Journal of Statistical Software

November 2015, Volume 68, Issue 1. doi: 10.18637/jss.v068.i01

CovSel: An R Package for Covariate Selection When Estimating Average Causal Effects

Jenny Häggström Umeå University
Emma Persson Umeå University
Ingeborg Waernbaum Umeå University
Xavier de Luna Umeå University

Abstract

We describe the R package **CovSel**, which reduces the dimension of the covariate vector for the purpose of estimating an average causal effect under the unconfoundedness assumption. Covariate selection algorithms developed in De Luna, Waernbaum, and Richardson (2011) are implemented using model-free backward elimination. We show how to use the package to select minimal sets of covariates. The package can be used with continuous and discrete covariates and the user can choose between marginal co-ordinate hypothesis tests and kernel-based smoothing as model-free dimension reduction techniques.

Keywords: causal inference, dimension reduction, **dr**, **np**, R.

1. Introduction

The theory and practice of causal inference from observational studies is an active research field within the statistical sciences (including econometrics and epidemiology). Typical observational studies have as purpose to evaluate the effect of a causal variable (often called treatment) on an outcome of interest. Since in observational studies pre-treatment variables, henceforth covariates, are not expected to have a distribution balanced between treatment groups, as opposed to a randomized study, one needs to control for confounding covariates. Under unconfoundedness, i.e., the potential outcomes are independent of the treatment assignment given a vector of covariates, an average causal effect may be identified.

The starting point for covariate selection should be subject matter knowledge, which in practice often gives only partial guidance, and this might result in an unnecessarily high-dimensional covariate vector. When estimating an average causal effect non-parametrically, controlling for too many covariates may result in poor performance of the estimator (e.g., Rubin 1997; Hahn 2004; De Luna *et al.* 2011) emphasizing the importance of avoiding conditioning on redundant covariates. A common practice has been to control for all covariates

affecting the treatment assignment without considering whether they are also related to outcome. The lack of theoretical results in the literature on the issue of covariate selection was pointed out in Imbens and Wooldridge (2009), and new results have recently appeared; see De Luna *et al.* (2011), Vansteelandt, Bekaert, and Claeskens (2012), Laan and Gruber (2010); see also earlier work by Robins and Rotnitzky (1995) and Hahn (2004).

In this article we introduce the R (R Core Team 2015) package **CovSel** (Häggström and Persson 2015) aiming at reducing the covariate set when the purpose of the analysis is to estimate an average causal effect non-parametrically. The package is available from the Comprehensive R Archive Network (CRAN) at `http://CRAN.R-project.org/package=CovSel` and implements the general algorithms for covariate selection proposed by De Luna *et al.* (2011) and Persson, Häggström, Waernbaum, and De Luna (2013). The former paper provides the theoretical foundation for the covariate selection algorithms and the latter studies a data-driven implementation of the algorithms showing, e.g., how dimension reduction of the covariate vector can yield important mean squared error (MSE) decrease for commonly used non-parametric estimators. The data-driven covariate selection builds on marginal coordinate hypothesis tests (Cook 2004; Li, Cook, and Nachtsheim 2005) for continuous-valued covariate vectors, and on kernel smoothing with smoothing parameter thresholding when discrete and continuous-valued covariates are available (Li, Racine, and Wooldridge 2009).

In Section 2 we give a brief description of the theoretical framework. We also introduce a classic dataset, the LaLonde data, which is used to illustrate the purpose of the package. Then follows a description of the **CovSel** package in Section 3. In Section 4 we demonstrate the use of the package **CovSel** by performing covariate selection for the LaLonde data as well as in a number of different situations using simulated data.

2. Covariate selection when estimating average causal effects

2.1. Model

Assume that we have a random sample of individuals from a large population and want to estimate the average causal effect of a binary treatment T on an outcome Y. The Neyman-Rubin model (Splawa-Neyman 1990, which is the translated version of a paper published in 1923; Rubin 1974) is commonly used as a framework for causal reasoning and inference. For each individual in the study, it defines two potential outcomes $Y(1)$, outcome if treated, and $Y(0)$ outcome if not treated. Then, the individual causal effect is defined as $(Y(1) - Y(0))$. Because only one of the two potential outcomes can be observed for a given individual, individual causal effects are not identified. Instead, population parameters are targeted, e.g., the average causal effect

$$\beta = \mathsf{E}[Y(1) - Y(0)].$$

Let $\mathbf{X}$ be a set of covariates observed before treatment on all individuals in the study. Then, the identification of the parameter β, and other population parameters, can be obtained under the following assumptions.

Assumption 1 (Unconfoundedness) $T \perp\!\!\!\perp (Y(1), Y(0))|\mathbf{X}$.

Assumption 2 (Overlap) $0 < \mathsf{P}(T = 1|\mathbf{X}) < 1$.

Algorithm 1

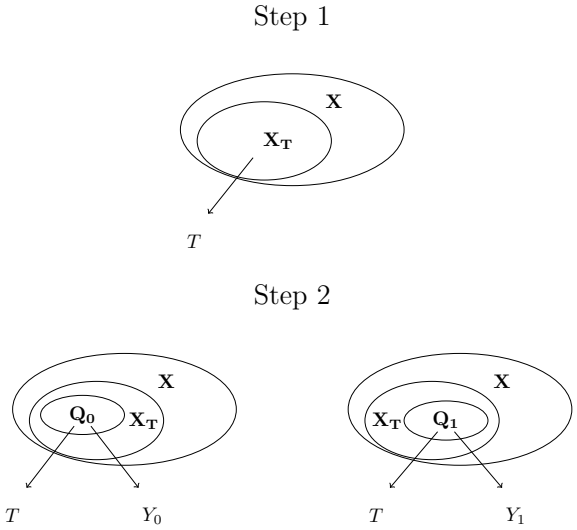

Figure 1: In Step 1 $\mathbf{X}_T$ such that $T \perp\!\!\!\perp \mathbf{X} \setminus \mathbf{X}_T | \mathbf{X}_T$ holds is identified. In Step 2, for $t = 0, 1$, $\mathbf{Q}_t \subseteq \mathbf{X}_T$ such that $Y_t \perp\!\!\!\perp \mathbf{X}_T \setminus \mathbf{Q}_t | \mathbf{Q}_t$ holds is identified.

