A Vignette for ATE Estimation

Zhiqiang Tan[1]

February 2019 (revised October 2020)

1 Introduction

The R package `RCAL` (version 2.0) can be used for three main tasks:

- to estimate the mean of an outcome in the presence of missing data,
- to estimate the average treatment effects (ATE) in causal inference,
- to estimate the local average treatment effects (LATE) in causal inference.

There are 3 high-level functions provided for the first task:

- `mn.nreg`: inference using non-regularized calibrated estimation,
- `mn.regu.cv`: inference using regularized calibrated estimation based on cross validation,
- `mn.regu.path`: inference using regularized calibrated estimation along a regularization path.

The first function `mn.nreg` is appropriate only in relatively low-dimensional settings, whereas the functions `mn.regu.cv` and `mn.regu.path` are designed to deal with high-dimensional data (namely, the number of covariates close to or greater than the sample size). In parallel, there are 3 functions for estimating the ATE in the second task, `ate.nreg`, `ate.regu.cv`, and `ate.regu.path`. These functions can also be used to perform inference for the average treatment effects on the treated or on the untreated. Currently, the treatment is assumed to be binary (i.e., untreated or treated). Extensions to multi-valued treatments will be incorporated in later versions. Estimation of LATE is discussed in a separate vignette.

The package also provides lower-level functions, including `glm.nreg` to implement non-regularized M-estimation and `glm.regu` to implement Lasso regularized

[1]Department of Statistics, Rutgers University. Address: 110 Frelinghuysen Road, Piscataway, NJ 08854. E-mail: ztan@stat.rutgers.edu.

M-estimation for fitting generalized linear models currently with continuous or binary outcomes. The latter function `glm.regu` uses an active-set descent algorithm, which enjoys a finite termination property for solving least-squares Lasso problems.

2 An example

We illustrate the use of the package on a simulated dataset as in Tan (2020b), Section 4. The dataset, `simu.data`, is included as part of the package.

```
> library(RCAL)
> data(simu.data)
```

The following shows the first 10 rows and the first 6 columns of the dataset, which is of size 800×202.

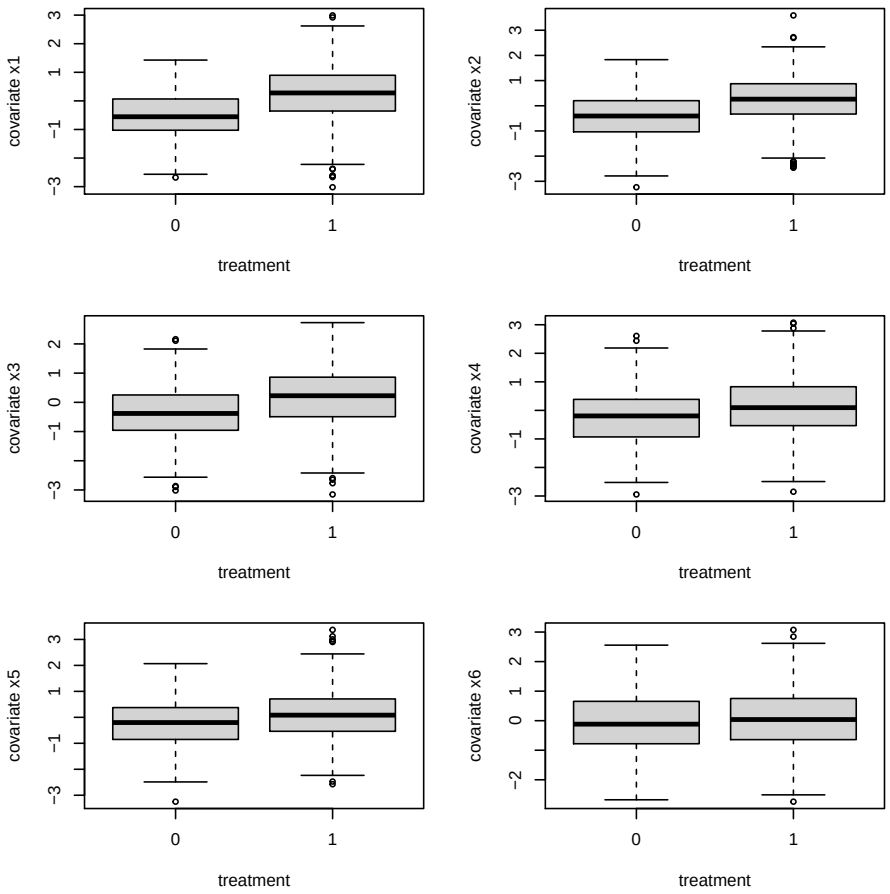
```
> simu.data[1:10, 1:6]
```

```
               y tr
 [1,]  0.6032936  1  0.9616951  0.5962625  0.4492321  0.8512571217
 [2,]  4.0070254  1  2.3647316  0.3364174  1.8668110  1.1255817363
 [3,]  1.3576326  1  0.1563721 -0.2232653  0.4548950  0.9605372221
 [4,]  0.9133211  1  0.2015582  1.4384546 -0.1350395  1.5087188960
 [5,]  2.5373546  1  1.2350352  1.5431279  1.0112577  0.3531083965
 [6,]  3.0240024  1  0.6151206  1.0198704  1.6962577 -0.2750277816
 [7,] -1.3507495  0 -1.7719684 -0.6344480 -1.2890632 -1.5098428847
 [8,] -2.4126343  0 -1.6242547 -0.1267854  0.7021270 -0.0009029026
 [9,] -1.1280024  0  0.1138376 -0.5874306  0.7784352 -0.4486271136
[10,] -1.0314948  0 -0.2667515  0.4259913 -0.3408476 -0.0722745940
```

