Problem Set 8: Regression Discontinuity Designs

GOV 2003

Due at 11:59 pm (ET) on Nov 17, 2021

Instruction

Before you begin, please read the following instructions **carefully**:

- **No late submission is allowed** without prior approval from the instructors.
- **All answers should be typed up**. We recommend the use of `Rmarkdown`. A `Rmarkdown` template for this problem set is provided. Answers to analytical solutions should also be typed up.
- **A PDF copy of your answer** including your computer code should be uploaded to `Gradescope` before the deadline. **Do not submit the markdown file itself**.
- This problem set includes a bonus question for extra credit. No deduction in the total points will be made from this question. Note that the maximum points of this problem set is 15 points. That is, if the student receives 3 points from the bonus question and 14 points from the other questions, the total points will be 15 points.

Introduction

In this problem set, we will replicate and extend the analysis of Hall (2015). The article examines how the nomination of an extremist changes general-election outcomes and legislative behavior in the U.S. House, 1980-2010, using a **sharp regression discontinuity design** in primary elections. Here, the forcing variable (X_{ipt}) is the extreme candidate's vote-share winning margin in the primary election[1], the treatment is an indicator variable for the extremist winning party p's primary in district i at time t, and the outcome variables is party vote share in general election. Please skim the "Empirical Approach" section (p. 20 – 23) for more details.

We will compare two different identification strategies, one assuming the *continuity assumption* and the other assuming the *local randomization.* Then, we will further investigate the plausibility of continuity assumption in this context using pretreatment covariates and McCrary test. Finally, we will check the robustness of the result using different choices of bandwidths and conduct an estimation under optimal bandwidth using the method developed by Calonico et al. (2014). Use the data frame `data` from `Hall2015.RData` throughout the questions.

[1] "In primary races with two major candidates, the race is tentatively identified as being between an extremist and a relatively moderate candidate if the difference between their estimated ideological positions is at or above the median in the distribution of ideological distances between the top two candidates in all contested primary elections (21–22)".

Variable	Description
Forcing variable	
`rv`	The extreme candidate's vote-share winning margin in the primary election (X_{ipt})
Treatment variable	
`treat`	An indicator variable for the extremist winning party p's primary in district i at time t (D_{ipt})
Outcome variable	
`dv`	Party vote share in general election (Y_{ipt})
Pre-treatment covariates	
`pres_normal_vote`	Presidential vote share
`prim_share`	Extremist share of primary Money
`prim_pac_share`	Extremist share of PAC primary money
`prim_total0`	Extremist total primary noney (100k)
`abs_dw_lag`	Lagged DW-NOMINATE
`abs_lag_wnom`	Lagged W-NOMINATE
`dv_lag`	Lagged vote share

Question 1: Identification and estimation (7 pts; 1 pt for (a) and (e); 2 pts for (b), (c), (d))

(a) We will first replicate the binned means plot (Figure 2 without regression lines) from the original paper. Subset the data as specified in the paper, and draw the binned means (large black dots) with the raw data points (gray dots). Note that each size of the bins is 0.02 and make sure to include all the data points in your figure (as opposed to the original paper where it constraints the y-axis to be [.35, .8]). Briefly explain the plot in words.

Hints:

- Subset the data to only include races between an extremist and a relatively moderate candidate within 0.2 margin.

```
subset(data, data$margin <= .2 & data$absdist > median(data$absdist))
```

- It may be useful to make a function for plotting the binned means plot so that we can reuse it in the following questions.

(b) Suppose that we assume *continuity assumption* from the lecture for the identification. Formally state the assumption using the mathematical expressions. Estimate the local average treatment effect at the cutoff (τ_{SRD}) under this assumption using the local linear regression given the bandwidth $h = 0.2$. Visualize the result along with the binned means from (a) and briefly explain the result in words.

Hint: Note that the visualization of the local linear regression would be exactly the same as Figure 2.

(c) Alternatively, suppose that now we assume *local randomization* (also called the as-if-random assumption) for the identification where we set local to be $-0.2 \leq X_i \leq 0.2$. Formally state the assumption using the mathematical expressions. Estimate the local average treatment effect at the cutoff under this assumption using the difference-in-means estimator. Visualize the result along with the binned means from (a) and briefly explain the result in words.

(d) Compare the two different identifications and its estimation from (b) and (c). Which identification strategy would you prefer and why?

Hint: (Optional) You may find the discussion from de la Cuesta and Imai (2016) useful.

(e) [Bonus] Under the continuity assumption, now fit local regression with cubic polynomials. Visualize the result as before and briefly discuss the result. What would be problematic with this estimation?

Answer 1

(a)

