Regression Discontinuity Designs and Difference-in-Differences

Jake Bowers, Ben Hansen & Tom Leavitt

August 7, 2016

Contents

```
if (!require("pacman")) {install.packages("pacman")}
```

```
## Loading required package:  pacman
```

```
pacman::p_load(
  plyr,
  dplyr,
  magrittr,
  tidyr,
  haven,
  broom,
  randomizr,
  lmtest,
  ggplot2,
  knitr,
  xtable,
  devtools,
  optmatch,
  RItools,
  sandwich,
  parallel,
  doParallel
)

opts_chunk$set(message = FALSE,
               warning = FALSE,
               size = "footnotesize")
```

1 Regression Discontinuity Design

1.1 General Setup

Today we are going to use the data from Caughey and Sekhon (2011), which is available for download here. Caughey and Sekhon (2011) engage in a debate whose participants seek to identify the causal effect of the so-called "incumbency advantage." That is, what effect does a candidate's status as an incumbent have on whether or not that candidate wins an election? Obviously, whether or not a candidate is an incumbent is *not* randomly assigned.

Let's first load the data:

```
rm(list=ls())

rdd_data <- read_dta("RDReplication.dta") %>%
  filter(Use == 1) ## Use is indicator for whether unit is included in RD incumbency advantage sample

rdd_data[] <- lapply(rdd_data[], unclass)
```

The Running Variable

The "running variable" is called DifDPct, which is defined as the Democratic margin of victory or defeat in the election; in other words, DifDPct is the difference between the percentage of all votes that were cast for the leading Democrat in the race and the percentage cast for the leading non-Democrat. Races in which no Democrat ran or in which the top two vote-getters were both Democrats are coded as missing.

```
running_var <- matrix(c('DifDPct', 'Democrat Margin of Victory'),
              ncol = 2,
              byrow = TRUE)

dimnames(running_var) <- list(1, c("Running Variable", "Description"))

kable(running_var)
```

Running Variable	Description
DifDPct	Democrat Margin of Victory

The Treatment Variable

The treatment variable is whether or not the Democratic candidate wins the election or not. If the candidate wins the election, then that candidate is assigned to "treatment." If the candidate loses the election, then he or she is assigned to "control."

```
treatment <- matrix(c('DemWin', 'Democrat Wins Election'),
              ncol = 2,
              byrow = TRUE)

dimnames(treatment) <- list(1, c("Treatment", "Description"))

kable(treatment)
```

Treatment	Description
DemWin	Democrat Wins Election

Now let's quickly look at the empirical distribution of the treatment variable:

```
table(rdd_data$DemWin)
```

	Democrat Wins Election
0	4507
1	5677

Outcome Variables

In Caughey and Sekhon (2011), the primary outcome variables of interest are as follows:

```r
dvs <- matrix(c('DWinNxt', 'Dem Win t + 1',
                'DPctNxt', 'Dem % t + 1',
                'DifDPNxt', 'Dem % Margin t + 1'),
              ncol = 2,
              byrow = TRUE)

dimnames(dvs) <- list(seq(from = 1,
                           to = 3,
                           by = 1),
                       c("Outcome", "Description"))

kable(dvs)
```

Outcome	Description
DWinNxt	Dem Win t + 1
DPctNxt	Dem % t + 1
DifDPNxt	Dem % Margin t + 1

Baseline Covariates

The relevant baseline (i.e., pre-treatment) covariates are:

```r
covs <- matrix(c('DWinPrv', 'Dem Win t - 1',
                 'DPctPrv', 'Dem % t - 1',
                 'DifDPPrv', 'Dem % Margin t - 1',
                 'IncDWNOM1', 'Inc\'s D1 NOMINATE',
                 'DemInc', 'Dem Inc in Race',
                 'NonDInc', 'Rep Inc in Race',
                 'PrvTrmsD', 'Dem\'s # Prev Terms',
                 'PrvTrmsO', 'Rep\'s # Prev Terms',
                 'RExpAdv', 'Rep Experience Adv',
                 'DExpAdv', 'Dem Experience Adv',
                 'ElcSwing', 'Partisan Swing',
                 'CQRating3', 'CQ Rating {-1, 0, 1}',
                 'DSpndPct', 'Dem Spending %',
                 'DDonaPct', 'Dem Donation %',
                 'SoSDem', 'Dem Sec of State',
                 'GovDem', 'Dem Governor',
                 'DifPVDec', 'Dem Pres % Margin', ## average over decade
                 'DemOpen', 'Dem-held Open Seat',
                 'NonDOpen', 'Rep-held Open Seat',
                 'OpenSeat', 'Open Seat',
                 'VtTotPct', 'Voter Turnout %',
                 'GovWkPct', 'Pct Gov\'t Worker',
                 'UrbanPct', 'Pct Urban',
                 'BlackPct', 'Pct Black',
                 'ForgnPct', 'Pct Foreign Born'),
               ncol = 2,
               byrow = TRUE)

dimnames(covs) <- list(seq(from = 1,
                           to = 25,
                           by = 1),
                       c("Covariate", "Description"))

kable(covs)
```