Here $A \perp\!\!\!\perp B|C$ means A "is independent of" B given C (Dawid 1979). The unconfoundedness assumption holds when $\mathbf{X}$ consists of all the covariates affecting both the causal agent T and the potential outcomes $Y(1), Y(0)$. An example where the unconfoundedness assumption holds trivially are randomized experiments, where we have $T \perp\!\!\!\perp (Y(1), Y(0))$. In observational studies, the unconfoundedness assumption may be realistic in situations where many covariates $\mathbf{X}$ are available to condition on.

2.2. Algorithms and minimal sets of covariates

The covariates one wishes to identify are those which affect treatment as well as the potential outcomes and if Assumptions 1 and 2 hold it is possible to define the minimal sets of covariates such that the treatment and the potential outcomes are independent given these sets, see De Luna *et al.* (2011). Algorithms for covariate selection, denoted Algorithm 1 and Algorithm 2, are described in Figures 1 and 2, where the sets $\mathbf{X}_T$, $\mathbf{Q}_t$, $\mathbf{X}_t$ and $\mathbf{Z}_t$, $t = 0, 1$, are defined as follows: $\mathbf{X}_T$ is the minimal set of the complete covariate vector $\mathbf{X}$ rendering T and the covariates not included in $\mathbf{X}_T$ conditionally independent, $T \perp\!\!\!\perp \mathbf{X} \setminus \mathbf{X}_T | \mathbf{X}_T$. Similarly, $\mathbf{Q}_t$ is the minimal subset of $\mathbf{X}_T$ rendering Y_t and the covariates not included in $\mathbf{Q}_t$ conditionally independent, $Y_t \perp\!\!\!\perp \mathbf{X}_T \setminus \mathbf{Q}_t | \mathbf{Q}_t$. $\mathbf{X}_t$ is the minimal set of the complete covariate vector $\mathbf{X}$ rendering Y_t and the covariates not included in $\mathbf{X}_t$ conditionally independent, $Y_t \perp\!\!\!\perp \mathbf{X} \setminus \mathbf{X}_t | \mathbf{X}_t$, and $\mathbf{Z}_t$ is the minimal subset of $\mathbf{X}_t$ rendering T and the covariates not included in $\mathbf{X}_t$ conditionally independent, $T \perp\!\!\!\perp \mathbf{X}_t \setminus \mathbf{Z}_t | \mathbf{Z}_t$. For existence and uniqueness of the sets defined above see De Luna *et al.* (2011).

Algorithm 2

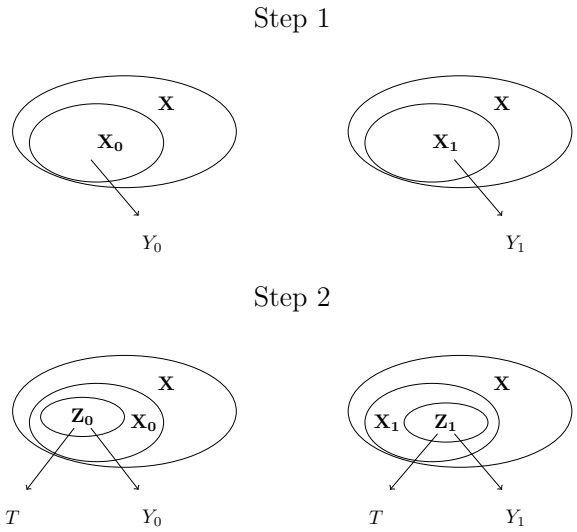

Figure 2: In Step 1, for $t = 0, 1$, $\mathbf{X}_t$ such that $Y_t \perp\!\!\!\perp \mathbf{X} \setminus \mathbf{X}_t | \mathbf{X}_t$ holds is identified. In Step 2, for $t = 0, 1$, $\mathbf{Z}_t \subseteq \mathbf{X}_t$ such that $T \perp\!\!\!\perp \mathbf{X}_t \setminus \mathbf{Z}_t | \mathbf{Z}_t$ holds is identified.

2.3. The LaLonde data

The LaLonde data was first analyzed in LaLonde (1986) and has since then been used numerous times (e.g., Dehejia and Wahba 1999; Smith and Todd 2005; Abadie and Imbens 2011). The data used in this paper is available on Dehejia's web page at `http://www.nber.org/~rdehejia/data/nswdata2.html` and consists of 297 treated units from a randomized evaluation of a labor training program, the national supported work (NSW) demonstration, and 314 non-experimental comparison units drawn from survey datasets. Below, using the text files from Dehejia's web page, we construct the data frame `lalonde` which will be used later on to demonstrate the selection of covariates for estimation of the average causal effect of participation in NSW on post-intervention earnings. The `lalonde` data frame is included in the **CovSel** package and can be accessed with the `data` command. Similar but not identical datasets are included in other R packages, e.g., **arm** (Gelman and Su 2015), **Matching** (Sekhon 2011), **MatchIt** (Ho, Imai, King, and Stuart 2011) and **cem** (Iacus, King, and Porro 2009). The following code is used to create the `lalonde` data frame included in package **CovSel**:

```
R> treated <- read.table(file = "nswre74_treated.txt")
R> controls <- read.table(file = "cps3_controls.txt")
R> nsw <- rbind(treated, controls)
R> ue <- function(x) factor(ifelse(x > 0, 0, 1))
R> UE74 <- mapply(ue, nsw[, 8])
R> UE75 <- mapply(ue, nsw[, 9])
R> nsw[, 4:7] <- lapply(nsw[, 4:7], factor)
R> lalonde <- cbind(nsw[, 1:9], UE74, UE75, nsw[, 10])
```

```
R> colnames(lalonde) <- c("treat", "age", "educ", "black", "hisp",
+    "married", "nodegr", "re74", "re75", "u74", "u75", "re78")
```

