2011). Marginal co-ordinate hypothesis testing is used in situations where all covariates are continuous while kernel-based smoothing appropriate for mixed data is used otherwise.

License GPL-3

Encoding UTF-8

NeedsCompilation no

Date 2025-04-09

Author Jenny Häggström [aut, cre],
Emma Persson [aut],
Sandy Weisberg [aut] (Author of functions originating from the package
dr version 3.0.10.)

Maintainer Jenny Häggström <jenny.haggstrom@umu.se>

Repository CRAN

Date/Publication 2025-04-09 06:50:01 UTC

Contents

---

cov.sel   *Model-Free Selection of Covariate Sets*

Description

Dimension reduction of the covariate vector under unconfoundedness using model-free backward elimination algorithms, based on either marginal co-ordinate hypothesis testing, (MCH), (continuous covariates only) or kernel-based smoothing, (KS).

Usage

```
cov.sel(T, Y, X, type=c("dr", "np"), alg = 3,scope = NULL, alpha = 0.1,
thru=0.5,thro=0.25,thrc=100,...)
```

Arguments

T	A vector, containing 0 and 1, indicating the binary treatment variable.
Y	A vector of observed outcomes.
X	A matrix or data frame containing columns of covariates. The covariates may be a mix of continuous, unordered discrete (to be specified in the data frame using `factor`), and ordered discrete (to be specified in the data frame using `ordered`).
type	The type of method used. `"dr"` for MCH and `"np"` for KS. MCH is suitable in situations with only continuous covariates while KS can be used if discrete covariates are present.
alg	Specifying which algorithm to be use. 1 indicates Algorithm A, 2 indicates Algorithm B and 3 runs them both. See Details. `alg = 3` is default.
scope	A character string giving the name of one (or several) covariate(s) that must not be removed.
alpha	Stopping criterion for MCH: will stop removing covariates when the p-value for the next covariate to be removed is less then `alpha`. The default is `alpha = 0.1`.
thru	Bandwidth threshold used for unordered discrete covariates if `type="np"`. Values in $[0,1]$ are valid. `thru=0` removes all unordered discrete covariates and `thru=1` removes none of them. Default is `thru=0.5`. See Details.
thro	Bandwidth threshold used for ordered discrete covariates if `type="np"`. Values in $[0,1]$ are valid. `thro=0` removes all unordered discrete covariates and `thro=1` removes none of them. Default is `thro=0.25`. See Details.
thrc	Bandwidth threshold used for continuous covariates if `type="np"`. Non-negative values are valid. Default is `thr=100`. See Details.
...	Additional arguments passed on to `dr`, `dr.step` or `npregbw`. If `type="dr"`, `method`, can be set to `"sir"` or `"save"`, the first being default, `trace=0` suppresses the output generated by `dr.step`. If `type="np"`, `regtype` can be set to `"lc"` or `"ll"`, the first being default and `bwtype` can be set to `"fixed"`, `"generalized_nn"` or `"adaptive_nn"`, defaults to `"fixed"`. See `dr` and `npregbw` for usage of `na.action`.

Details

Performs model-free selection of covariates for situations where the parameter of interest is an average causal effect. This function is based on the framework of sufficient dimension reduction, that under unconfoundedness, reduces dimension of the covariate vector. A two-step procedure searching for a sufficient subset of the covariate vector is implemented in the form of algorithms. This function uses MCH (if type="dr") or KS (if type="np") in the form of two backward elimination algorithms, Algorithm A and Algorithm B proposed by de Luna, Waernbaum and Richardson (2011).

Algorithm A (alg = 1): First the covariates conditionally independent of the treatment, T, given the rest of the variables (X.T) are removed. Then the covariates conditionally independent of the potential outcomes (in each of the treatment groups) given the rest of the covariates are removed. This yields two subsets of covariates; Q.1 and Q.0 for the treatment and control group respectively.

Algorithm B (alg = 2): First the covariates conditionally independent of the potential outcome (in each of the treatment groups), given the rest of the covariates (X.0 and X.1) are removed. Then the covariates conditionally independent of the treatment, T, given the rest of the covariates are removed. This yields two subsets of covariates; Z.1 and Z.0 for the treatment and control group respectively.

alg=3 runs both Algorithm A and B.