Covariate	Description
DWinPrv	Dem Win t - 1
DPctPrv	Dem % t - 1
DifDPPrv	Dem % Margin t - 1
IncDWNOM1	Inc's D1 NOMINATE
DemInc	Dem Inc in Race
NonDInc	Rep Inc in Race
PrvTrmsD	Dem's # Prev Terms
PrvTrmsO	Rep's # Prev Terms
RExpAdv	Rep Experience Adv
DExpAdv	Dem Experience Adv
ElcSwing	Partisan Swing
CQRating3	CQ Rating {-1, 0, 1}
DSpndPct	Dem Spending %
DDonaPct	Dem Donation %
SoSDem	Dem Sec of State
GovDem	Dem Governor
DifPVDec	Dem Pres % Margin
DemOpen	Dem-held Open Seat
NonDOpen	Rep-held Open Seat
OpenSeat	Open Seat
VtTotPct	Voter Turnout %
GovWkPct	Pct Gov't Worker
UrbanPct	Pct Urban
BlackPct	Pct Black
ForgnPct	Pct Foreign Born

1.2 Core RDD Assumptions

Let $W_0 = [\underline{r}, \overline{r}]$, where $\underline{r} < r_0 < \overline{r}$, denote the window around the cutpoint (or threshold value), r_o, that sorts units into treatment or control.

And, let $F_{R_i | R_i \in W_0}(r)$ denote the conditional distribution function of the running variable R_i given $R_i \in W_0$ for each unit i.

The primary assumption of the regression discontinuity design is:

(1) **Assumption 1**: $F_{R_i|R_i \in W_0}(r) = F(r)$.

In other words, within a window around the cutpoint, r_0, the probability distribution of the running variable is the same for all units. In other words, the running variable is "as-if" randomly assigned within the window, W_0.

A second core assumption is that the running variable is related to potential outcomes, (y_c, y_t), through only $\mathbf{Z}_{W_0}$:

(2) **Assumption 2**: $y_i(\mathbf{r}) = y_i(\mathbf{z}_{W_0}) \; \forall \; \mathbf{r}$.

Question for Students:

- Translate (2) above into words.

1.3 Optimal Bandwidth Selection

We know from Berger et al. (1988); Hansen and Sales (2015); Rosenbaum (2008) that "if a researcher pre-specifies a sequence of hypotheses and corresponding level-α tests, tests those hypotheses in order, and stops testing after the first non-rejected hypothesis, then the probability of incorrectly rejecting at least one correct hypothesis is at most α" (Hansen and Sales, 2015, p. 185).

As applied to bandwidth selection in the RDD context, the SIUP implies that one should start with a set of candidate bandwidths and sequentially test for covariate balance (beginning from either the largest candidate bandwidth or the smallest candidate bandwidth).

Questions for Students:

- Give one reason why we might want to begin testing from the *largest* candidate bandwidth?
- Give one reason why we might want to begin testing from the *smallest* candidate bandwidth?

Let's specify a set of candidate bandwidths and then sequentially test covariate balance. Before actually testing, though, we want to specify a balance criterion and then maximize effective sample size subject to that criterion.

```
bal_fmla <- reformulate(covs[1:25], response = "DemWin")

candidate_bands <- seq(from = -5,
                       to = 5,
                       by = .1)
```

Now let's first filter our dataset and check for balance in the largest candidate bandwidth spanning from -5 to 5.

```
lower_bound <- seq(from = -5, to = -0.1, by = 0.1)

upper_bound <- seq(from = 0.1, to = 5, by = 0.1) %>%
  sort(decreasing = TRUE)

rdd_data %<>% filter(DifDPct > lower_bound[1] & DifDPct < upper_bound[1])

rdd_data %$% summary(DifDPct)


##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -4.99200 -2.49500 -0.13290  0.03576  2.80600  4.98700


xBalance(fmla = bal_fmla,
         data = rdd_data,
         report = "chisquare.test")


## ---Overall Test---
##         chisquare df p.value
## unstrat       569 45 8.9e-92
```

Question for Students:

- What can we infer from the results of the Chi-Squared balance test above?