The data frame `lalonde` consists of 614 observations on 12 variables. The first ten rows are shown below.

```
R> lalonde[1:10, ]
```

	treat	age	educ	black	hisp	married	nodegr	re74	re75	u74	u75	re78
1	1	37	11	1	0	1	1	0	0	1	1	9930.0460
2	1	22	9	0	1	0	1	0	0	1	1	3595.8940
3	1	30	12	1	0	0	0	0	0	1	1	24909.4500
4	1	27	11	1	0	0	1	0	0	1	1	7506.1460
5	1	33	8	1	0	0	1	0	0	1	1	289.7899
6	1	22	9	1	0	0	1	0	0	1	1	4056.4940
7	1	23	12	1	0	0	0	0	0	1	1	0.0000
8	1	32	11	1	0	0	1	0	0	1	1	8472.1580
9	1	22	16	1	0	0	0	0	0	1	1	2164.0220
10	1	33	12	0	0	1	0	0	0	1	1	12418.0700

The first variable, `treat`, is the treatment status indicator variable, with `1` indicating participation in NSW. The next two variables are age in years (`age`) and schooling in years (`educ`). Next, `black`, `hisp`, `married` and `nodegr` are indicator variables (`1` = yes, `0` = no) for black, hispanic, marital status and high school diploma, respectively. Real earnings during the years 1974, 1975 and 1978 are given in `re74`, `re75`, `re78`, respectively, and `u74`, `u75` are indicator variables for earnings in 1974, 1975 being zero.

3. The R package CovSel

The R package **CovSel** contains the functions:

- `cov.sel` is the main function called by the user for selecting covariate sets.
- `cov.sel.np` is called by `cov.sel` if kernel smoothing should be used.
- `summary` method for '`cov.sel`' objects produces a summary of the results returned by `cov.sel`.

The package also contains the simulated data sets `datc`, `datf` and `datfc` which are described and analyzed in the examples in Section 4.

3.1. Function `cov.sel`

The function `cov.sel` can be used for reducing the dimension of the covariate vector in situations where we want to estimate an average causal effect and the unconfoundedness assumption holds. It is used as:

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

and takes the following arguments:

- `T` is a binary vector indicating the treatment status.
- `Y` is a numeric vector of observed outcomes.
- `X` is 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` is the type of method used, `"dr"` for marginal co-ordinate hypothesis tests and `"np"` for kernel-based smoothing. Marginal co-ordinate hypothesis tests are suitable in situations with only continuous covariates while kernel-based smoothing can be used if discrete covariates are also present.
- `alg` is used to specify which algorithm to use. `alg = 1` indicates Algorithm 1, `alg = 2` indicates Algorithm 2 and `alg = 3` runs them both. `alg = 3` is default.
- `scope` is a character string giving the name of one (or several) covariate(s) that must not be removed.
- `alpha` is a stopping criterion for the marginal co-ordinate hypothesis tests, i.e., the algorithm 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` is the 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`.
- `thro` is the bandwidth threshold used for ordered discrete covariates if `type = "np"`. Values in [0, 1] are valid. `thro = 0` removes all ordered discrete covariates and `thro = 1` removes none of them. Default is `thro = 0.25`.
- `thrc` is the bandwidth threshold used for continuous covariates if `type = "np"`. Non-negative values are valid. Default is `thr = 100`.
- `...` are additional arguments passed on to `dr` or `npregbw`. If `type = "dr"`, `method` can be set to `"sir"` or `"save"`, the first being the default. `trace = 0` suppresses the output generated by `dr.step`. If `type = "np"`, `regtype` can be set to `"lc"` or `"ll"`, the first being the default and `bwtype` can be set to `"fixed"`, `"generalized_nn"` or `"adaptive_nn"`, where default is `"fixed"`. See `dr` and `npregbw` for usage of `na.action`.

If `type = "dr"`, marginal co-ordinate hypothesis tests are performed in each step of the algorithm using the function `dr` from package **dr** (Weisberg 2002). If on the other hand `type = "np"` then non-parametric kernel regression is repeatedly performed, using the function `npregbw` from package **np** (Hayfield and Racine 2008), to determine which covariates can be removed from the full covariate set.

With `dr` one can choose between sliced inverse regression, `"sir"`, or sliced average variance estimation, `"save"`. Both are methods based on studying an inverse regression problem, where the former considers dependencies only through the first moment (the mean) while

the latter looks at the second moment (Cook and Weisberg 1991). Though `"save"` is more general than `"sir"`, it may miss first moment information, e.g., linear trends. Thus, one may want to use both to see if they result in different choices of covariate sets. For kernel-based smoothing the regression type can be set to using a local constant or local linear kernel and the bandwidth type can be set to fixed, generalized nearest neighbors or adaptive nearest neighbors. See `dr` and `npregbw` for details.