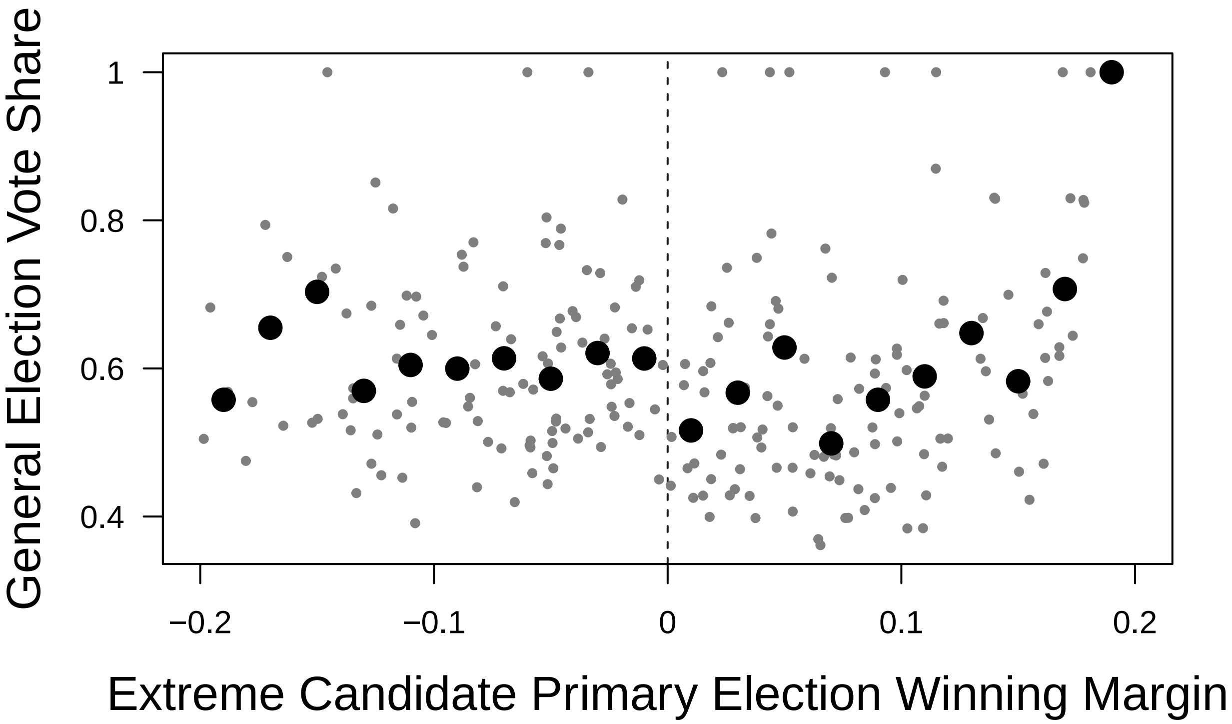
```
## Codes from the replication materials
load("Hall2015.RData")
# subset graph to only elections within .2 margin and above median distance
rd.data <- subset(data, data$margin <= .2 & data$absdist > .1095)

binnedmeans <- function(dv, y.title = "General Election Vote Share",
                        bin.size = .02) {
  # make the binned averages, first the bins to the left
  count <- 1
  # bin.size <- .02
  binx.left <- vector(length=length(seq(-.2, 0, bin.size)))
  biny.left <- vector(length=length(binx.left))
  last <- -.2
  for(j in seq(-.2, 0, bin.size)) {
    biny.left[count] <- mean(dv[rd.data$rv >= j-bin.size & rd.data$rv < j], na.rm=T)
    binx.left[count] <- (j+last)/2
    last <- j
    count <- count + 1
  }

  # now the bins on the right
  count <- 1
  # bin.size <- .02
  binx.right <- vector(length=length(seq(0.02,.2, bin.size)))
  biny.right <- vector(length=length(binx.right))
  last <- 0
  for(j in seq(0.02,.2, bin.size)) {
    biny.right[count] <- mean(dv[rd.data$rv >= j-bin.size & rd.data$rv < j], na.rm=T)
    binx.right[count] <- (j+last)/2
    last <- j
    count <- count + 1
  }
```

```
  plot(x=rd.data$rv, y=dv, col="white", xlab="Extreme Candidate Primary Election Winning Margi
  abline(v=0, col="gray10", lty=2)
  points(x=rd.data$rv, y=dv, pch=16, col="gray50", cex=.7)
  points(x=binx.left, y=biny.left, cex=1.7, col="black", pch=16)
  points(x=binx.right, y=biny.right, cex=1.7, pch=16)
  axis(side=2, las=1, cex.axis=.9, at=seq(0, 1, .2), labels=seq(0, 1, 0.2), cex.axis=1)
  axis(side=1, at=seq(-0.2, 0.2, 0.1), labels=seq(-0.2, 0.2, 0.1), cex.axis=1)
  # text(x=0.18,y=0.37, paste("N=", nrow(rd.data), sep=""), cex=1.4)
}

binnedmeans(dv = rd.data$dv)
```



(b)

CEFs of potential outcomes are **continuous** in X_{ipt}

- $\mu_1(x) = \mathbb{E}[Y_{ipt}(1) \mid X_{ipt} = x]$ is continuous
- $\mu_0(x) = \mathbb{E}[Y_{ipt}(0) \mid X_{ipt} = x]$ is continuous