Now let's write a function to perform this same procedure over all candidate bandwidth sizes beginning with the largerst candidate bandwidth and subsequently testing smaller and smaller bandwidths in order.

```
chi_squared_balance <- function(i,
                                running_var,
                                bal_fmla,
                                data){

  # Preliminaries
  suppressMessages(stopifnot(require(dplyr, quietly = TRUE)))
  suppressMessages(stopifnot(require(magrittr, quietly = TRUE)))
  suppressMessages(stopifnot(require(RItools, quietly = TRUE)))

  lower_bound <- seq(from = -5, to = -0.1, by = 0.1)

  upper_bound <- seq(from = 0.1, to = 5, by = 0.1) %>%
  sort(decreasing = TRUE)

  data %<>% filter(running_var > lower_bound[i] & running_var < upper_bound[i])

  # Effective Sample Size
  ess <- nrow(data)

  p_value <- xBalance(fmla = bal_fmla,
                      data = data,
                      report = "chisquare.test")$overall[[3]]

  bands <- cbind(ess, p_value)

  return(bands)

  }

is <- seq(from = 1,
          to = length(seq(from = -5,
                          to = -0.1,
                          by = 0.1)),
                      by = 1)

cl <- makeCluster(parallel::detectCores())

band_df <- data.frame(t(parSapply(cl, is,
       chi_squared_balance,
       running_var = rdd_data$DifDPct,
       bal_fmla = bal_fmla,
       data = rdd_data))) %>%
  rename(ess = X1,
         p_value = X2)

parallel::stopCluster(cl)

kable(band_df)
```

ess	p_value
856	0.0000000
843	0.0000000
826	0.0000000
811	0.0000000
797	0.0000000
782	0.0000000
767	0.0000000
751	0.0000000
725	0.0000000
701	0.0000000
675	0.0000000
652	0.0000000
626	0.0000000
613	0.0000000
600	0.0000000
586	0.0000000
565	0.0000000
547	0.0000000
525	0.0000000
515	0.0000000
499	0.0000000
474	0.0000000
456	0.0000000
429	0.0000000
418	0.0000000
404	0.0000000
387	0.0000000
375	0.0000000
357	0.0000001
339	0.0000204
322	0.0008590
302	0.0053269
287	0.0022828
274	0.0040121
257	0.0056912
238	0.0310896
219	0.0820914
204	0.0875435
190	0.1070626
176	0.1010950
168	0.1321781
146	0.1411153
136	0.1775630
122	0.2477234
107	0.3565075
85	0.3693273
70	0.5784824
55	0.5074914
37	0.6061527
19	0.4556526

Questions for Students:

- There's something interesting going on as we make the window around the cut point smaller and smaller. What is it?
- And what are its implications, which are also mentioned by Caughey and Sekhon (2011)?

1.4 Local Random Assignment Does *Not* Imply Exclusion Restriction

Now let's assume that within a certain bandwidth around the cutpoint, W_0, the assumption of a local randomized experiment—described in (1) above—obtains. Yet even if this assumption obtains, the running variable, R_i, might still relate to potential outcomes through a mechanism other than Z_{iW_0}.

Question for Students:

- Explain why we refer to a violation of the sort described directly above (and in (2)) as a violation of the *exclusion restriction*?

(2) is a strong assumption because treatment assignment, Z_{iW_0}, is a deterministic function of the running variable, R_i, which implies that treatment and control groups, by construction, will be imbalanced on the running variable. Thus, if R_i relates to (y_{ic}, y_{it}), through a mechanism other than Z_{iW_0}, then the exlcusion restriction is violated.

What can we do if the exclusion restriction is violated? One approach is to model potential outcomes as a function of the running variable, and then to "de-trend" (or "transform") the outcome variable and to subsequently make the claim of "residual ignorability."