In kernel-based smoothing the bandwidth range for an unordered discrete covariate `x` is `0` to `1/length(levels(x))`, while for ordered discrete covariates, no matter how many levels, the range is `0` to `1`. For continuous covariates the bandwidth ranges from `0` to `infinity`. Ordered discrete and continuous covariates are removed if their bandwidths exceed their respective thresholds (`thro` and `thrc`). Unordered discrete covariates are removed if their bandwidths are larger than `thru` times the maximum bandwidth.

Since cross-validation is used to select bandwidths for the kernel smoothing this is computationally intensive and a doubling of the sample size will increase the run time of `npregbw` by a factor of four. This means that `cov.sel` with `type = "np"` will be slow for large sample sizes since `npregbw` is called multiple times. The computation time can be reduced by setting some of the arguments in `npregbw` to non-default values. For more on this subject the reader is referred to Hayfield and Racine (2008) and to the frequently asked questions document on Jeffrey S. Racine's website (`http://socserv.mcmaster.ca/racine/np_faq.pdf`).

The default values for the thresholds `thru`, `thro`, `thrc` and `alpha` are arbitrary. On the other hand, we know that the bandwidth for relevant covariates will tend to zero as sample size increases (Li *et al.* 2009). The thresholds used should therefore be smaller the larger the sample sizes in order to avoid selecting irrelevant covariates. This said, the default values for the thresholds have shown to yield good results in simulations studies reported in Persson *et al.* (2013), for a wide range of situations and sample sizes of 500 and 1000. In applications, we recommend the user to investigate the sensitivity of the results to small changes in the threshold values.

4. Examples

We illustrate the use of the main function `cov.sel` for different data situations. We begin by selecting covariates in the simulated data sets and then move on to the LaLonde data. Three simulated data sets, `datc`, `datfc` and `datf`, are included in package **CovSel** and can be accessed with the `data` command, see below.

4.1. Continuous-valued covariates

Let us first install and load package **CovSel** in R.

```
R> install.packages("CovSel")
R> library("CovSel")
```

```
Loading required package: dr
Loading required package: MASS
Loading required package: np
Nonparametric Kernel Methods for Mixed Datatypes (version 0.60-2)
[vignette("np_faq",package="np") provides answers to frequently asked questions]
```

As can be seen above the **CovSel** package depends on and loads the additional packages **dr** (Weisberg 2002), **MASS** (Venables and Ripley 2002) and **np** (Hayfield and Racine 2008).

The data in `datc` contains ten normally distributed covariates as well as the potential outcomes, the treatment indicator T and the response Y. It is generated by

```
R> set.seed(9327529)
R> n <- 1000
R> eta <- mvrnorm(n, rep(0, 2), diag(1, 2, 2))
R> Sigma <- diag(1, 10, 10)
R> Sigma[7, 8] <- Sigma[8, 7] <- 0.5
R> X <- mvrnorm(n, rep(0, 10), Sigma)
R> y0 <- 2 + 2 * X[, 1] + 2 * X[, 2] + 2 * X[, 5] + 2 * X[, 6] +
+    2 * X[, 8] + eta[, 1]
R> y1 <- 4 + 2 * X[, 1] + 2 * X[, 2] + 2 * X[, 5] + 2 * X[, 6] +
+    2 * X[, 8] + eta[, 2]
R> e <- 1 / (1 + exp( - 0.5 * X[, 1] - 0.5 * X[, 2] - 0.5 * X[, 3] -
+    0.5 * X[, 4] - 0.5 * X[, 7]))
R> T <- rbinom(n, 1, e)
R> y <- y1 * T + y0 * (1 - T)
R> 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)
```

Since this data only contains continuous covariates we can use marginal co-ordinate hypothesis tests in the selection algorithms, which is set by `type = "dr"`. Using the default values (and adding `trace = 0` to suppress the extra output generated by `dr.step`), covariate selection is performed as follows

```
R> ans <- cov.sel(T = datc$T, Y = datc$y, X = datc[, 1:10], type = "dr",
+    alg = 3, scope = NULL, alpha = 0.1, trace = 0)
R> ans

$X.T
[1] "x1" "x2" "x3" "x4" "x7"

$Q.0
[1] "x1" "x2" "x3" "x7"

$Q.1
[1] "x1" "x2" "x4" "x7"

$X.0
[1] "x1"  "x2"  "x5"  "x6"  "x8"  "x10"

$X.1
[1] "x1" "x2" "x5" "x6" "x8" "x9"
```

```
$Z.0
[1] "x1" "x2" "x8"

$Z.1
[1] "x1" "x2" "x8"

$evectorsQ.0
         Dir1       Dir2       Dir3       Dir4
x1 0.68345145  0.5336071  0.1570979  0.4333477
x2 0.59615208 -0.1855380  0.1490656 -0.7798567
x3 0.05556844  0.5080012 -0.8713177 -0.2648708
x7 0.41762299 -0.6502106 -0.4403465  0.3658916

$evectorsQ.1
          Dir1        Dir2        Dir3       Dir4
x1  0.65122801 -0.28351936 -0.03334562  0.6525600
x2  0.70513331 -0.08037438  0.24224417 -0.6855881
x4 -0.04685983 -0.89753273 -0.37176312 -0.2161204
x7  0.27657413  0.32801177 -0.89554342 -0.2396380

$evectorsZ.0
        Dir1       Dir2         Dir3
x1 0.7388869 -0.6469896  0.008586759
x2 0.6409990  0.7253482  0.332497832
x8 0.2077653  0.2351052 -0.943064929

$evectorsZ.1
        Dir1       Dir2         Dir3
x1 0.7388869 -0.6469896  0.008586759
x2 0.6409990  0.7253482  0.332497832
x8 0.2077653  0.2351052 -0.943064929

$method
[1] "sir"

$covar
 [1] "x1"  "x2"  "x3"  "x4"  "x5"  "x6"  "x7"  "x8"  "x9"  "x10"

attr(,"class")
[1] "cov.sel"
```