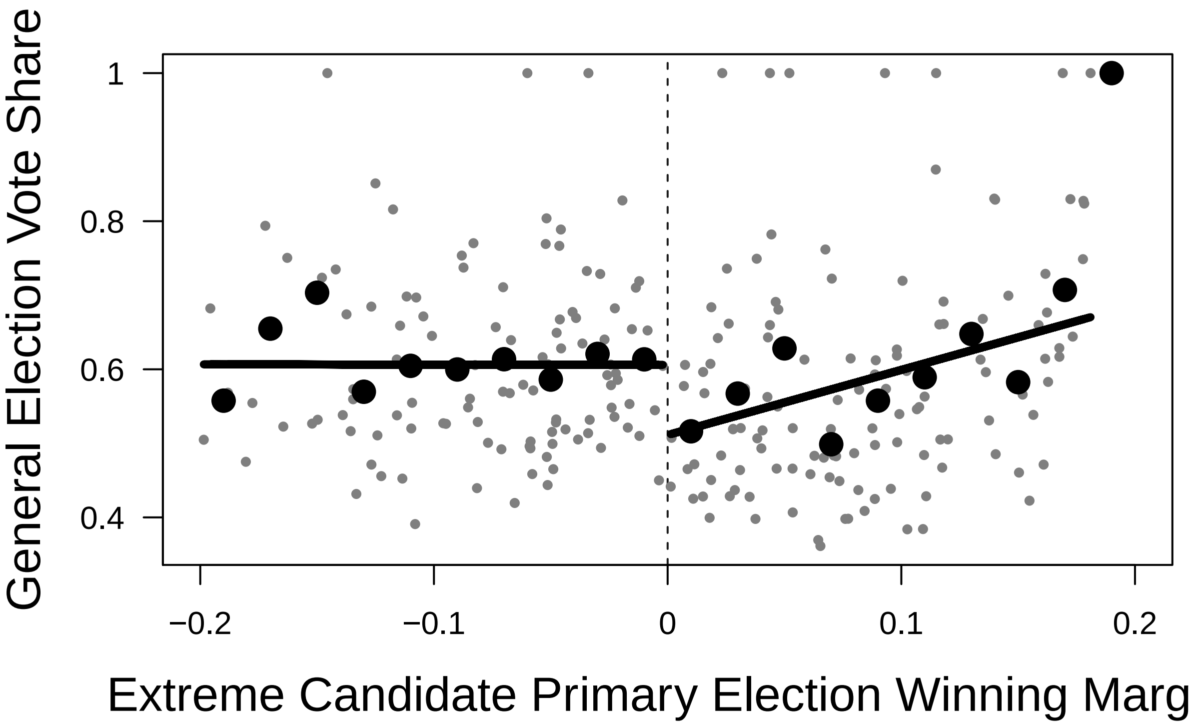
```
reg1 <- lm(dv ~ rv, data = rd.data, subset = rv < 0)
reg2 <- lm(dv ~ rv, data = rd.data, subset = rv > 0)

tau_srd <- reg2$coefficients[1] - reg1$coefficients[1]
tau_srd
```

```
## (Intercept)
## -0.09472337
```

```
fits1 <- reg1$coefficients[1] + reg1$coefficients[2] * rd.data$rv[rd.data$rv<0]
fits2 <- reg2$coefficients[1] + reg2$coefficients[2] * rd.data$rv[rd.data$rv>0]

binnedmeans(dv = rd.data$dv)
lines(x=rd.data$rv[rd.data$rv<0], y=fits1, lwd=4, col="black")
lines(x=rd.data$rv[rd.data$rv>0], y=fits2, lwd=4, col="black")
```



(c)

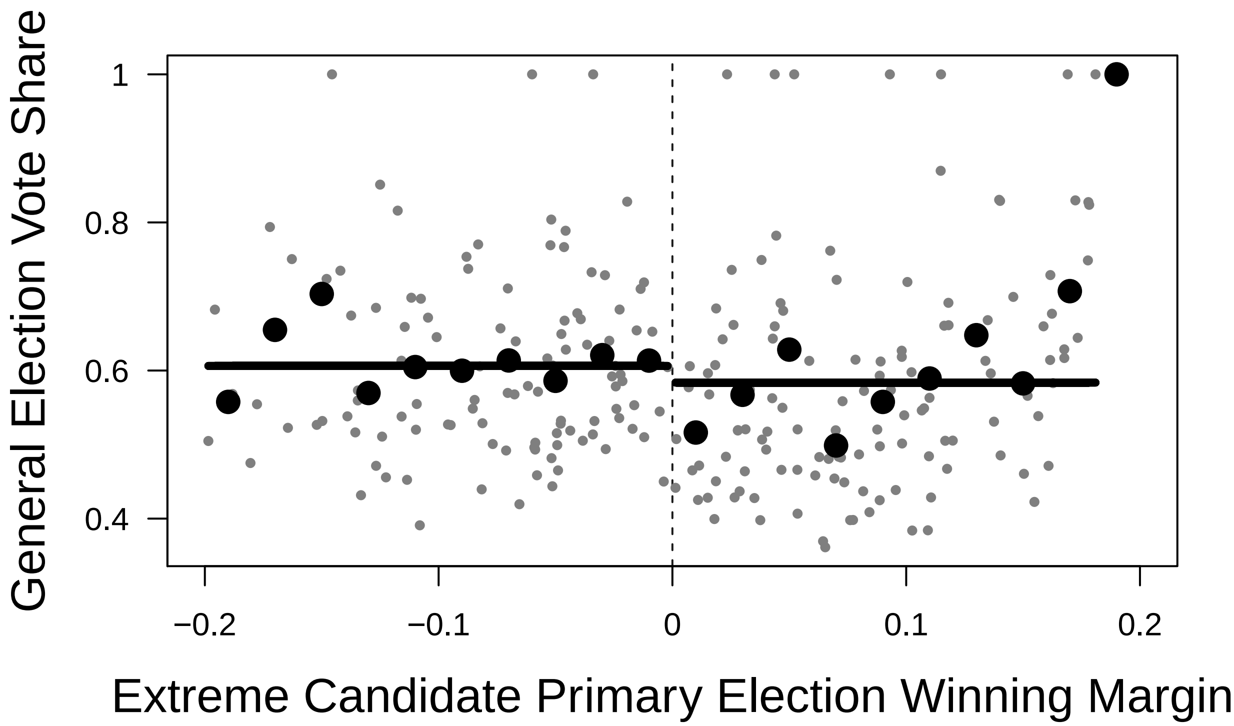
$$\{Y_{ipt}(1), Y_{ipt}(0)\} \perp\!\!\!\perp \mathbb{1}\{X_{ipt} > c\} \mid c_0 \leq X_{ipt} \leq c_1$$