In KS the bandwidth range for unordered discrete covariates is [0, 1/#levels] while for ordered discrete covariates, no matter how many levels, the range is [0, 1]. For continuous covariates bandwidths ranges from 0 to infinity. Ordered discrete and continuous covariates are removed if their bandwidths exceed their respective thresholds. Unordered discrete covariates are removed if their bandwidths are larger than thru times the maximum bandwidth.

In case of MCH one can choose between sliced inverse regression, SIR, or sliced average variance estimation, SAVE. For KS the regression type can be set to local constant kernel or local linear and the bandwidth type can be set to fixed, generalized nearest neighbors or adaptive nearest neighbors. See dr and npregbw for details. Since type="np" results in a fully nonparametric covariate selection procedure this can be much slower than if type="dr".

Value

cov.sel returns a list with the following content:

X.T	The of covariates with minimum cardinality such that $P(T|X) = P(T|X.T)$.
Q.0	The set of covariates with minimum cardinality such that $P(Y.0|X.T) = P(Y.0|Q.0)$. Where Y.0 is the response in the control group.
Q.1	The set of covariates with minimum cardinality such that $P(Y.1|X.T) = P(Y.1|Q.1)$. Where Y.1 is the response in the treatment group.
X.0	The set of covariates with minimum cardinality such that $P(Y.0|X) = P(Y.0|X.0)$.
X.1	The set of covariates with minimum cardinality such that $P(Y.1|X) = P(Y.1|X.1)$.
Z.0	The set of covariates with minimum cardinality such that $P(T|X.0) = P(T|Z.0)$.
Z.1	The set of covariates with minimum cardinality such that $P(T|X.1) = P(T|Z.1)$.

If type="dr" the following type-specific content is returned:

evectorsQ.0	The eigenvectors of the matrix whose columns span the reduced subspace Q.0.

evectorsQ.1   The eigenvectors of the matrix whose columns span the reduced subspace Q.1.

evectorsZ.0   The eigenvectors of the matrix whose columns span the reduced subspace Z.0.

evectorsZ.1   The eigenvectors of the matrix whose columns span the reduced subspace Z.1.

method   The method used, either "sir" or "save".

If type="np" the following type-specific content is returned:

bandwidthsQ.0   The selected bandwidths for the covariates in the reduced subspace Q.0.

bandwidthsQ.1   The selected bandwidths for the covariates in the reduced subspace Q.1.

bandwidthsZ.0   The selected bandwidths for the covariates in the reduced subspace Z.0.

bandwidthsZ.1   The selected bandwidths for the covariates in the reduced subspace Z.1.

regtype   The regression method used, either "lc" or "ll".

bwtype   Type of bandwidth used, "fixed", "generalized_nn" or "adaptive_nn"

covar   Names of all covariates given as input X.

For marginal co-ordinate hypothesis test, type="dr", as a side effect a data frame of labels, tests, and p.values is printed.

Note

cov.sel calls the functions dr, dr.step and npregbw so the packages dr and np are required.

Author(s)

Emma Persson, <emma.persson@umu.se>, Jenny Häggström, <jenny.haggstrom@umu.se>

References

Cook, R. D. (2004). Testing Predictor contributions in Sufficient Dimension Reduction. *The Annals of statistics 32*. 1061-1092

de Luna, X., I. Waernbaum, and T. S. Richardson (2011). Covariate selection for the nonparametric estimation of an average treatment effect. *Biometrika 98*. 861-875

Häggström, J., E. Persson, I. Waernbaum and X. de Luna (2015). An R Package for Covariate Selection When Estimating Average Causal Effects. *Journal of Statistical Software 68*. 1-20

Hall, P., Q. Li and J.S. Racine (2007). Nonparametric estimation of regression functions in the presence of irrelevant regressors. *The Review of Economics and Statistics, 89*. 784-789

Li, L., R. D. Cook, and C. J. Nachtsheim (2005). Model-free Variable Selection. *Journal of the Royal Statistical Society, Series B 67*. 285-299

See Also

[np](np)