Applying the `summary` function to the '`cov.sel`' object gives us information on the different covariate sets defined in Section 2.

```
R> summary(ans)

 Original covariate vector:
 x1 x2 x3 x4 x5 x6 x7 x8 x9 x10
```

```
Minimal subsets of the covariate vector:
Q.0  =  x1 x2 x3 x7
Q.1  =  x1 x2 x4 x7
Z.0  =  x1 x2 x8
Z.1  =  x1 x2 x8

Removed variables:
Q.0comp  =  x4 x5 x6 x8 x9 x10
Q.1comp  =  x3 x5 x6 x8 x9 x10
Z.0comp  =  x3 x4 x5 x6 x7 x9 x10
Z.1comp  =  x3 x4 x5 x6 x7 x9 x10

method  =  sir
```

From the code generating the data we can see that the true subsets in Algorithm 1 are $\mathbf{X}_T = \{X_1, X_2, X_3, X_4, X_7\}$ and $\mathbf{Q}_0 = \mathbf{Q}_1 = \{X_1, X_2, X_7\}$. Similarly, the true subsets in Algorithm 2 are $\mathbf{X}_0 = \mathbf{X}_1 = \{X_1, X_2, X_5, X_6, X_8\}$ and $\mathbf{Z}_0 = \mathbf{Z}_1 = \{X_1, X_2, X_8\}$. If we compare the covariate sets selected by `cov.sel` we see that none of the covariates that theoretically should be included is removed in this case.

Increasing the stopping criterion value to 0.3 and changing the method from the default `"sir"` to `"save"` results in the following covariate sets

```
R> ans <- cov.sel(T = datc$T, Y = datc$y, X = datc[, 1:10], type = "dr",
+    alg = 3, scope = NULL, alpha = 0.3, trace = 0, method = "save")
R> ans

$X.T
[1] "x1" "x3" "x4" "x8"

$Q.0
[1] "x1" "x3" "x8"

$Q.1
[1] "x1" "x4" "x8"

$X.0
[1] "x1" "x2" "x5" "x6" "x8"

$X.1
[1] "x1" "x2" "x5" "x6" "x8"

$Z.0
[1] "x1" "x8"

$Z.1
[1] "x1" "x8"
```

```
$evectorsQ.0
         Dir1        Dir2       Dir3
x1 -0.5408152  0.53814880  0.6893291
x3  0.3384066  0.84193897 -0.4430845
x8 -0.7700649 -0.03917445 -0.5731506

$evectorsQ.1
         Dir1       Dir2        Dir3
x1 -0.4946413  0.8030985 -0.32314659
x4  0.1384715 -0.2132580 -0.94107811
x8 -0.8579951 -0.5563756 -0.09974101

$evectorsZ.0
         Dir1       Dir2
x1 -0.8608260  0.5135257
x8 -0.5088994 -0.8580742

$evectorsZ.1
         Dir1       Dir2
x1 -0.8608260  0.5135257
x8 -0.5088994 -0.8580742

$method
[1] "save"

$covar
 [1] "x1"  "x2"  "x3"  "x4"  "x5"  "x6"  "x7"  "x8"  "x9"  "x10"

attr(,"class")
[1] "cov.sel"
```

The increased stopping criterion value would under `"sir"` have led to larger subsets but here we see that this larger value in combination with `"save"` resulted in subsets smaller than in the previous example. In this case the returned final subsets are too small, in that unconfoundedness is no longer upheld given these subsets.

4.2. Categorical covariates

The data in `datf` contains eight binary covariates as well as the potential outcomes, the treatment indicator T and the response Y. It is generated, using the package **bindata** (Leisch, Weingessel, and Hornik 2012), as follows

```
R> library(bindata)
R> set.seed(9327529)
R> n<-500
R> x1 <- rbinom(n, 1, prob = 0.5)
```

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

Since this data only contains categorical covariates we use kernel smoothing in the selection algorithms, which is set by `type = "np"`. Using the default values, covariate selection is performed as follows

```
R> ans <- cov.sel(T = datf$T, Y = datf$y, X = datf[, 1:8], type = "np",
+    alg = 3, scope = NULL, alpha = 0.1, thru = 0.5, thro = 0.25, thrc = 100)
R> ans

$X.T
[1] "x1" "x2" "x3" "x5" "x8"

$Q.0
[1] "x1" "x2" "x3" "x5"

$Q.1
[1] "x1" "x3" "x5"

$X.0
[1] "x1" "x4" "x5" "x6"

$X.1
[1] "x1" "x4" "x5" "x6"

$Z.0
[1] "x1" "x4" "x5"
```

```
$Z.1
[1] "x1" "x4" "x5"

$bandwidthsQ.0
[1] 0.020341307 0.231325017 0.017142417 0.003798791

$bandwidthsQ.1
[1] 6.569470e-03 1.417726e-02 3.371529e-09

$bandwidthsZ.0
[1] 0.04257388 0.05589723 0.05354814

$bandwidthsZ.1
[1] 0.04257380 0.05589683 0.05354823

$regtype
[1] "Local-Constant"

$bwtype
[1] "fixed"

$covar
[1] "x1" "x2" "x3" "x4" "x5" "x6" "x7" "x8"

attr(,"class")
[1] "cov.sel"
```

