Lab 14 - Regression Discontinuity (RDD) and Instrumental variables (IV):Answers

Eric Parajon, with code adapted from Colin Case and Sean Norton

Regression Discontinuity

Regression discontinuity designs (RDD) exploits the structure of non-random treatment assignment to treatment to give us *as-if random* assignment to treatment. It does this by exploiting treatments that are assigned with respect to the level of another variable.

In less abstract terms, let's consider the motivating example of the first RDD. Thistlewaite and Campbell(1960) study the effect of a merit scholarship on students' academic performance. Of course, the high-performing students were most likely to receive the scholarship, so would do better anyways. There's no existing counterfactual for us to compare to in a standard set-up, such as OLS or difference-in-differences.

However, let's say the scholarship is assigned (either purposefully or unintentionally) based on SAT scores, with a certain score serving as the cutoff between those who received the scholarship and those who didn't. People in the area right around that cutoff are likely to be pretty similar in terms of academic performance; missing the cutoff by just a few points could plausibly be due to random influences like what you ate for breakfast, a mistake on the scantron, etc. In other words, within some small region of the cutoff, assignment to treatment is *as-if* random, and those who didn't receive the scholarship will be a plausible counterfactual. We call the variable that the cutoff exists on the *running variable* or the *forcing variable* - in this case, SAT scores.

There are two types of RDD:

1. *Fuzzy*: the running variable probabilistically increases or decreases assignment to treatment along its range.
2. *Sharp*: has a strict, binary division between treatment and control

We'll be working with sharp RDD in this lab because a) it's simpler and b) the fuzzy RDD is just a different formulation of instrumental variables.

RDD Example: The Incumbency Effect

We'll be working with a dataset from the creators of the `rdrobust` package on Senate races. The outcome variable is `vote`, which is the vote share for the Democratic candidate in that election. Our forcing variable is `margin`, which is that candidate's margin of victory in the *previous election*.

We want to identify the effect of having won the previous election on the subsequent election, i.e the incumbency effect. The obvious problem with this is that states with long-serving Democratic senators are likely to be Democratic strongholds.

However, senators who *barely won* their last election are probably in more competitive state, so by comparing these senators to those who barely lost we can recover a causal effect for experience.

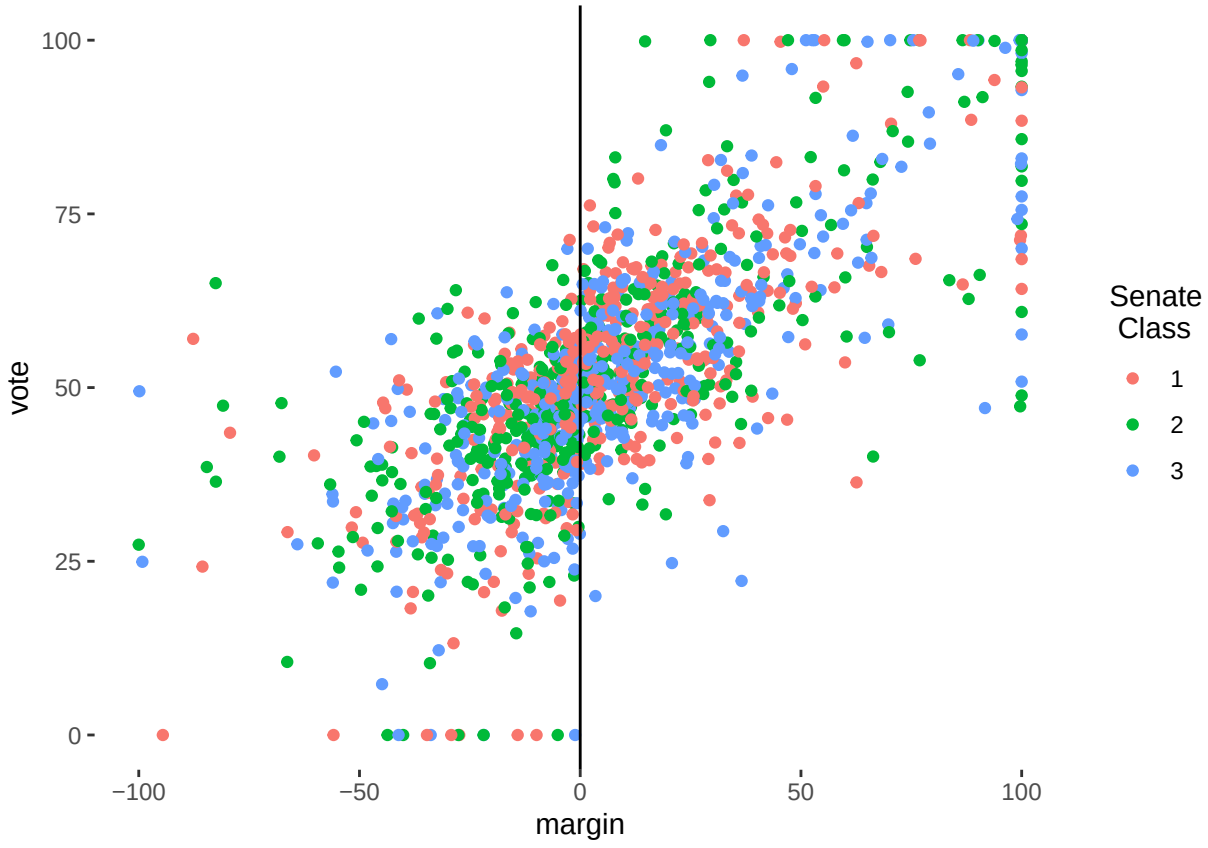
Let's download the data and visualize it.

```
# Clear Environment
rm(list = ls())
```

```
# Set seed
set.seed(1996)
# Load packages
pacman::p_load(haven,rdrobust,rdd,readr,ggplot2,AER)

# Load data
senate <- read_csv('https://raw.githubusercontent.com/rdpackages/rdrobust/master/R/rdrobust_senate.csv')

# Plot  previous vote margin (x) versus vote for current election (y) by Senate class (color)
ggplot(senate, aes(x = margin, y = vote, color = as.factor(class))) +
  geom_point() +
  theme_bw() +
  scale_color_discrete(name = 'Senate \n Class', labels = c('1', '2', '3')) +
  geom_vline(xintercept = 0) +
  theme(legend.position = 'right',
        plot.background = element_blank(),
        panel.grid.minor = element_blank(),
        panel.grid.major = element_blank(),
        panel.border = element_blank())
```