Examples

```
## Marginal co-ordinate hypothesis test, continuous covariates only

data(datc)


##Algorithm A, keeping x6 and x7

ans <- cov.sel(T = datc$T, Y = datc$y, X = datc[,1:8], type="dr",
                alpha = 0.1, alg = 1, scope=c("x6","x7"))

summary(ans)

##Algorithm B, method "save"

ans <- cov.sel(T = datc$T, Y = datc$y, X = datc[,1:10], type="dr",
                alg = 2, method = "save", alpha = 0.3, na.action = "na.omit")

## Kernel-based smoothing, both categorical and continuous covariates

data(datfc)
##The example below with default setting takes about 9 minutes to run.
## ans <- cov.sel(T = datfc$T, Y = datfc$y, X = datfc[,1:8], type="np",
##                 alpha = 0.1, alg = 3, scope=NULL, thru=0.5, thro=0.25, thrc=100)

## For illustration purposes we run Algorithm A using only the first 100 observations
##and x1, x2, x3, x4 in datfc
ans <- cov.sel(T = datfc$T[1:100], Y = datfc$y[1:100], X = datfc[1:100,1:4],
      type="np",alpha = 0.1, alg = 1, scope=NULL, thru=0.5, thro=0.25, thrc=100)

##The example below running Algorithm A, keeping x6 and x7 with regtype="ll"
##takes about 7 minutes to run.
##ans <- cov.sel(T = datfc$T, Y = datfc$y, X = datfc[,1:8], type="np",
##                 alpha = 0.1, alg = 3, scope=c("x6","x7"), thru=0.5, thro=0.25,
##                 thrc=100, regtype="ll")
```

---

cov.sel.np	*cov.sel.np*

Description

Function called by `cov.sel` if type="np". Not meant to be used on its own.

Usage

```
cov.sel.np(T, Y, X, alg, scope, thru, thro, thrc, dat, data.0,
data.1, covar, ...)
```

Arguments

T	A vector, containing 0 and 1, indicating the binary treatment variable.
Y	A vector of observed outcomes.
X	A matrix or data frame containing columns of covariates. The covariates may be a mix of continuous, unordered discrete (to be specified in the data frame using `factor`), and ordered discrete (to be specified in the data frame using `ordered`).
alg	Specifying which algorithm to be use. 1 indicates Algorithm A, 2 indicates Algorithm B and 3 runs them both. See Details. alg = 3 is default.
scope	A character string giving the name of one (or several) covariate(s) that must not be removed.
thru	Bandwidth threshold for unordered discrete covariates. Values in $[0, 1]$ are valid. thru=0 removes all unordered discrete covariates and thru=1 removes none of them. Default is thru=0.5.
thro	Bandwidth threshold for ordered discrete covariates. Values in $[0, 1]$ are valid. thro=0 removes all unordered discrete covariates and thro=1 removes none of them. Default is thro=0.25.
thrc	Bandwidth threshold for continuous covariates. Non-negative values are valid. Default is thr=100.
dat	Passed on from cov.sel
data.0	Passed on from cov.sel
data.1	Passed on from cov.sel
covar	Passed on from cov.sel
...	Additional arguments passed on to npregbw. regtype can be set to "lc" or "ll", the first being default and bwtype can be set to "fixed", "generalized_nn" or "adaptive_nn", defaults to "fixed".

Details

See cov.sel for details.

Value

Function returns subsets, methods and removed covariates. See cov.sel for details.

Note

cov.sel.np calls the function npregbw so the package np is required.

Author(s)

Jenny Häggström, <jenny.haggstrom@umu.se>

References

de Luna, X., I. Waernbaum, and T. S. Richardson (2011). Covariate selection for the nonparametric estimation of an average treatment effect. *Biometrika 98*. 861-875

Häggström, J., E. Persson, I. Waernbaum and X. de Luna (2015). An R Package for Covariate Selection When Estimating Average Causal Effects. *Journal of Statistical Software 68*. 1-20

Hall, P., Q. Li and J.S. Racine (2007). Nonparametric estimation of regression functions in the presence of irrelevant regressors. *The Review of Economics and Statistics, 89*. 784-789