Here, X_1, X_2, X_3, X_4, X_5 and X_8 are all correlated with and thus affect T. However, X_2 and X_5 as well as X_3 and X_4 are strongly correlated and the correlation between X_2 and T is larger than the correlation between X_5 and T, similarly, the correlation between X_3 and T is larger than the correlation between X_4 and T. In light of this, we almost have that $T \perp\!\!\!\perp X_5 | X_2$ and similarly that $T \perp\!\!\!\perp X_4 | X_3$. This results in $\mathbf{X}_T = \{X_1, X_2, X_3, X_8\}$ and $\mathbf{Q}_0 = \mathbf{Q}_1 = \{X_1, X_2, X_3\}$ which is almost what is selected by `cov.sel`, the exceptions are that X_5 is included instead of X_2 in `Q.1` and in addition to the true sets in `X.T` and `Q.0`. The true subsets in Algorithm 2 are $\mathbf{X}_0 = \mathbf{X}_1 = \{X_1, X_4, X_5, X_6\}$ and $\mathbf{Z}_0 = \mathbf{Z}_1 = \{X_1, X_4, X_5\}$ and these are recovered by `cov.sel`.

4.3. Mixed valued covariates

The data in `datfc` contains four normally distributed covariates, four binary covariates as well as the potential outcomes, the treatment indicator T and the response Y. It is generated by:

```
R> set.seed(9327529)
R> n<-500
R> x1 <- rnorm(n, mean = 0, sd = 1)
R> x2 <- rbinom(n, 1, prob = 0.5)
```

```
R> x25 <- rmvbin(n, bincorr=cbind(c(1,0.7),c(0.7,1)), margprob=c(0.5,0.5))
R> x2 <- x25[,1]
R> Sigma <- matrix(c(1,0.5,0.5,1),ncol=2)
R> x34 <- mvrnorm(n, rep(0, 2), Sigma)
R> x3 <- x34[,1]
R> x4 <- x34[,2]
R> x5 <- x25[,2]
R> x6 <- rbinom(n, 1, prob = 0.5)
R> x7<- rnorm(n, mean = 0, sd = 1)
R> x8 <- rbinom(n, 1, prob = 0.5)
R> e0<-rnorm(n)
R> e1<-rnorm(n)
R> p <- 1/(1 + exp(3 - 1.2 * x1 - 3.7 * x2 - 1.5 * x3 - 0.3 * x4 - 0.3 * x5 -
+        1.9 * x8))
R> T <- rbinom(n, 1, prob = p)
R> y0 <- 4 + 2 * x1 + 3 * x4 + 5 * x5 + 2 * x6 + e0
R> y1 <- 2 + 2 * x1 + 3 * x4+ 5 * x5 + 2 * x6 + e1
R> y <- y1 * T + y0 * (1 - T)
R> datfc <- data.frame(x1, x2, x3, x4, x5, x6, x7, x8, y0, y1, y, T)
R> datfc[, c(2, 5, 6, 8)] <- lapply(datfc[, c(2, 5, 6, 8)], factor)
R> datfc[, 12] <- as.numeric(datfc[, 12])
```

As in Section 4.3, we use kernel smoothing in the selection algorithms as this data contains categorical covariates.

```
R> ans <- cov.sel(T = datfc$T, Y = datfc$y, X = datfc[, 1:8], type = "np",
+    alg = 3, scope = NULL, alpha = 0.1, thru = 0.5, thro = 0.25, thrc = 100)
R> ans

$X.T
[1] "x1" "x2" "x3" "x4" "x7" "x8"

$Q.0
[1] "x1" "x2" "x4"

$Q.1
[1] "x1" "x2" "x4" "x8"

$X.0
[1] "x1" "x4" "x5" "x6"

$X.1
[1] "x1" "x4" "x5" "x6"

$Z.0
[1] "x1" "x4" "x5"
```

```
$Z.1
[1] "x1" "x4" "x5"

$bandwidthsQ.0
[1] 0.425579270 0.002582504 0.324867709

$bandwidthsQ.1
[1] 0.27246792 0.00594458 0.33618826 2.10872755

$bandwidthsZ.0
[1] 0.54726004 0.56496781 0.01344475

$bandwidthsZ.1
[1] 0.54726131 0.56496582 0.01344468

$regtype
[1] "Local-Constant"

$bwtype
[1] "fixed"

$covar
[1] "x1" "x2" "x3" "x4" "x5" "x6" "x7" "x8"

attr(,"class")
[1] "cov.sel"
```

Similarly to the data in Section 4.3, X_1, X_2, X_3, X_4, X_5 and X_8 are all correlated with T but X_4 and X_5 affect T mainly through X_2 and X_3. Using the same reasoning as in Section 4.3 we have that $\mathbf{X}_T = \{X_1, X_2, X_3, X_8\}$, $\mathbf{Q}_0 = \mathbf{Q}_1 = \{X_1, X_2, X_3\}$, $\mathbf{X}_0 = \mathbf{X}_1 = \{X_1, X_4, X_5, X_6\}$ and $\mathbf{Z}_0 = \mathbf{Z}_1 = \{X_1, X_4, X_5\}$. Here, the subsets returned by Algorithm 2 is identical to the true subsets. Algorithm 1 returned subsets of reduced dimension, sufficient for unconfoundedness to hold, although none of the returned subsets equals the true subsets.