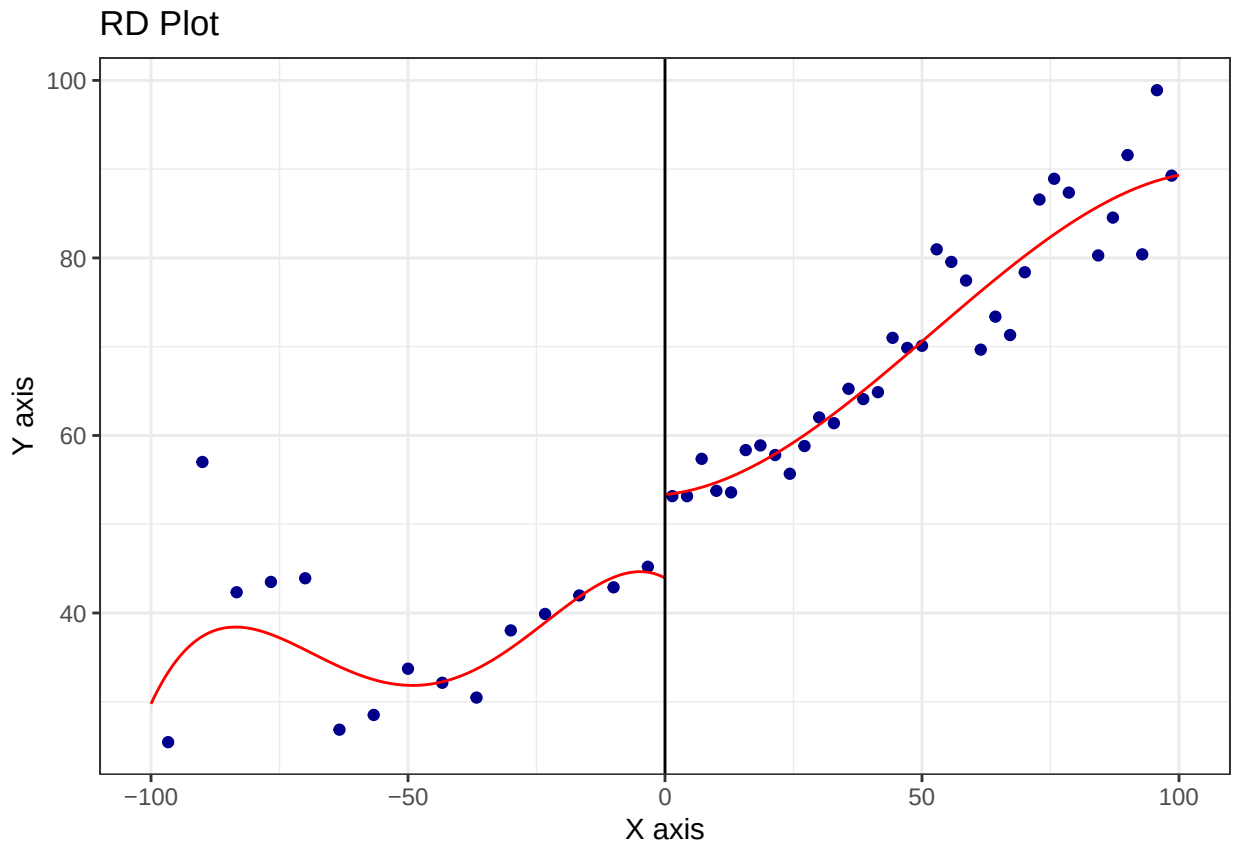


As you can see, we have quite a few observations around the cutoff of 0. This is what you want to see - obviously, observations made on few data points won't be very robust.

To get a sense of if the cutoff actually makes sense, we can use the `rdplot` function.

Consult the documentation and use `rdplot` to visualize how the outcome varies around the cutoff. All of the default arguments that control the bins and the kernel used to fit a line to the data are fine.

```
# Use Rdplot to see how outcome varies on either side of cutoff
rdplot(senate$vote, senate$margin)
```



Seems like there's definitely a jump after the cutoff - good! If you didn't really see anything here, your cutoff is probably incorrect.

Now let's use a simple model, where the outcome is only a function of the running variable. Use the `rdrobust()` function to do so. Summarize it.

```
# Fit model
fit_rd <- rdrobust(senate$vote, senate$margin)
summary(fit_rd)
```

```
## Sharp RD estimates using local polynomial regression.
##
## Number of Obs.                 1297
## BW type                       mserd
## Kernel                   Triangular
## VCE method                       NN
##
## Number of Obs.                 595         702
## Eff. Number of Obs.            360         323
## Order est. (p)                   1           1
## Order bias  (q)                  2           2
## BW est. (h)                 17.754      17.754
## BW bias (b)                 28.028      28.028
## rho (h/b)                    0.633       0.633
```

```
## Unique Obs.                     595          665
##
## ===============================================================================
##         Method     Coef. Std. Err.         z     P>|z|      [ 95% C.I. ]
## ===============================================================================
##   Conventional     7.414     1.459     5.083     0.000     [4.555 , 10.273]
##         Robust         -         -     4.311     0.000     [4.094 , 10.919]
## ===============================================================================
```

We'll ignore most this output in order to move through the lab - but I encourage you to read through the paper introducing the package to understand what's going on under the hood.