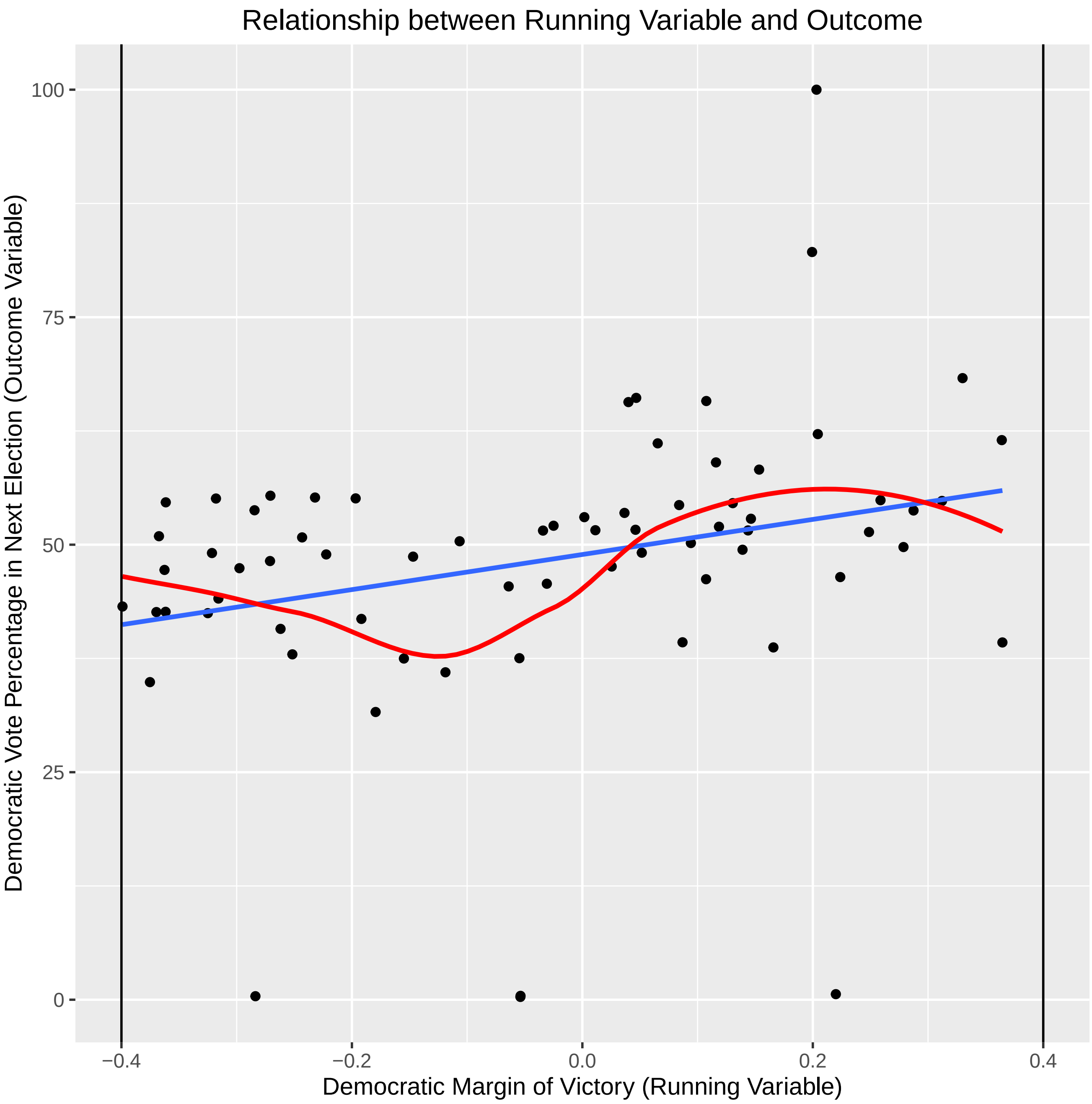
```
rdd_data %<>% filter(DifDPct > lower_bound[47] & DifDPct < upper_bound[47])

rdd_data %>% nrow


## [1] 70


rdd_data %<>% mutate(resid_DPctNxt = resid(lm(DPctNxt ~ DifDPct, data = rdd_data)))
```

Question for Students:

- Do we think modeling the outcome variable as a linear function of the running variable is appropriate?

- What information might we garner from the plot below?
- And what could we do besides a linear model?