The first column represents an observed outcome `y`, the second column represents a binary treatment `tr`, and the remaining 200 columns represent covariates.

```
> n <- dim(simu.data)[1]
> p <- 100 # include the first 100 covariates due to CRAN time constraint
```

Figure 1: Boxplots of covariates in the untreated and treated groups.



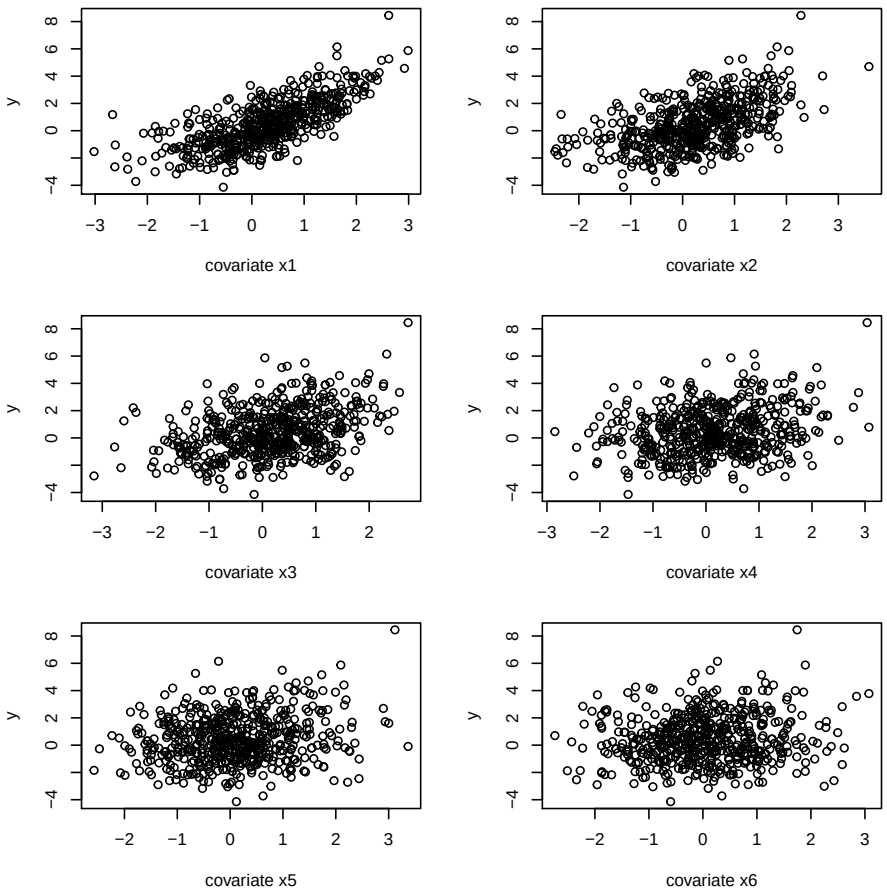

```
> y <- simu.data[,1]
> tr <- simu.data[,2]
> x <- simu.data[,2+1:p]
> x <- scale(x)
```

To examine the data, Figure 1 shows the boxplots of the first 6 covariates in the untreated and treated groups, and Figure 2 shows the scatterplots of observed outcomes and the first 6 covariates in the treated group.