What's important for now is the the estimate. In this case the conventional and robust estimate are the same, with the robust estimate having a more conservative z-score. We can interpret this as the effect of being over the cutoff increasing the following election's vote share by 7.4%. In other words, we can determine that having won the past election (even barely) increases the Democrat's vote share in the following election. We were able to recover this estimate despite the fact that Democrats were more likely to win in some states/years than others.

What about covariates though? Obviously there are some things that affect vote share other than previous margin of victory. Let's include those in the RDD estimation.

You'll use the `covs` argument to do so. Includes `termssenate`, `termshouse`, and `class`. Summarize it, and compare it to the estimate with no covariates.

(`covs` expects a matrix. As a hint, you can subset specific named columns of a dataframe or matrix using a character vector and `[]` notation. e.g.: `dataframe[ , c('variable', 'names')]`)

```
# Use rdrobust to estimtate the model with covs argument specified
fit_rd_covs <- rdrobust(senate$vote, senate$margin, covs=senate[,c("termssenate","termshouse","class")])

summary(fit_rd_covs)
```

```
## Covariate-adjusted Sharp RD estimates using local polynomial regression.
##
## Number of Obs.                 1108
## BW type                       mserd
## Kernel                   Triangular
## VCE method                       NN
##
## Number of Obs.                  491          617
## Eff. Number of Obs.             315          283
## Order est. (p)                    1            1
## Order bias  (q)                   2            2
## BW est. (h)                  18.033       18.033
## BW bias (b)                  28.988       28.988
## rho (h/b)                     0.622        0.622
## Unique Obs.                     491          580
##
## ===============================================================================
##         Method     Coef. Std. Err.         z     P>|z|      [ 95% C.I. ]
## ===============================================================================
##   Conventional     6.850     1.407     4.869     0.000     [4.093 , 9.607]
##         Robust         -         -     4.201     0.000     [3.728 , 10.249]
## ===============================================================================
```

That accounted for some of the effect of an additional term, but the effect still exists and is quite large. We can't actually get estimates for the covariates, because `rdrobust` is using a non-parametric kernel regression

under-the-hood.