where $c = 0, c_0 = -0.2$, and $c_1 = 0.2$.

```
tau_srd_rand <- mean(rd.data$dv[rd.data$rv>0]) - mean(rd.data$dv[rd.data$rv<0])
tau_srd_rand
```

```
## [1] -0.02257234
```

```
binnedmeans(dv = rd.data$dv)
lines(x=rd.data$rv[rd.data$rv<0],
      y=rep(mean(rd.data$dv[rd.data$rv<0]), length(rd.data$rv[rd.data$rv<0])), lwd=4, col="blac
lines(x=rd.data$rv[rd.data$rv>0],
      y=rep(mean(rd.data$dv[rd.data$rv>0]), length(rd.data$rv[rd.data$rv>0])), lwd=4, col="blac
```



(d) Local randomization is stronger than continuity because it rules out the direct effect of extreme candidate's vote-share on the outcome (i.e., confounding around the cutoff 0) and thus implies no slope in the CEFs around 0.

(e)

The estimation result using cubic polynomial may be sensitive to the extreme points (e.g., the points near 0.2).

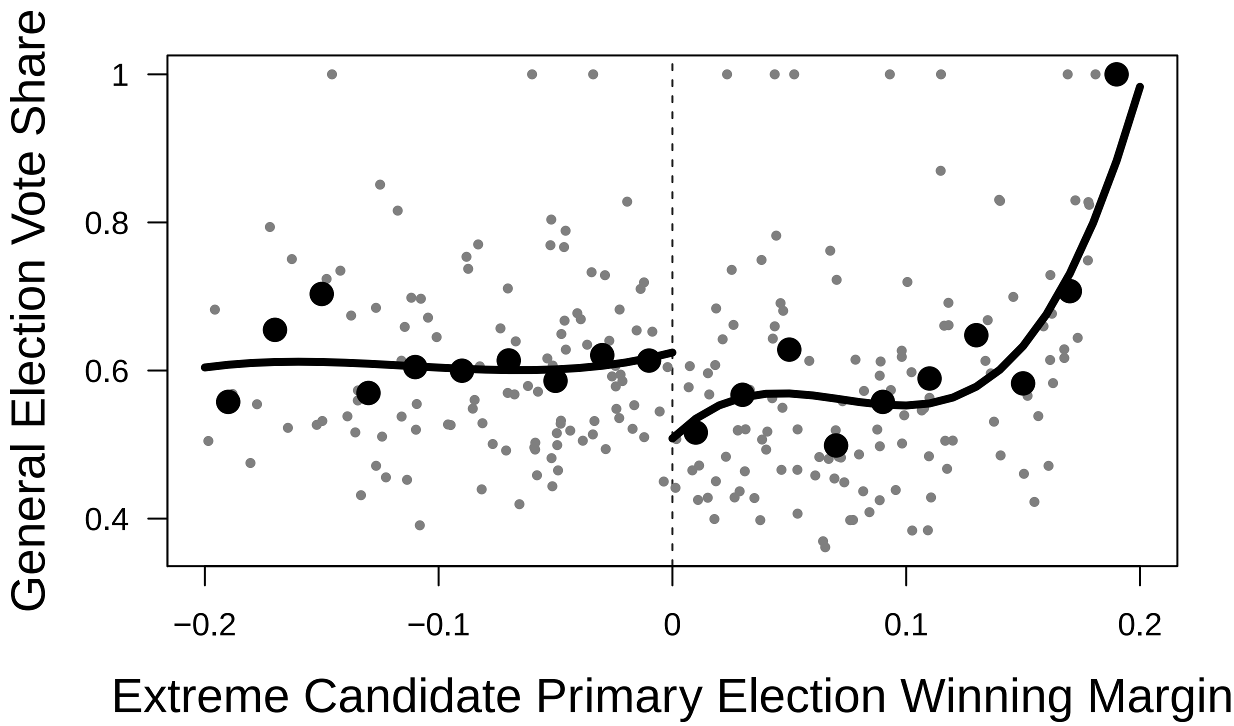
```
reg1_cubic <- lm(dv ~ rv + I(rv^2) + I(rv^3), data = rd.data, subset = rv < 0)
reg2_cubic <- lm(dv ~ rv + I(rv^2) + I(rv^3), data = rd.data, subset = rv > 0)

tau_srd <- reg2_cubic$coefficients[1] - reg1_cubic$coefficients[1]
tau_srd
```

```
## (Intercept)
##  -0.1160054
```

```
lo.seq <- seq(-0.2, 0, by = 0.01)
hi.seq <- seq(0, 0.2, by = 0.01)
fits1_cubic <- predict(reg1_cubic, data.frame(rv = lo.seq))
fits2_cubic <- predict(reg2_cubic, data.frame(rv = hi.seq))

binnedmeans(dv = rd.data$dv)
lines(x=lo.seq, y=fits1_cubic, lwd=4, col="black")
lines(x=hi.seq, y=fits2_cubic, lwd=4, col="black")
```



Question 2: Diagnostics (4 pts; 2 pts for each)

(a) In this question, we will investigate the mean of pre-treatment covariates to check the plausibility of continuity assumption. Replicate the binned means plot of pre-treatment covariates in Figure A.2 using the same bandwidth as before. What does this result tell us about the continuity assumption?