2.1 Estimation of a population mean with missing data

We use the potential outcome framework for causal inference (Neyman 1923; Rubin 1974). For each individual i, the potential outcome Y_i^1 with the treatment is the

Figure 2: Scatterplots of observed outcomes and covariates in the treated group.



observed outcome Y_i if treatment variable T_i is 1, or missing otherwise. Similarly, the potential outcome Y_i^0 without the treatment is observed if treatment variable T_i is 0, or missing otherwise. To estimate the means of the potential outcomes amounts to estimation of population means with missing data.

In this section, we consider the problem of estimating the mean μ^1 of potential outcomes Y_i^1 with the treatment, which are observed when T_i is 1 but missing otherwise. The covariates X_i are observed on all individuals in the sample, and can be relevant to the estimation of μ^1 in two distinct ways. On one hand, the covariates X_i can be associated with the treatment variable T_i. In other words, individuals with different covariates may differ in their probabilities of receiving the treatment, which are denoted as $\pi(X_i)$ and called propensity scores (Rosenbaum and Rubin 1983). On

the other hand, the covariates X_i can also be associated with the outcome variable Y_i in the treated group $\{T_i = 1\}$. The conditional mean of Y_i given X_i and $T_i = 1$ is called the outcome regression function in the treated and denoted as $m^1(X_i)$. These associations can be seen from Figures 1 and 2.

Ignoring the covariates and using the simple sample average of observed outcomes in the treated yield an estimate 0.47 with standard error 0.076. This inference would be biased, since the true value of μ^1 is 0 by the design of the simulated data, as described in `help(simu.data)`.

```
> mean(y[tr==1])   # point estimate

[1] 0.4706937

> sqrt(var(y[tr==1]) / sum(tr) )   # standard error

[1] 0.07643911
```

The function `mn.regu.cv` implements a two-step method for estimating μ^1. First, propensity score and outcome regression models are fitted. Denote by $\hat{\pi}^1(X_i)$ and $\hat{m}^1(X_i)$ the fitted propensity score and outcome regression function respectively. Then the augmented IPW estimator of μ^1 is applied (Robins et al. 1994):

$$\hat{\mu}^1_{\text{AIPW}} = \frac{1}{n} \sum_{i=1}^{n} \left[\frac{T_i Y_i}{\hat{\pi}^1(X_i)} - \left\{ \frac{T_i}{\hat{\pi}^1(X_i)} - 1 \right\} \hat{m}^1(X_i) \right].$$

For `ploss`="cal", regularized calibrated estimation is performed with cross validation as in Tan (2020a, 2020b). The method then leads to model-assisted inference, in which confidence intervals are valid with high-dimensional data if the propensity score model is correctly specified but the outcome regression model may be misspecified. With linear outcome models, the inference is also doubly robust. For `ploss`="ml", regularized maximum likelihood estimation is used (Belloni et al. 2014; Farrell 2015). In this case, standard errors are only shown to be valid if both the propensity score model and the outcome regression model are correctly specified.

For this example, both the propensity score and outcome regressions models are (slightly) misspecified, by the design of the simulated data; see `help(simu.data)`.