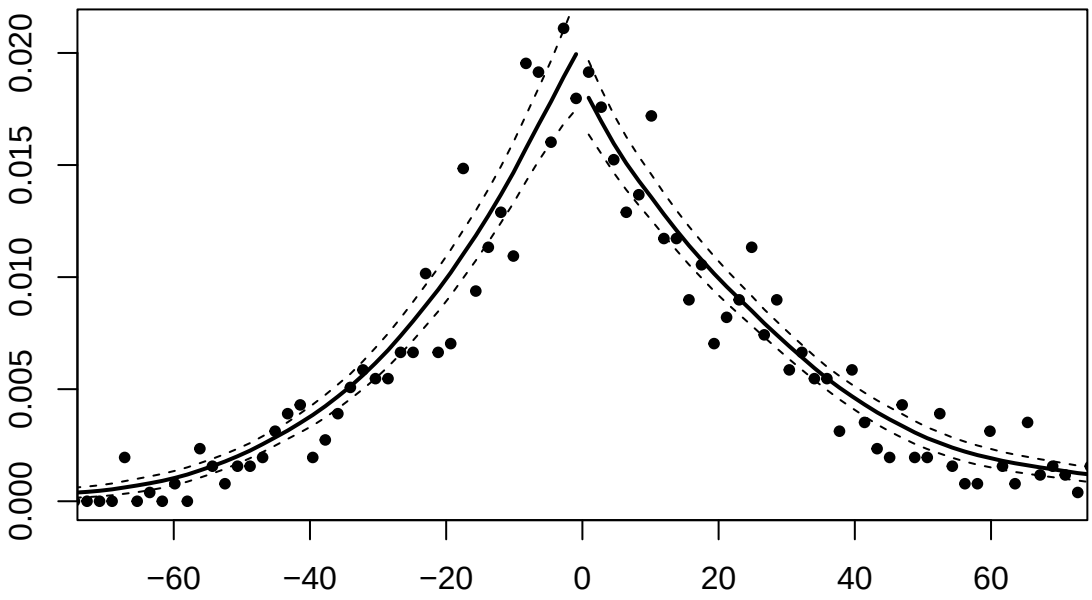
The idea is intuitive as well, which is always a plus when arguing your model can recover causality. A word of caution though: *this only works if the cutoff is exogenous!* Actors are perfectly capable of manipulating cutoffs or scores on the running variable to their advantage, e.g. low margins may reflect electoral fraud in some contexts. Additionally, if you aren't careful in how you define the running variable, you may violate the assumption of as-if random assignment. See Erikson and Rader (2017) for a fun discussion of how the paper this data is from is wrong.Marshall (2022) also just published a paper in AJPS that is somewhat critical of this design for certain applications and how the assumptions can be violated.

The McCrary Test

One of the key assumptions about the RD design is that units cannot "choose" where they are vis-a-vis the running variable - i.e. respondents can't choose to be before the running variable, or after - so they can't "select" themselves to the treatment/ control conditions. This can be an issue with as-if random experiments. For example, take Germany's 5% representation threshold as an example - that parties will only get a seat in the Bundestag if they get 5% of votes nationally. Since the institution is well-known to the players of the game, they may try in every way to get into the parliament, so in that case the treatment is not randomly assigned.

To test this assumption, we can conduct what we call a McCrary Test (aka Density Test). The McCrary test plots a histogram of the running variable. With that histogram we plot the density of the distribution of the running variable. We then examin the density around the cut-off. We are looking at whether the respondents could somehow get through the cut-off. If we see that around the cut-off, i.e. if there's a sudden decrease on either side of the cut-off, then there's some evidence suggesting that the respondents do have an influence on where they are in the running variable. In other words, the cut-off assignment is not entirely exogeneous to the outcome, and thus the Regression Discontinuity Design could be violated.

```
# Density test
DCdensity(senate$margin, 0)
```




```
## [1] 0.3897849
```

It seems like in this example there's a tiny break between those that are below the cutoff and higher than the cutoff of 0, although the break is not large. The p-value from the function also suggests that we did not reject the null of no non-random assignment (p=.39). However, this could be a sign that the treatment assignment is not entirely random.

Instrumental variables (IV) regression:

To perform IV regression, we use a two-stage least squares (TSLS) function (ivreg()) from the aer package. The formula is relatively straight forward. Assume that we wish to estimate the model

$Y_i = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i} + W_{1i} + u_i$

Where $X1i$ and $X2i$ are endogenous regressors instrumented by $Z1i$, $Z2i$, and $Z3i$ and W_{1i} is an exogenous regressor.