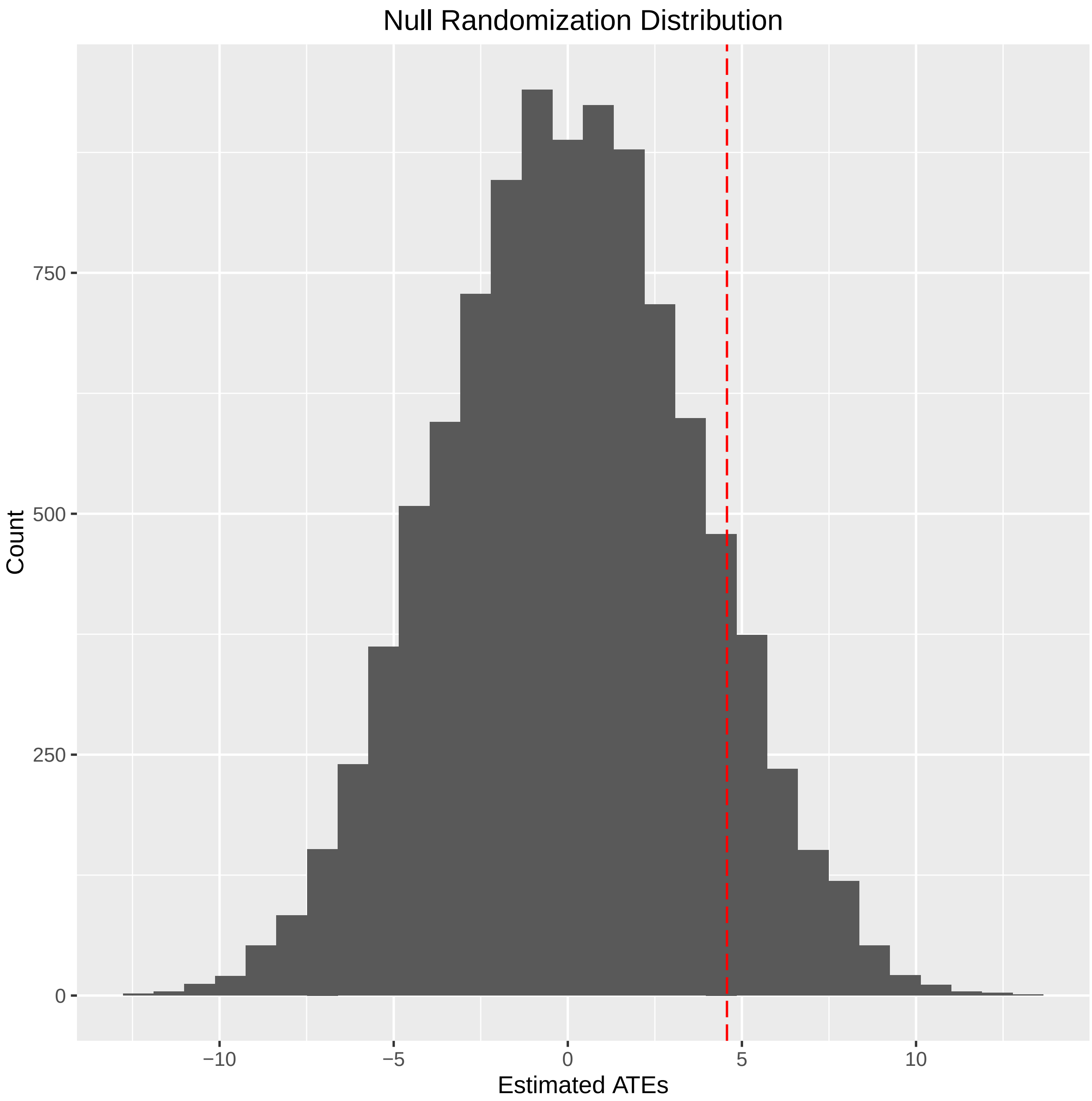
```
ggplot(rdd_data, aes(DifDPct, DPctNxt)) + geom_point() +
  geom_smooth(method = lm, se = FALSE) + geom_smooth(se = FALSE, colour= "red") +
  xlab("Democratic Margin of Victory (Running Variable)") +
  ylab("Democratic Vote Percentage in Next Election (Outcome Variable)") +
  ggtitle("Relationship between Running Variable and Outcome") +
  geom_vline(xintercept = c(-.4, .4))
```