(b) Now, we will conduct McCrary test to check the plausibility of continuity assumption. Visualize the McCrary test as in the lecture slides. What does this result tell us about the continuity assumption?

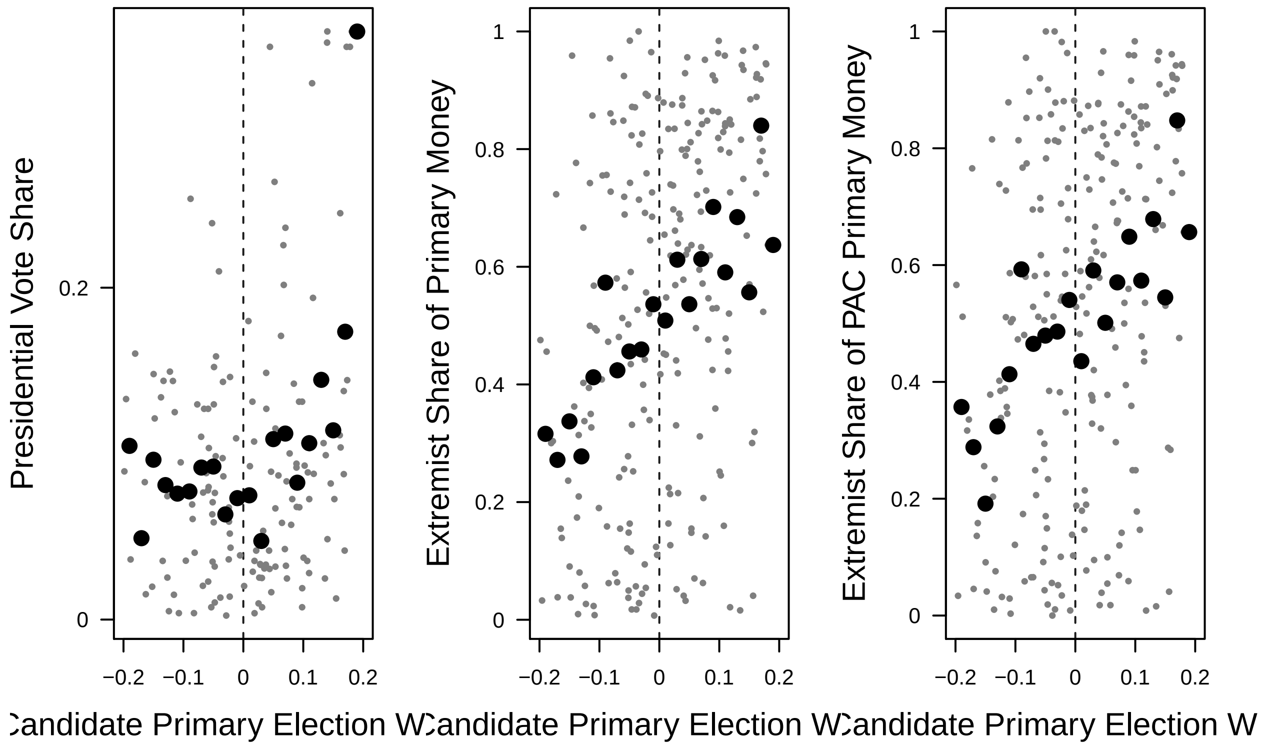
Hint: You may use `rdd::DCdensity()` function.

Answer 2

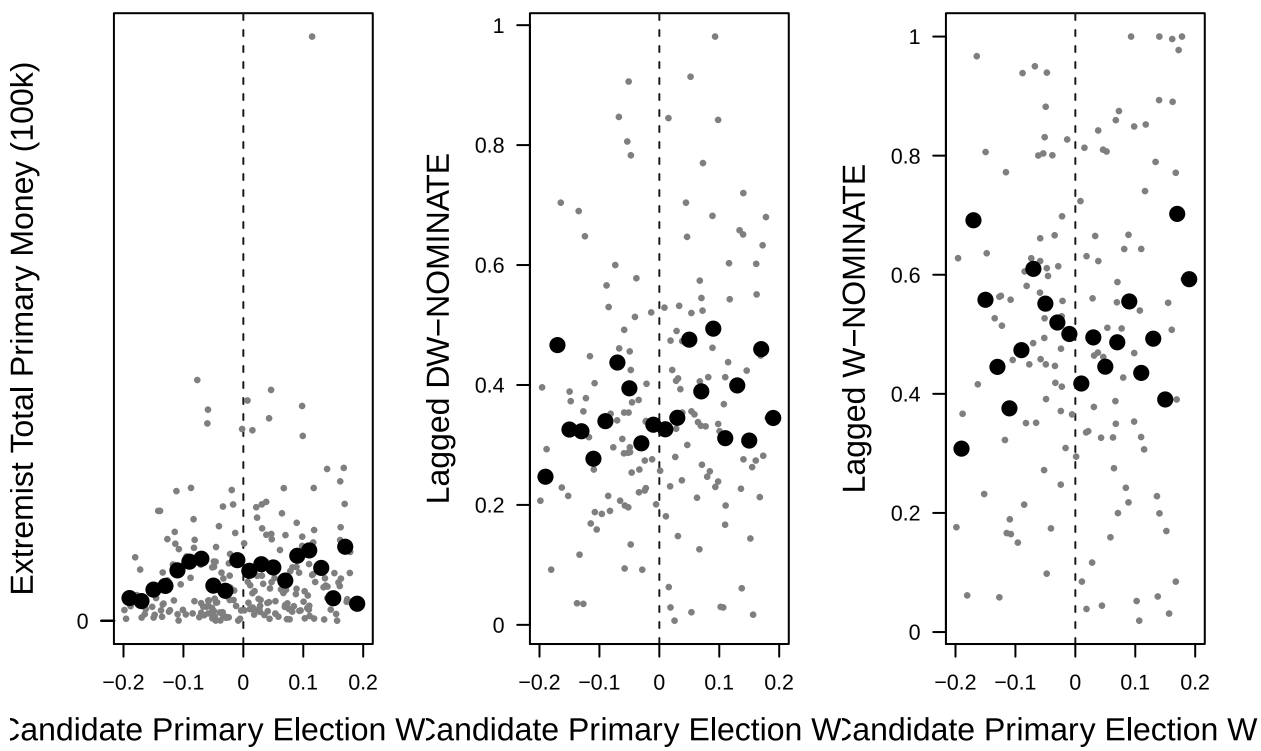
(a)