See Also

np

---

datc   *Simulated Data, Continuous*

Description

This data is simulated. The covariates, X, are all generated from a standard normal distribution and they are all independent except for x_7 and x_8 (cor(x_7,x_8)=0.5). The code generating the data is

```
library(MASS)
set.seed(9327529)
n<-1000
eta<-mvrnorm(n,rep(0,2),diag(1,2,2))
Sigma=diag(1,10,10)
Sigma[7,8]<-Sigma[8,7]<-0.5
X<-mvrnorm(n,rep(0,10),Sigma)
y0<-2+2*X[,1]+2*X[,2]+2*X[,5]+2*X[,6]+2*X[,8]+eta[,1]
y1<-4+2*X[,1]+2*X[,2]+2*X[,5]+2*X[,6]+2*X[,8]+eta[,2]
e<-1/(1+exp(-0.5*X[,1]-0.5*X[,2]-0.5*X[,3]-0.5*X[,4]-0.5*X[,7]))
T<-rbinom(n,1,e)
y<-y1*T+y0*(1-T)
datc<-data.frame(x1=X[,1],x2=X[,2],x3=X[,3],x4=X[,4],x5=X[,5],x6=X[,6],
x7=X[,7],x8=X[,8],x9=X[,9],x10=X[,10],y0,y1,y,T)
```

Usage

```
data(datc)
```

Format

A data frame with 1000 observations on the following 14 variables.

x1 a numeric vector

x2 a numeric vector

x3 a numeric vector

   x4 a numeric vector

   x5 a numeric vector

   x6 a numeric vector

   x7 a numeric vector

   x8 a numeric vector

   x9 a numeric vector

   x10 a numeric vector

   y0 a numeric vector

   y1 a numeric vector

   y a numeric vector

   T a numeric vector

---

datf   *Simulated Data, Factors*

Description

This data is simulated. The covariates, X, and the treatment, T, are all generated by simulating in-
dependent bernoulli distributions or from a multivariate normal distribution and then dichotomizing
to get binary variables with a certain dependence structure.The code generating the data is

```
library(bindata)
set.seed(9327529)
n<-500
x1 <- rbinom(n, 1, prob = 0.5)
x25 <- rmvbin(n, bincorr=cbind(c(1,0.7),c(0.7,1)), margprob=c(0.5,0.5))
x34 <- rmvbin(n, bincorr=cbind(c(1,0.7),c(0.7,1)), margprob=c(0.5,0.5))
x2 <- x25[,1]
x3 <- x34[,1]
x4 <- x34[,2]
x5 <- x25[,2]
x6 <- rbinom(n, 1, prob = 0.5)
x7<- rbinom(n, 1, prob = 0.5)
x8 <- rbinom(n, 1, prob = 0.5)
e0<-rnorm(n)
e1<-rnorm(n)
p <- 1/(1 + exp(3 - 1.5 * x1 - 1.5 * x2 - 1.5 * x3 - 0.1 * x4 - 0.1 * x5 - 1.3 * x8))
T <- rbinom(n, 1, prob = p)
y0 <- 4 + 2 * x1 + 3 * x4 + 5 * x5 + 2 * x6 + e0
y1 <- 2 + 2 * x1 + 3 * x4+ 5 * x5 + 2 * x6 + e1
y <- y1 * T + y0 * (1 - T)
datf <- data.frame(x1, x2, x3, x4, x5, x6, x7, x8, y0, y1, y, T)
datf[, 1:8] <- lapply(datf[, 1:8], factor)
datf[, 12] <- as.numeric(datf[, 12])
```

Usage

```
data(datf)
```

Format

A data frame with 500 observations on the following 12 variables.