1.5 Outcome Analysis

Consider modeling the relationship between the running variable and the outcome differently, and then perform outcome analysis in a manner similar to below.

```r
est_ate <- coef(lm(resid_DPctNxt ~ DemWin, data = rdd_data))[["DemWin"]]

permute_sharp_null <- function(z, y){

  Z = sample(z)

  est_ate_null = coef(lm(y ~ Z))[["Z"]]

  return(est_ate_null)

  }

rand_dist <- replicate(10^4, permute_sharp_null(z = rdd_data$DemWin,
                                                y = rdd_data$resid_DPctNxt))

qplot(rand_dist, geom = "histogram") +
  labs(title = "Null Randomization Distribution",
       x = "Estimated ATEs",
       y = "Count") + geom_vline(xintercept = est_ate,
                                  colour="red",
                                  linetype = "longdash")
```



```
## One-sided p-value

p_value_one_sided <- mean(rand_dist >= est_ate)

p_value_two_sided <- mean(abs(rand_dist) >= abs(est_ate))
```

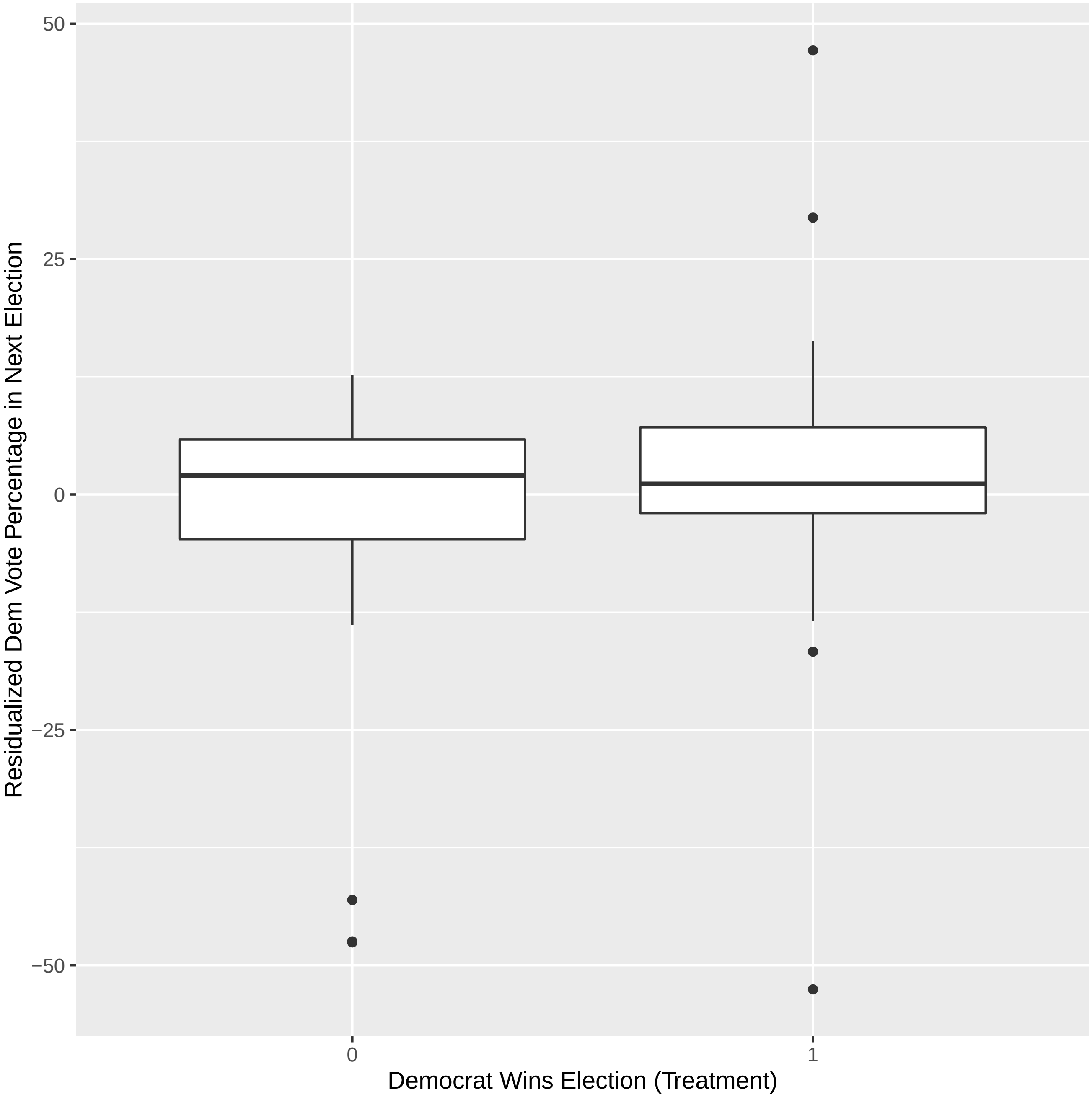
Questions for Students:

- Are we able to reject the sharp null hypothesis of no effect with an $\alpha = 0.05$?
- What could we have done to do to make the randomization distribution under the sharp null

hypothesis tighter?