It is difficult to discern any evidence of sorting.

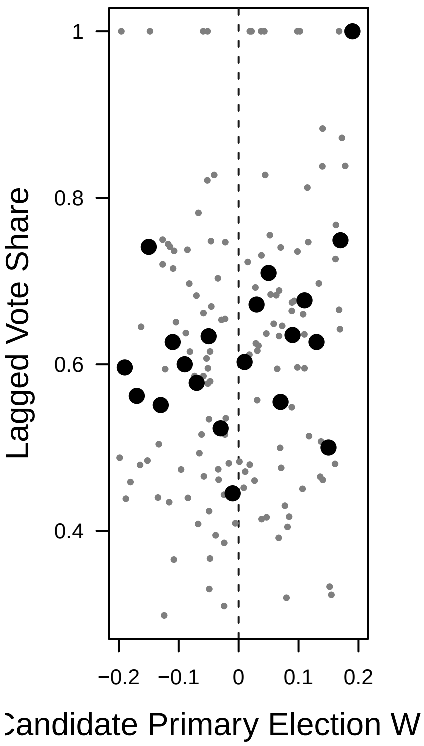
```
par(mfrow=c(1,3))
binnedmeans(dv = rd.data$pres_normal_vote, y.title = "Presidential Vote Share")
binnedmeans(dv = rd.data$prim_share, y.title = "Extremist Share of Primary Money")
binnedmeans(dv = rd.data$prim_pac_share, y.title = "Extremist Share of PAC Primary Money")
```




```r
binnedmeans(dv = rd.data$prim_total0, y.title = "Extremist Total Primary Money (100k)")
binnedmeans(dv = rd.data$abs_dw_lag, y.title = "Lagged DW-NOMINATE")
binnedmeans(dv = rd.data$abs_lag_wnom, y.title = "Lagged W-NOMINATE")
```



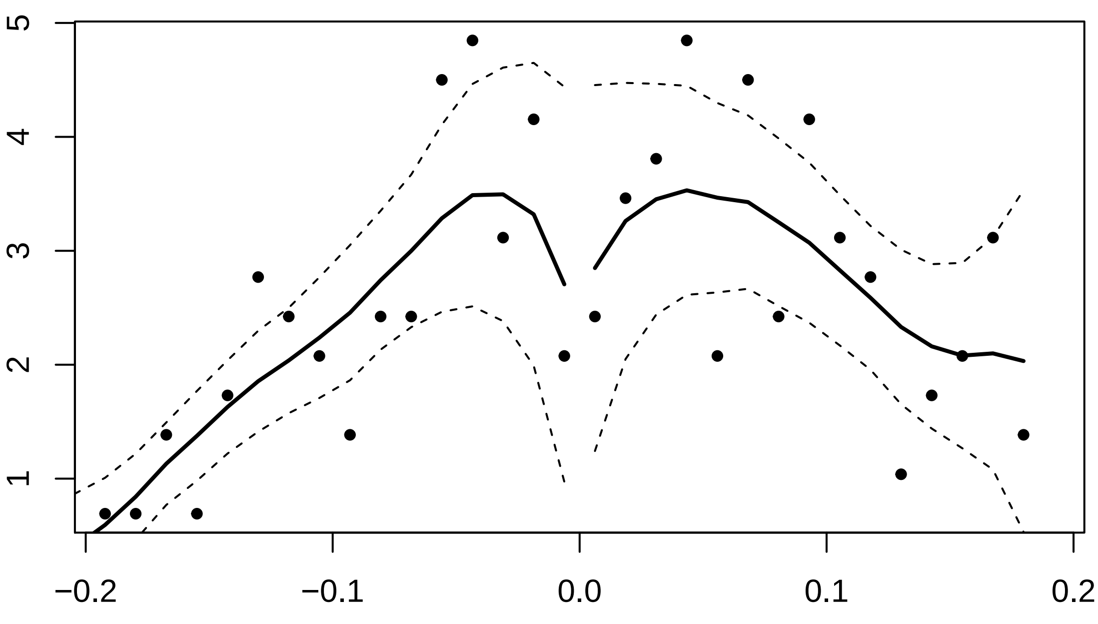
```
binnedmeans(dv = rd.data$dv_lag, y.title = "Lagged Vote Share")
```