4.4. Real data: LaLonde

For the LaLonde data we set the arguments in `cov.sel` as follows: `T` is set equal to `treat` and `Y` to `re78`, columns 1 and 12 in `lalonde`, respectively. The rest of the variables in the data frame are given as covariates, `X`. Since both continuous (`age`, `educ`) and categorical covariates are included we set `type` equal to `"np"`. We begin with running Algorithm 1, `alg = 1`, with the default threshold values for the bandwidths, and store the result in `cs`.

```
R> cs <- cov.sel(T = lalonde[, 1], Y = lalonde[, 12], X = lalonde[, 2:11],
+    type = "np", alg = 1, thru = 0.5, thro = 0.25, thrc = 100)
```

Looking at the resulting covariate selection we see that in Step 1 `X.T` includes `age`, `black`, `nodegr`, `u74` and `u75` and in Step 2 the subsets `Q.0` and `Q.1` both consist of `age`, `nodegr`, `u74` and `u75`.

```
R> cs

$X.T
[1] "age"    "black"  "nodegr" "u74"    "u75"

$Q.0
[1] "age"    "nodegr" "u74"    "u75"

$Q.1
[1] "age"    "nodegr" "u74"    "u75"

$bandwidthsQ.0
[1] 4.289575022 0.211009655 0.087257373 0.002986449

$bandwidthsQ.1
[1] 32.14950550  0.16942699  0.06907088  0.08516276

$regtype
[1] "lc"

$bwtype
[1] "fixed"

$covar
 [1] "age"     "educ"    "black"   "hisp"    "married" "nodegr"  "re74"
 [8] "re75"    "u74"     "u75"

attr(,"class")
[1] "cov.sel"
```

Next we change to Algorithm 2, `alg = 2`, but leave everything else unchanged:

```
R> cs <- cov.sel(T = lalonde[, 1], Y = lalonde[, 12], X = lalonde[, 2:11],
+    type = "np", alg = 2, thru = 0.5, thro = 0.25, thrc = 100)
```

Here, in Step 1 the covariates `age`, `black`, `educ`, `hisp`, `nodegr` and `u74` are retained in `X.0` while `X.1` includes `educ`, `hisp`, `married` and `nodegr`. In Step 2 these sets remain unchanged, i.e., `Z.0` = `X.0` and `Z.1` = `X.1`; see De Luna *et al.* (2011) and Section 5 for a discussion on differences between Algorithms 1 and 2.

```
> cs

$X.0
[1] "age"    "black"  "educ"   "hisp"   "nodegr" "u74"

$X.1
[1] "educ"    "hisp"    "married" "nodegr"
```

```
$Z.0
[1] "age"    "black"  "educ"   "hisp"   "nodegr" "u74"

$Z.1
[1] "educ"    "hisp"    "married" "nodegr"

$bandwidthsZ.0
[1] 2.188735e+00 4.500190e-02 2.197854e+00 8.929507e-02 1.942976e-14
[6] 6.620783e-02

$bandwidthsZ.1
[1] 1.05345229 0.01691264 0.06745027 0.09992094

$regtype
[1] "lc"

$bwtype
[1] "fixed"

$covar
 [1] "age"     "educ"    "black"   "hisp"    "married" "nodegr"  "re74"
 [8] "re75"    "u74"     "u75"

attr(,"class")
[1] "cov.sel"
```

5. Summary

We have described the R package **CovSel** which implements the covariate selection algorithms developed in De Luna *et al.* (2011) and Persson *et al.* (2013) using model-free backward elimination. The use of the package has been illustrated using a classic real world dataset as well as simulated data. Controlling for confounding is an essential part in all evaluation studies of effects of non-randomized treatments. Observational studies with a rich reservoir of covariates are nowadays common, for instance with record linkage databases. Package **CovSel** has the potential to help empirical scientists performing such evaluation studies by identifying relevant covariate sets for estimating average causal effects.

Further insights on how to use package **CovSel** can be found in Persson *et al.* (2013), where a large set of simulations experiments are reported. There, different selected covariate sets are evaluated by comparing empirical bias, variance and MSE in the estimation of the average causal effect. In particular, this Monte Carlo study indicates that Algorithm 2 is to be preferred to Algorithm 1, the former yielding often lower MSE. Moreover, basing the estimation on the covariate set $\mathbf{X}_0 \cup \mathbf{X}_1$ yields lower variance but not necessarily lower bias, and either $\mathbf{X}_0 \cup \mathbf{X}_1$ or $\mathbf{Z}_0 \cup \mathbf{Z}_1$ may yield lower MSE depending on the data generating mechanism. These finite sample properties are in line with the theoretical results derived earlier in De Luna *et al.* (2011).

Acknowledgments

The Swedish Research Council and the Riksbankens Jubileumsfond are acknowledged for their financial support.

References

Abadie A, Imbens G (2011). "Bias-Corrected Matching Estimators for Average Treatment Effects." *Journal of Business & Economic Statistics*, **29**(1), 1–11. doi:10.1198/jbes.2009.07333.

Cook RD (2004). "Testing Predictor Contributions in Sufficent Dimension Reduction." *The Annals of Statistics*, **32**(3), 1061–1092. doi:10.1214/009053604000000292.

Cook RD, Weisberg S (1991). "Sliced Inverse Regression for Dimension Reduction: Comment." *Journal of the American Statistical Association*, **86**(414), 328–332. doi:10.2307/2290564.

Dawid A (1979). "Conditional Independence in Statistical Theory." *Journal of the Royal Statistical Society B*, **41**(1), 1–31.

De Luna X, Waernbaum I, Richardson T (2011). "Covariate Selection for the Non-Parametric Estimation of an Average Treatment Effect." *Biometrika*, **98**(4), 861–875. doi:10.1093/biomet/asr041.