Notice that in the outcome analysis above we only tested one null hypothesis, which was Fisher's sharp null (Fisher, 1935). What if we wanted to test other hypothetical models of effects?

```
ggplot(rdd_data, aes(factor(DemWin), resid_DPctNxt)) +
  geom_boxplot() + xlab("Democrat Wins Election (Treatment)") +
  ylab("Residualized Dem Vote Percentage in Next Election")
```



For now, let's stick to the constant, additive effect model. Looking only at units within the window, W_0, we will

1. Reconstruct outcomes according to the function $Y_{i0} = Y_i - \tau_i Z_i$;
2. residualize Y_{i0} by regressing it on R_i;
3. calculate the observed test-statistic;
4. Test the sharp null hypothesis of no effect on the reconstructed outcomes Y_{i0};
5. Retain all hypothetical τ_is with a p-value that is at least as large as α.

```
source("Hl_Est_and_Confidence_Set_RDD.R")
```

```
HL_CS <- Confidence_Set_and_HL_Est_RDD(.Y = rdd_data$DPctNxt,
                              .Z = rdd_data$DemWin,
                              .R = rdd_data$DifDPct,
                              .block = NULL,
                              ## Max:
                              #rdd_data %$% {max(DPctNxt[DemWin==1]) -
                              #min(rdd_data$DPctNxt[rdd_data$DemWin==0])}
                              ## Min:
                              #rdd_data %$% {min(DPctNxt[DemWin==1]) -
                              #max(rdd_data$DPctNxt[rdd_data$DemWin==0])}
                              .tau_range = c(-55, 100),
                              .tau_length = length(seq(from = -15,
                                                  to = 15,
                                                  by = .1)),
                              .sims = 10^4,
                              .alpha = 0.05,
                              .cores = parallel::detectCores())

save(HL_CS, file = "HL_CS.RData")
```

```
load("HL_CS.RData")

HL_CS[c(1:3)]


## $lower_cs
## [1] -12.11667
##
## $upper_cs
## [1] 48.85
##
## $HL_est
## [1] 18.36667
```

Question for Students:

- Interpret the results above.

2 Difference-in-Differences

Now let's see if we can perform the same analysis using a difference-in-differences approach. Difference-in-differences entails constructing the "gain score"—in this case, the change in the percentage of votes before the treatment relative to after the treatment—and performing outcome analysis on the "gain score."

This method should increase precision. But, also, insofar as there are no time-varying covariates, then the difference-in-differences approach can also serve as an identification strategy.