(b)

It is difficult to discern an evidence of sorting.

```
rdd::DCdensity(rd.data$rv, cutpoint = 0)
```




```
## [1] 0.7654695
```

Question 3: Bandwidth selection and bias correction (4 pts; 1 pt for (a) and (b); 2 pts for (c))

(a) Now, we correct the bias of the point estimate obtained in Q1 (b). Under the continuity assumption, estimate the local average treatment effect using a bias-corrected estimator with triangular kernel weights and bandwidth of $h = 0.2$, and estimate the standard error that accounts for the uncertainty of bias estimation as in Calonico et al. (2014). Briefly explain the result.

Hint: You may use `rdrobust::rdrobust()` function.

(b) Estimate the optimal bandwidth using the same estimator from (a). Also, estimate the bias-corrected point estimate and its robust standard error using this optimal bandwidth. Briefly explain the result.

(c) Additionally, we will examine the sensitivity of the conclusion obtained in (a) in terms of different choices of bandwidth. For each of $h \in \{0.05, 0.1, \ldots, 0.45, 0.5\}$, repeat the estimation as in (a). Visualize the result where x-axis is the bandwidth and y-axis is the estimate with 95% confidence interval. Briefly explain the results.

Answer 3

(a)

Observe that the 95% confidence interval of bias-correct estimate includes 0.

```r
library(rdrobust)
res = rdrobust(y = rd.data$dv, x = rd.data$rv, h = 0.2, p = 1, kernel = "tri")
summary(res)
```

```
## Call: rdrobust
##
## Number of Obs.                  233
## BW type                      Manual
## Kernel                   Triangular
## VCE method                       NN
##
## Number of Obs.                  109         124
## Eff. Number of Obs.             109         124
## Order est. (p)                    1           1
## Order bias  (q)                   2           2
## BW est. (h)                   0.200       0.200
## BW bias (b)                   0.200       0.200
## rho (h/b)                     1.000       1.000
## Unique Obs.                     109         124
##
## =============================================================================
##         Method     Coef. Std. Err.         z     P>|z|      [ 95% C.I. ]
## =============================================================================
##   Conventional    -0.075     0.031    -2.411     0.016    [-0.136 , -0.014]
```

```
##         Robust          -           -    -1.320     0.187    [-0.141 , 0.028]
## =============================================================================
```

```r
res[["Estimate"]]
```

```
##           tau.us      tau.bc      se.us      se.rb
## [1,] -0.07521651 -0.05687483 0.03119114 0.04309386
```

```r
## Additionally, result w/o weights:
res_unif = rdrobust(y = rd.data$dv, x = rd.data$rv, h = 0.2, p = 1, kernel = "uniform")
summary(res_unif)
```

```
## Call: rdrobust
##
## Number of Obs.                  233
## BW type                      Manual
## Kernel                      Uniform
## VCE method                       NN
##
## Number of Obs.                 109          124
## Eff. Number of Obs.            109          124
## Order est. (p)                   1            1
## Order bias  (q)                  2            2
## BW est. (h)                  0.200        0.200
## BW bias (b)                  0.200        0.200
## rho (h/b)                    1.000        1.000
## Unique Obs.                    109          124
##
## =============================================================================
##         Method     Coef. Std. Err.         z     P>|z|      [ 95% C.I. ]
## =============================================================================
##   Conventional    -0.095     0.031    -3.101     0.002    [-0.155 , -0.035]
##         Robust          -          -    -0.716     0.474    [-0.116 , 0.054]
## =============================================================================
```

```r
res_unif[["Estimate"]]
```

```
##           tau.us     tau.bc      se.us      se.rb
## [1,] -0.09472337 -0.0309449 0.03054742 0.04322363
```

(b)

```r
res_opt = rdrobust(y = rd.data$dv, x = rd.data$rv, bwselect = "mserd", p = 1, kernel = "tri")
summary(res_opt)
```

```
## Call: rdrobust
##
## Number of Obs.                  233
## BW type                       mserd
## Kernel                   Triangular
## VCE method                       NN
```

```
##
## Number of Obs.                 109          124
## Eff. Number of Obs.             26           28
## Order est. (p)                   1            1
## Order bias  (q)                  2            2
## BW est. (h)                  0.036        0.036
## BW bias (b)                  0.064        0.064
## rho (h/b)                    0.566        0.566
## Unique Obs.                    109          124
##
## =============================================================================
##         Method     Coef. Std. Err.         z     P>|z|      [ 95% C.I. ]
## =============================================================================
##   Conventional    -0.078     0.059    -1.307     0.191    [-0.194 , 0.039]
##         Robust         -         -    -0.945     0.345    [-0.207 , 0.072]
## =============================================================================
```

```r
res_opt[["Estimate"]]
```

```
##           tau.us      tau.bc      se.us      se.rb
## [1,] -0.07764892 -0.06730798 0.05939979 0.07123753
```

```r
res_opt[["bws"]]
```

```
##         left      right
## h 0.03635638 0.03635638
## b 0.06428765 0.06428765
```

```r
## Additionally, result w/o weights:
res_opt_unif = rdrobust(y = rd.data$dv, x = rd.data$rv, bwselect = "mserd", p = 1, kernel = "un
summary(res_opt_unif)
```

```
## Call: rdrobust
##
## Number of Obs.                   233
## BW type                        mserd
## Kernel                       Uniform
## VCE method                        NN
##
## Number of Obs.                 109          124
## Eff. Number of Obs.             28           31
## Order est. (p)                   1            1
## Order bias  (q)                  2            2
## BW est. (h)                  0.039        0.039
## BW bias (b)                  0.071        0.071
## rho (h/b)                    0.544        0.544
## Unique Obs.                    109          124
##
## =============================================================================
##         Method     Coef. Std. Err.         z     P>|z|      [ 95% C.I. ]
```

```
## ==========================================================================
##   Conventional    -0.055     0.063    -0.864     0.388    [-0.179 , 0.069]
##         Robust         -         -    -0.880     0.379    [-0.214 , 0.081]
## ==========================================================================
```

```
res_opt_unif[["Estimate"]]
```

```
##           tau.us      tau.bc      se.us      se.rb
## [1,] -0.05471249 -0.06637522 0.06333658 0.07539673
```