Nevertheless, regularized calibrated estimation yields an estimate 0.12 with standard error 0.068, whereas regularized maximum likelihood estimation yields an estimate 0.094 with standard error 0.071. Both estimates are much closer to the true value 0, with smaller standard errors, than the unadjusted estimate.

```
> ## regularized calibrated estimation
> RNGversion('3.5.0')
> set.seed(0)   #this affects random split of data in cross validation
> mn.cv.rcal <-
+ mn.regu.cv(fold=5*c(1,1), nrho=(1+10)*c(1,1), rho.seq=NULL, y, tr, x,
+             ploss="cal", yloss="gaus")
> unlist(mn.cv.rcal$est)

       one         ipw          or         est         var          ze
 1.00025707  0.19199549 -0.05104572  0.11666356  0.00465415  1.71007445

> sqrt(mn.cv.rcal$est $var)

[1] 0.06822133

> mn.cv.rcal$ps$sel.nz[1]

[1] 9

> fp.cv.rcal <- mn.cv.rcal$ps$sel.fit[,1]
> ## regularized maximum likelihood estimation
> set.seed(0)   #this affects random split of data in cross validation
> mn.cv.rml <-
+ mn.regu.cv(fold=5*c(1,1), nrho=(1+10)*c(1,1), rho.seq=NULL, y, tr, x,
+             ploss="ml", yloss="gaus")
> unlist(mn.cv.rml$est)

       one        ipw         or        est        var         ze
0.97783503 0.15140034 0.05754193 0.09375530 0.00497098 1.32976473
```

```
> sqrt(mn.cv.rml$est $var)

[1] 0.07050518

> mn.cv.rml$ps$sel.nz[1]

[1] 24

> fp.cv.rml <- mn.cv.rml$ps$sel.fit[,1]
```

The following codes show how the same results can be obtained as above, but using the lower-level function `glm.regu.cv` to perform regularized M-estimation for fitting propensity score and outcome regression models, and using the function `mn.aipw` to compute the augmented IPW estimates.

```
> ## regularized calibrated estimation
> set.seed(0)
> ps.cv.rcal <-
+ glm.regu.cv(fold=5, nrho=1+10, y=tr, x=x, loss="cal")
> ps.cv.rcal$sel.nz[1]
> fp.cv.rcal <- ps.cv.rcal $sel.fit[,1]
> or.cv.rcal <-
+ glm.regu.cv(fold=5, nrho=1+10, y=y[tr==1], x=x[tr==1,],
+              iw=1/fp.cv.rcal[tr==1]-1, loss="gaus")
> fo.cv.rcal <- c( cbind(1,x)%*%or.cv.rcal$sel.bet[,1] )
> mn.cv.rcal2 <- unlist(mn.aipw(y, tr, fp=fp.cv.rcal, fo=fo.cv.rcal))
> mn.cv.rcal2
> ## regularized maximum likelihood estimation
> set.seed(0)
> ps.cv.rml <-
+ glm.regu.cv(fold=5, nrho=1+10, y=tr, x=x, loss="ml")
> ps.cv.rml$sel.nz[1]
> fp.cv.rml <- ps.cv.rml $sel.fit[,1]
```

```
> or.cv.rml <-
+ glm.regu.cv(fold=5, nrho=1+10, y=y[tr==1], x=x[tr==1,],
+              iw=NULL, loss="gaus")
> fo.cv.rml <- c( cbind(1,x)%*%or.cv.rml$sel.bet[,1] )
> mn.cv.rml2 <- unlist(mn.aipw(y, tr, fp=fp.cv.rml, fo=fo.cv.rml))
> mn.cv.rml2
```

2.2 Closer look at propensity score estimation

One of the difficulties in estimating the population mean μ^1 is that the treated group is, by definition, a selected sub-sample and hence may not be representative of the entire sample. The idea of inverse probability weighting is to reweight individuals in the treated group by the inverse of propensity scores, so that the weighted averages of covariates in the treated group are similar to the simple averages in the entire sample. Hence it is desirable to reduce the following differences as much as possible given the sample size and the number of covariates under study:

$$\frac{1}{n}\sum_{i=1}^{n}\left\{\frac{T_i}{\hat{\pi}^1(X_i)}-1\right\}X_{ji} = \sum_{i:\,T_i=1, 1\le i\le n}\hat{w}_i X_{ji} - \frac{1}{n}\sum_{i=1}^{n}X_{ji}, \quad j=1,\ldots,p,$$

where $\hat{w}_i = \{n\hat{\pi}^1(X_i)\}^{-1}$ and X_{ji} denotes the jth component of X_i. If the covariates are standardized with sample means 0 and variances 1, then the above gives the standardized calibration differences as in Tan (2020a), Section 6.