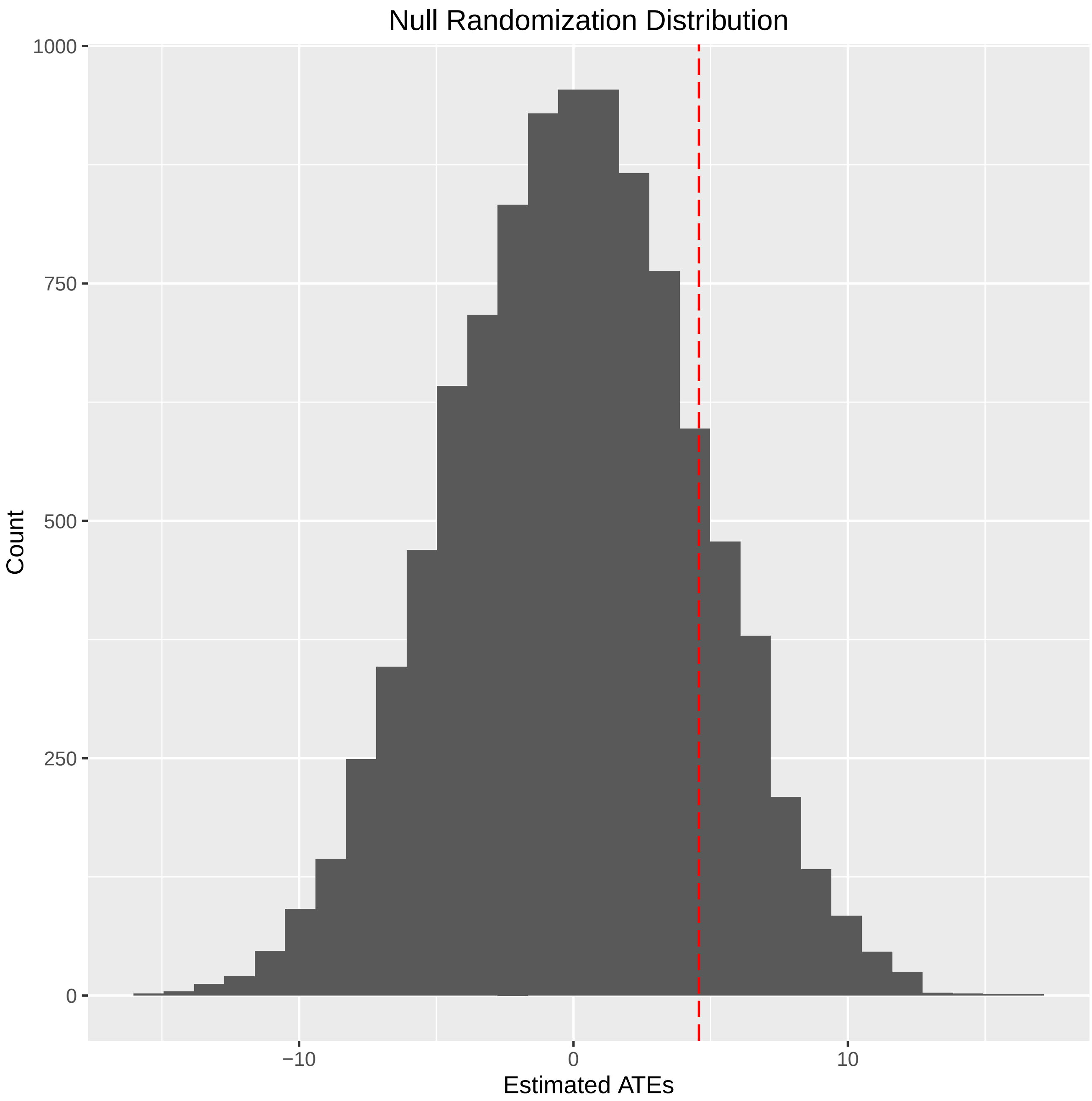
```
rdd_data %<>% mutate(gain_pct = DPctNxt - DPctPrv)

rdd_data %<>% mutate(gain_pct_resid = resid(lm(gain_pct ~ DifDPct, data = rdd_data)))

est_gain_score_ate <- coef(lm(gain_pct_resid ~ DemWin, data = rdd_data))[["DemWin"]]

c(est_ate, est_gain_score_ate)


## [1] 4.570781 4.115518
```

Question for Students:

- How different is the estimated average treatment effect for the "gain score" relative to the estimated ate for the Democratic percentage of votes in the next election?

Let's test the sharp null hypothesis of no effect:

```
rand_dist <- replicate(10^4, permute_sharp_null(z = rdd_data$DemWin,
                                                 y = rdd_data$gain_pct_resid))

qplot(rand_dist, geom = "histogram") +
  labs(title = "Null Randomization Distribution",
       x = "Estimated ATEs",
       y = "Count") + geom_vline(xintercept = est_ate,
                                 colour="red",
                                 linetype = "longdash")
```



```
## One-sided p-value

p_value_one_sided <- mean(rand_dist >= est_ate)

p_value_two_sided <- mean(abs(rand_dist) >= abs(est_ate))

c(p_value_one_sided, p_value_two_sided)


## [1] 0.1571 0.3192
```

Exercise for Students:

- Now, construct a $(1-\alpha)100\%$ confidence interval using the HC2 standard errors discussed in previous classes. Or construct a $(1-\alpha)100\%$ confidence set using the function above making sure to use the "gain score" as the outcome.

References

Berger, R. L., D. D. Boos, and F. M. Guess (1988). Tests and confidence sets for comparing two mean residual life functions. *Biometrics*, 103–115. 6

Caughey, D. and J. S. Sekhon (2011). Elections and the regression discontinuity design: Lessons from close us house races, 1942–2008. *Political Analysis 19*(4), 385–408. 2, 3, 10

Fisher, R. A. (1935). *The Design of Experiments*. Edinburgh, SCT: Oliver and Boyd. 14

Hansen, B. B. and A. Sales (2015). Comment on cochran's "observational studies". *Observational Studies*, 184–193. 6

Rosenbaum, P. R. (2008). Testing hypotheses in order. *Biometrika*. 6