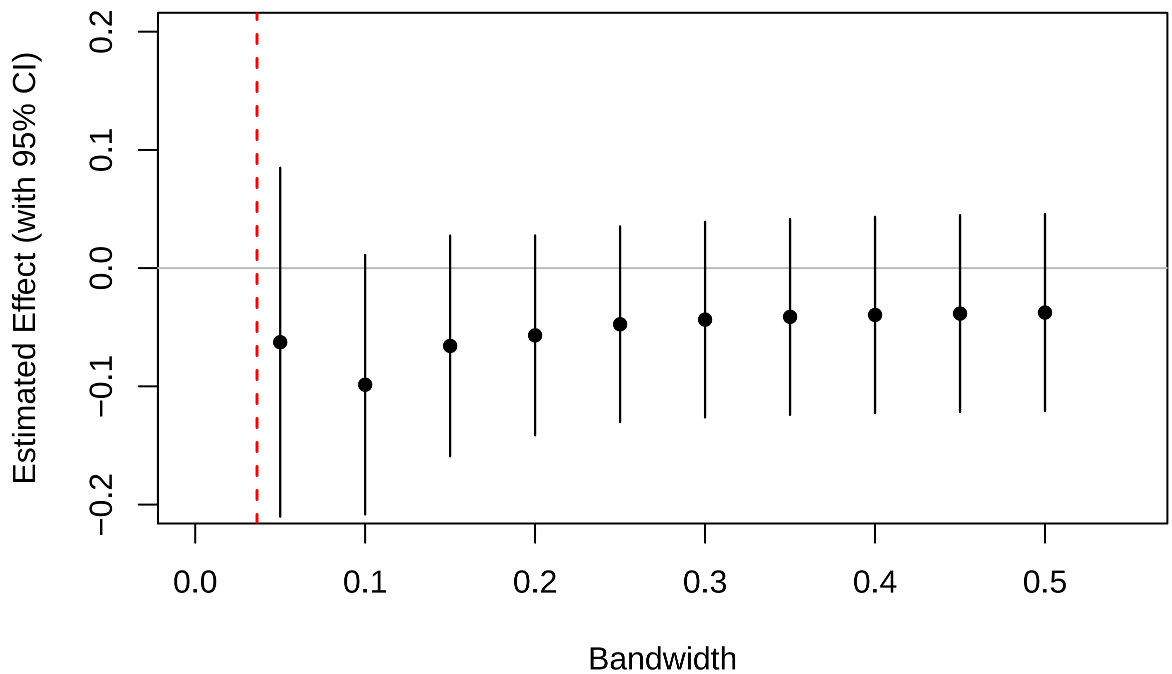
```
res_opt_unif[["bws"]]
```

```
##         left      right
## h 0.03862340 0.03862340
## b 0.07094042 0.07094042
```

(c)

```
h.cand = seq(0.05, 0.5, 0.05)
tau_srd.ls <- list()
for (i in 1:length(h.cand)) {
  tau_srd.ls[[i]] = rdrobust(y = rd.data$dv, x = rd.data$rv, h = h.cand[i], p = 1, kernel = "t
}

plot(1, 1, type = 'n', xlim = c(0, 0.55), ylim = c(-0.2, 0.2),
xlab = 'Bandwidth', ylab = 'Estimated Effect (with 95% CI)')
abline(v = res_opt$bws[1,1], col = 'red', lwd = 1.5, lty=2)
abline(h = 0, col="grey", lwd = 1)
for (b in 1:length(h.cand)) {
  points(h.cand[b], tau_srd.ls[[b]]$coef[3], pch = 16)
  lines(c(h.cand[b], h.cand[b]), tau_srd.ls[[b]]$ci[3,], lwd = 1.2)
}
```



References

Calonico, S., Cattaneo, M. D., and Titiunik, R. (2014). Robust nonparametric confidence intervals for regression-discontinuity designs. *Econometrica*, 82(6):2295–2326.

de la Cuesta, B. and Imai, K. (2016). Misunderstandings about the regression discontinuity design in the study of close elections. *Annual Review of Political Science*, 19(1):375–396.

Hall, A. B. (2015). What happens when extremists win primaries? *American Political Science Review*, 109(1):18–42.