The following shows the calculation of such calibration differences using the function `mn.ipw`. The results are plotted in Figure 3.

```
> fp.raw <- rep(mean(tr), n)   #constant propensity scores
> check.raw <- mn.ipw(x, tr, fp.raw)
> check.cv.rcal <- mn.ipw(x, tr, fp.cv.rcal)
> check.cv.rml <- mn.ipw(x, tr, fp.cv.rml)
> par(mfrow=c(2,2))
> par(mar=c(4,4,2,2))
> plot(check.raw$est, xlim=c(0,p), ylim=c(-.3,.3),
```

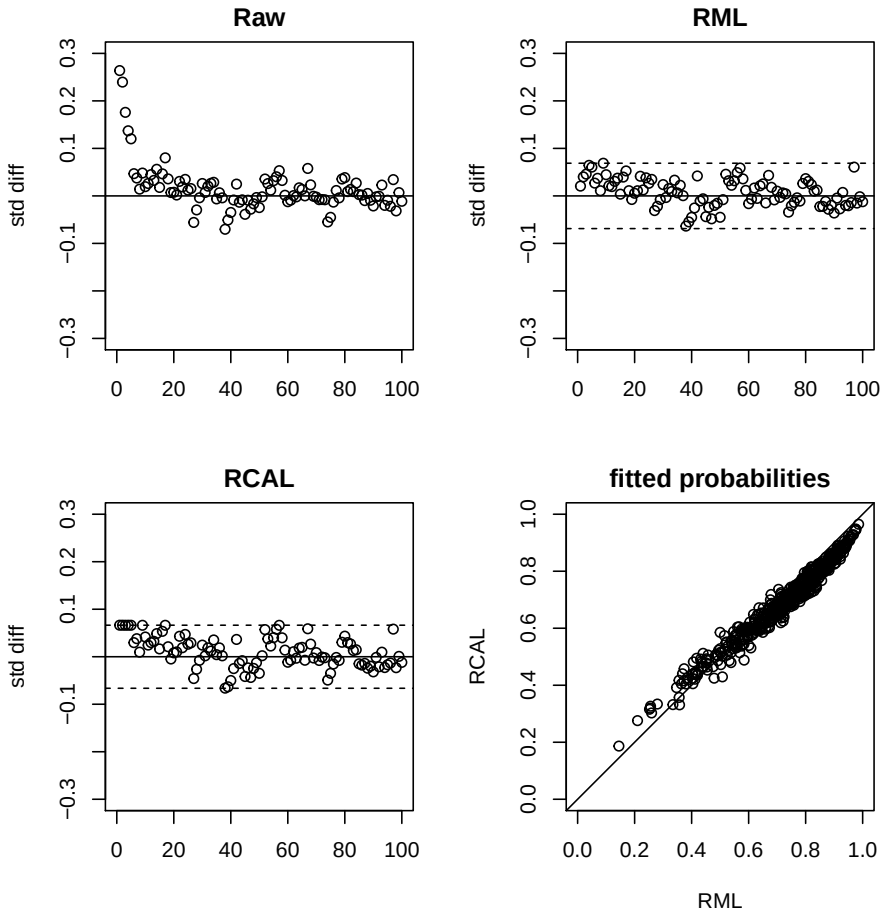
```
+      xlab="", ylab="std diff", main="Raw")
> abline(h=0)
> plot(check.cv.rml$est, xlim=c(0,p), ylim=c(-.3,.3),
+      xlab="", ylab="std diff", main="RML")
> abline(h=0)
> abline(h=max(abs(check.cv.rml$est)) *c(-1,1), lty=2)
> plot(check.cv.rcal$est, xlim=c(0,p), ylim=c(-.3,.3),
+      xlab="", ylab="std diff", main="RCAL")
> abline(h=0)
> abline(h=max(abs(check.cv.rcal$est)) *c(-1,1), lty=2)
> plot(fp.cv.rml[tr==1], fp.cv.rcal[tr==1], xlim=c(0,1), ylim=c(0,1),
+      xlab="RML", ylab="RCAL", main="fitted probabilities")
> abline(c(0,1))
```

The maximum standardized calibration differences from the two methods appear similar to each other. However, the number of nonzero coefficients estimated out of a total of 100 is 9 for regularized calibrated estimation, but much larger, 24, for regularized maximum likelihood estimation.