Dehejia R, Wahba S (1999). "Causal Effects in Nonexperimental Studies: Reevaluating the Evaluation of Training Programs." *Journal of the American Statistical Association*, **94**(448), 1053–1062. doi:10.1080/01621459.1999.10473858.

Gelman A, Su YS (2015). ***arm**: Data Analysis Using Regression and Multilevel/Hierarchical Models*. R package version 1.8-6, URL http://CRAN.R-project.org/package=arm.

Häggström J, Persson E (2015). ***CovSel**: Model-Free Covariate Selection*. R package version 1.2.1, URL http://CRAN.R-project.org/package=CovSel.

Hahn J (2004). "Functional Restriction and Efficiency in Causal Inference." *The Review of Economics and Statistics*, **86**(1), 73–76. doi:10.1162/003465304323023688.

Hayfield T, Racine J (2008). "Nonparametric Econometrics: The **np** Package." *Journal of Statistical Software*, **27**(5), 1–32. doi:10.18637/jss.v027.i05.

Ho D, Imai K, King G, Stuart EA (2011). "**MatchIt**: Nonparametric Preprocessing for Parametric Causal Inference." *Journal of Statistical Software*, **42**(8), 1–28. doi:10.18637/jss.v042.i08.

Iacus S, King G, Porro G (2009). "**cem**: Software for Coarsened Exact Matching." *Journal of Statistical Software*, **30**(9), 1–27. doi:10.18637/jss.v030.i09.

Imbens G, Wooldridge J (2009). "Recent Developments in the Econometrics of Program Evaluation." *Journal of Economic Literature*, **47**(1), 5–86. doi:10.1257/jel.47.1.5.

Laan M, Gruber S (2010). "Collaborative Double Robust Targeted Maximum Likelihood Estimation." *International Journal of Biostatistics*, **6**(1), Article 17. doi:10.2202/1557-4679.1181.

LaLonde R (1986). "Evaluating the Econometric Evaluations of Training Programs with Experimental Data." *The American Economic Review*, **76**(4), 604–620.

Leisch F, Weingessel A, Hornik K (2012). ***bindata**: Generation of Artificial Binary Data*. R package version 0.9-19, URL http://CRAN.R-project.org/package=bindata.

Li L, Cook RD, Nachtsheim C (2005). "Model-Free Variable Selection." *Journal of the Royal Statistical Society B*, **67**(2), 285–299. doi:10.1111/j.1467-9868.2005.00502.x.

Li Q, Racine J, Wooldridge J (2009). "Efficient Estimation of Average Treatment Effects with Mixed Categorical and Continuous Data." *Journal of Business & Economic Statistics*, **27**(2), 206–223. doi:10.1198/jbes.2009.0015.

Persson E, Häggström J, Waernbaum I, De Luna X (2013). "Data-Driven Algorithms for Dimension Reduction in Causal Inference: Analyzing the Effect of School Achievements on Acute Complications of Type 1 Diabetes Mellitus." arXiv:1309.4054 [stat.ME], URL http://arxiv.org/abs/1309.4054.

R Core Team (2015). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. URL http://www.R-project.org/.

Robins J, Rotnitzky A (1995). "Semiparametric Efficiency in Multivariate Regression Models with Missing Data." *Journal of the American Statistical Association*, **90**(429), 122–129. doi:10.1080/01621459.1995.10476494.

Rubin D (1974). "Estimating Causal Effects of Treatments in Randomized and Nonrandomized Studies." *Journal of Educational Psychology*, **66**(5), 688–701. doi:10.1037/h0037350.

Rubin D (1997). "Estimating Causal Effects from Large Data Sets Using Propensity Scores." *The Annals of Internal Medicine*, **127**(8), 757–763. doi:10.7326/0003-4819-127-8_Part_2-199710151-00064.

Sekhon J (2011). "Multivariate and Propensity Score Matching Software with Automated Balance Optimization: The **Matching** Package for R." *Journal of Statistical Software*, **42**(7), 1–52. doi:10.18637/jss.v042.i07.

Smith J, Todd P (2005). "Does Matching Overcome LaLonde's Critique of Nonexperimental Estimators?" *Journal of Econometrics*, **125**(1–2), 305–353. doi:10.1016/j.jeconom.2004.04.011.

Splawa-Neyman J (1990). "On the Application of Probability Theory to Agricultural Experiments. Essay on Principles. Section 9." *Statistical Science*, **5**(4), 465–472. Translated and edited by D. M. Dabrowska and T. P. Speed from the Polish original, which appeared in Roczniki Nauk Rolniczych Tom X (1923) 1–51 (Annals of Agricultural Sciences).

Vansteelandt S, Bekaert M, Claeskens G (2012). "On Model Selection and Model Misspecification in Causal Inference." *Statistical Methods in Medical Research*, **21**(1), 7–30. doi:10.1177/0962280210387717.

Venables W, Ripley B (2002). *Modern Applied Statistics with S*. 4th edition. Springer-Verlag, New York. doi:10.1007/978-0-387-21706-2.

Weisberg S (2002). "Dimension Reduction Regression in R." *Journal of Statistical Software*, **7**(1), 1–22. doi:10.18637/jss.v007.i01.

Affiliation:

Jenny Häggström
Department of Statistics, USBE
Umeå University
901 87, Umeå, Sweden
E-mail: jenny.haggstrom@umu.se
URL: http://www.usbe.umu.se/om/personal/jeyhom02/

Journal of Statistical Software http://www.jstatsoft.org/
published by the Foundation for Open Access Statistics http://www.foastat.org/
November 2015, Volume 68, Issue 1 *Submitted:* 2013-10-09
doi:10.18637/jss.v068.i01 *Accepted:* 2014-12-29