x1   a factor with two levels

x2   a factor with two levels

x3   a factor with two levels

x4   a factor with two levels

x5   a factor with two levels

x6   a factor with two levels

x7   a factor with two levels

x8   a factor with two levels

y0   a numeric vector

y1   a numeric vector

y   a numeric vector

T   a numeric vector

---

datfc   *Simulated Data, Mixed*

Description

This data is simulated. The covariates, X, and the treatment, T, are all generated by simulating from independent or multivariate normal distributions and then some variables are dichotomized to get binary variables with a certain dependence structure. The code generating the data is

```
library(bindata)
set.seed(9327529)
n<-500
x1 <- rnorm(n, mean = 0, sd = 1)
x2 <- rbinom(n, 1, prob = 0.5)
x25 <- rmvbin(n, bincorr=cbind(c(1,0.7),c(0.7,1)), margprob=c(0.5,0.5))
x2 <- x25[,1]
Sigma <- matrix(c(1,0.5,0.5,1),ncol=2)
x34 <- mvrnorm(n, rep(0, 2), Sigma)
x3 <- x34[,1]
x4 <- x34[,2]
x5 <- x25[,2]
x6 <- rbinom(n, 1, prob = 0.5)
x7<- rnorm(n, mean = 0, sd = 1)
```

```
x8 <- rbinom(n, 1, prob = 0.5)
e0<-rnorm(n)
e1<-rnorm(n)
p <- 1/(1 + exp(3 - 1.2 * x1 - 3.7 * x2 - 1.5 * x3 - 0.3 * x4 - 0.3 * x5 - 1.9 * x8))
T <- rbinom(n, 1, prob = p)
y0 <- 4 + 2 * x1 + 3 * x4 + 5 * x5 + 2 * x6 + e0
y1 <- 2 + 2 * x1 + 3 * x4+ 5 * x5 + 2 * x6 + e1
y <- y1 * T + y0 * (1 - T)
datfc <- data.frame(x1, x2, x3, x4, x5, x6, x7, x8, y0, y1, y, T)
datfc[, c(2, 5, 6, 8)] <- lapply(datfc[, c(2, 5, 6, 8)], factor)
datfc[, 12] <- as.numeric(datfc[, 12])
```

Usage

```
data(datfc)
```

Format

A data frame with 500 observations on the following 12 variables.

x1 a numeric vector

x2 a factor with two levels

x3 a numeric vector

x4 a numeric vector

x5 a factor with two levels

x6 a factor with two levels

x7 a numeric vector

x8 a factor with two levels

y0 a numeric vector

y1 a numeric vector

y a numeric vector

T a numeric vector

---

lalonde   *Real data, Lalonde*

Description

In order for the code used to create this data frame to work text files available on Dehejia's webpage <https://users.nber.org/~rdehejia/nswdata2.html> need to be downloaded and stored in the working directory. The data frame consists of 185 treated units from a randomized evaluation of a labor training program, the National Supported Work (NSW) Demonstration, and 429 nonexperimental comparison units drawn from survey datasets.

```
treated <- read.table(file = "nswre74_treated.txt")
controls <- read.table(file = "cps3_controls.txt")
nsw <- rbind(treated, controls)
ue <- function(x) factor(ifelse(x > 0, 0, 1))
UE74 <- mapply(ue, nsw[, 8])
UE75 <- mapply(ue, nsw[, 9])
nsw[, 4:7] <- lapply(nsw[, 4:7], factor)
lalonde <- cbind(nsw[, 1:9], UE74, UE75, nsw[, 10])
colnames(lalonde) <- c("treat", "age", "educ", "black", "hisp", "married",
"nodegr", "re74", "re75", "u74", "u75", "re78")
```

Usage

```
data(lalonde)
```

Format

A data frame with 614 observations on the following 12 variables.

treat   a numeric vector

age   a numeric vector

educ   a numeric vector

black   a factor with two levels

hisp   a factor with two levels

married   a factor with two levels

nodegr   a factor with two levels

re74   a numeric vector

re75   a numeric vector

u74   a factor with two levels

u75   a factor with two levels

re78   a numeric vector

---

summary.cov.sel   *Summary*

Description

This function produce a summary of the results of the covariate selection done by invoking cov.sel.

Usage

```
## S3 method for class 'cov.sel'
summary(object, ...)
```

Arguments

`object`	The list that `cov.sel` returns
`...`	additional arg

Details

Function gives subsets, method and removed variables.

Value

`X.T`	subset `X.T`
`X.0`	subset `X.0`
`X.1`	subset `X.1`
`Q.0`	subset `Q.0`
`Q.1`	subset `Q.1`
`Z.0`	subset `Z.0`
`Z.1`	subset `Z.1`
`method`	The method
`Q.0comp`	The complement subset of covariates to `Q.0`
`Q.1comp`	The complement subset of covariates to `Q.1`
`Z.0comp`	The complement subset of covariates to `Z.0`
`Z.1comp`	The complement subset of covariates to `Z.1`

Author(s)

Emma Persson, <emma.persson@umu.se>

Index