For further comparison, the following uses the function `glm.regu.path` to compute fitted propensity scores over regularization paths. Figure 4 shows how the maximum absolute standardized differences vary against the numbers of nonzero coefficients and relative variances, similarly as in Tan (2020a), Section 6. In this example, it seems impossible for regularized maximum likelihood estimation to reduce calibration differences to lower than 0.05, even with decreased Lasso penalties and increased numbers of nonzero coefficients and relative variance.

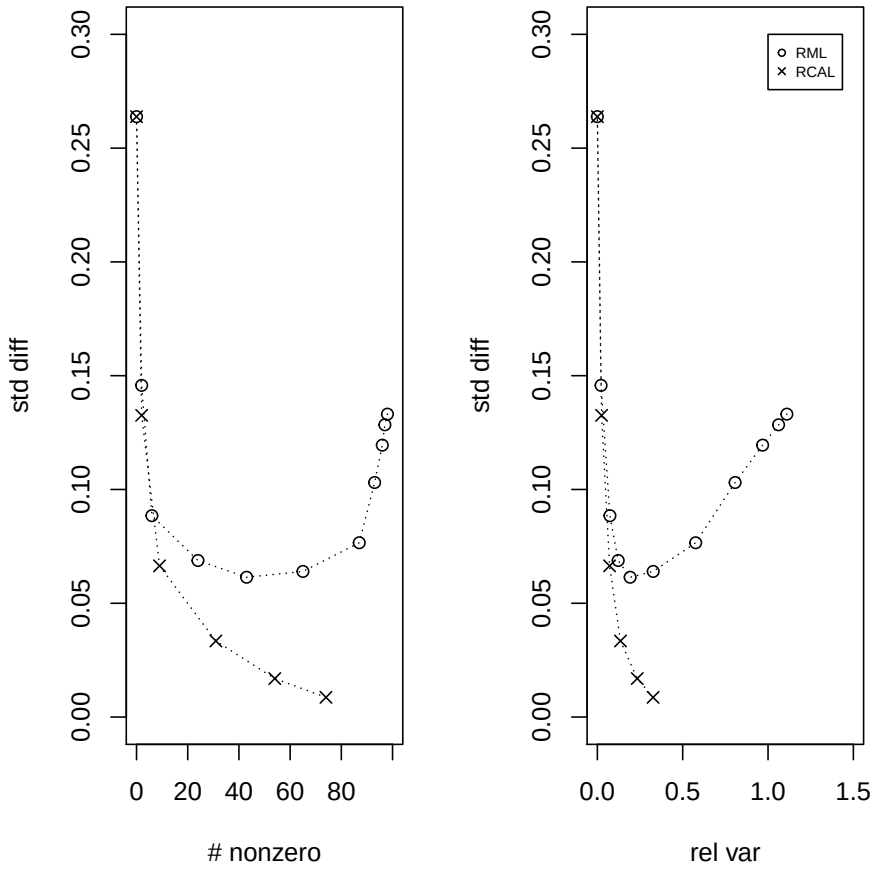
```
> set.seed(0)
> ps.path.rcal <-
+ glm.regu.path(nrho=1+10, rho.seq=NULL, y=tr, x=x, loss="cal")
> fp.path.rcal <- ps.path.rcal $fit.all[, !ps.path.rcal$non.conv]
> mdiff.path.rcal <- rep(NA, dim(fp.path.rcal)[2])
```

Figure 3: Standardized calibration differences and scatterplot of propensity scores.




```
> rvar.path.rcal <- rep(NA, dim(fp.path.rcal)[2])
> for (j in 1:dim(fp.path.rcal)[2]) {
+   check.path.rcal <- mn.ipw(x, tr, fp.path.rcal[,j])
+   mdiff.path.rcal[j] <- max(abs(check.path.rcal$est))
+   rvar.path.rcal[j] <-
+   var(1/fp.path.rcal[tr==1,j])/mean(1/fp.path.rcal[tr==1,j])^2
+ }
> set.seed(0)
> ps.path.rml <-
+ glm.regu.path(nrho=1+10, rho.seq=NULL, y=tr, x=x, loss="ml")
> fp.path.rml <- ps.path.rml $fit.all[, !ps.path.rml$non.conv]
```

Figure 4: Maximum absolute standardized differences against the numbers of nonzero coefficients and relative variances




```
> mdiff.path.rml <- rep(NA, dim(fp.path.rml)[2])
> rvar.path.rml <- rep(NA, dim(fp.path.rml)[2])
> for (j in 1:dim(fp.path.rml)[2]) {
+   check.path.rml <- mn.ipw(x, tr, fp.path.rml[,j])
+   mdiff.path.rml[j] <- max(abs(check.path.rml$est))
+   rvar.path.rml[j] <-
+   var(1/fp.path.rml[tr==1,j])/mean(1/fp.path.rml[tr==1,j])^2
+ }
```

2.3 Estimation of average treatment effects

The following codes show the use of the function `ate.regu.cv`, to estimate the two means (μ^0, μ^1) and the ATE, $\mu^1 - \mu^0$.

```
> ## regularized calibrated estimation
> set.seed(0)
> ate.cv.rcal <-
+ ate.regu.cv(fold=5*c(1,1), nrho=(1+10)*c(1,1), rho.seq=NULL, y, tr, x,
+             ploss="cal", yloss="gaus")
> matrix(unlist(ate.cv.rcal$est), ncol=2, byrow=T,
+ dimnames=list(c("one", "ipw", "or", "est", "var", "ze",
+ "diff.est", "diff.var", "diff.ze"), c("untreated", "treated")))
```

```
            untreated      treated
one       0.999918656  1.000257074
ipw      -0.358056858  0.191995489
or       -0.186425979 -0.051045723
est      -0.210128782  0.116663561
var       0.008127681  0.004654150
ze       -2.330784967  1.710074450
diff.est           NA  0.326792342
diff.var           NA  0.008974677
diff.ze            NA  3.449550099
```

```
> ## regularized maximum likelihood estimation
> set.seed(0)
> ate.cv.rml <-
+ ate.regu.cv(fold=5*c(1,1), nrho=(1+10)*c(1,1), rho.seq=NULL, y, tr, x,
+             ploss="ml", yloss="gaus")
> matrix(unlist(ate.cv.rml$est), ncol=2, byrow=T,
+ dimnames=list(c("one", "ipw", "or", "est", "var", "ze",
+ "diff.est", "diff.var", "diff.ze"), c("untreated", "treated")))
```

```
             untreated     treated
one       0.867832577 0.977835026
ipw      -0.459412254 0.151400336
or       -0.458609601 0.057541927
est      -0.314830195 0.093755295
var       0.007477782 0.004970980
ze       -3.640742676 1.329764733
diff.est           NA 0.408585490
diff.var           NA 0.009268557
diff.ze            NA 4.244014742
```

REFERENCES

Angrist, J.D., Imbens, G.W. and Rubin, D.B. (1996) Identification of causal effects using instrumental variables, *Journal of the American Statistical Association*, 91, 444–455.

Belloni, A., Chernozhukov, V., and Hansen, C. (2014) Inference on treatment effects after selection among high-dimensional controls, *Review of Economic Studies*, 81, 608-650.

Farrell, M.H. (2015) Robust inference on average treatment effects with possibly more covariates than observations. *Journal of Econometrics*, 189, 1-23.

Neyman, J. (1923) On the application of probability theory to agricultural experiments: Essay on principles, Section 9, translated in *Statistical Science*, 1990, 5, 465-480.

Robins, J.M., Rotnitzky, A., and Zhao, L.P. (1994) Estimation of regression coefficients when some regressors are not always observed, *Journal of the American Statistical Association*, 89, 846-866.

Rosenbaum, P.R. and Rubin, D.B. (1983) The central role of the propensity score in observational studies for causal effects, *Biometrika*, 70, 41-55.

Rubin, D.B. (1974) Estimating causal effects of treatments in randomized and non-randomized studies, *Journal of Educational Psychology*, 66, 688-701.

Tan, Z. (2020a) Regularized calibrated estimation of propensity scores with model misspecification and high-dimensional data, *Biometrika*, 107, 137–158.

Tan, Z. (2020b) Model-assisted inference for treatment effects using regularized calibrated estimation with high-dimensional data, *Annals of Statistics*, 48, 811–837.