The correct formula in ivreg() would be:

y ~ x1 + x2 + w1 | w1 + z1 + z2 + z3

Example

Below I model an IV model regressing education (educ), age age, actual labor market experience (exper) and its squared term expersq on log wages lwage as the dependent variable. I also use the parents' education (fatheduc and motheduc) as instruments for education. I then conduct an F-test of relevance and a Sargan test for validity.

```r
#Reading in the data
women<- read.csv("http://eclr.humanities.manchester.ac.uk/images/5/5f/Mroz.csv")

#Removing null values for log wages and converting to numeric
women<- subset(women, lwage != ".")
women$lwage<-as.numeric(women$lwage)

#Running iv model using ivreg from the aer package
reg_iv1 <- ivreg(lwage~ educ+age+exper+expersq|
                   .-educ+fatheduc+motheduc, data=women)

#Calling summary on model
summary(reg_iv1)
```

```
##
## Call:
## ivreg(formula = lwage ~ educ + age + exper + expersq | . - educ +
##     fatheduc + motheduc, data = women)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.1017 -0.3216  0.0545  0.3685  2.3496
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.0667186  0.4591101   0.145   0.8845
## educ         0.0609945  0.0315798   1.931   0.0541 .
## age         -0.0003542  0.0049318  -0.072   0.9428
## exper        0.0441960  0.0134524   3.285   0.0011 **
## expersq     -0.0008947  0.0004075  -2.196   0.0287 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6756 on 423 degrees of freedom
## Multiple R-Squared: 0.1353,  Adjusted R-squared: 0.1272
## Wald test: 6.085 on 4 and 423 DF,  p-value: 9.085e-05
```

```r
#Diagnostics:
#Conducting an F-test
#Running first_stage model
first_stage <- lm(educ~fatheduc+motheduc,data=women) #regress X on Z
#Running null model
null <- lm(educ~1,data=women)
relevance_test <- waldtest(first_stage, null)
relevance_test
```

```
## Wald test
##
## Model 1: educ ~ fatheduc + motheduc
## Model 2: educ ~ 1
##   Res.Df Df     F    Pr(>F)
## 1    425
## 2    427 -2 55.83 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
#Sargan test
sargan <- lm(reg_iv1$residuals ~ age+exper+expersq+fatheduc+motheduc,data=women)
r.squared<-summary(sargan)$r.squared
sargan_test <- r.squared*nrow(women)

(1-pchisq(sargan_test,1))  #prints p-value
```

```
## [1] 0.5260076
```

```r
#The null hypothesis of the Sargan test implies all instruments are valid.
#So the high p value confirms that we can accept H0, that is all instruments are valid.

#Based on the output, I find that the instruments are both relevant (confirmed by F-test) and valid (co
```

Exercise: Do Voters change the mind of their representatives? Evidence from the U.S. House.

For the exercise we are going to replicate the paper by @lmb2004 : *Do Voters Affect or Elect Policies? Evidence from the U.S. House* in the QJE (referred to as LMB below). You don't have to read the paper, but essentially in their paper they are trying to test which model of public policy is more plausible: Do elections push elected officials somewhere to the middle (i.e. following the Downsian Model of voting), such that the resultant policy outcomes are all clustered around the median voter? Or do elections push the politician to a specific direction? If politicians cannot make credible promises to moderate their policies, then politicians would "change their minds" after an election, to follow the general ideological direction expressed by the electorate in the election.

To test this proposition, LMB fitted two simple regression discontinuity models. They used the ADA Score and the DW-Nominate Scores (figures I and VI respectively in the paper), both are quantitative measures of the ideological position of representatives. They argue that the running variable to the Democratic Vote Share at time t - So if democrats receive more than half the share of votes, they would win in that district and a democratic representative would be elected. If elections can coerce the politician to change their policy predispositions, then at time $t+1$ we should observe a movement in the ADA Score and DW-Nominate Score (Increase in the former - converging to 100; and decrease in the latter, converging to -1).

Please replicate the regression discontinuity analysis they performed - fit the model and plot the results. `realada` is the ADA score and `dwnom1` is the DW-Nominate Score. The running variable is the lagged vote share of democrats `lagdemvoteshare`. Note that in the original paper they reported cluster-robust SEs, so you would have to do the same with `clusterid`. (Hint: check out the `cluster` parameter in the call to `rdrobust()`).

Then, conduct the McCrary Test and verify that the House Representatives did not get to "choose" whether they would win the election or lose it (which is quite a hard test!).

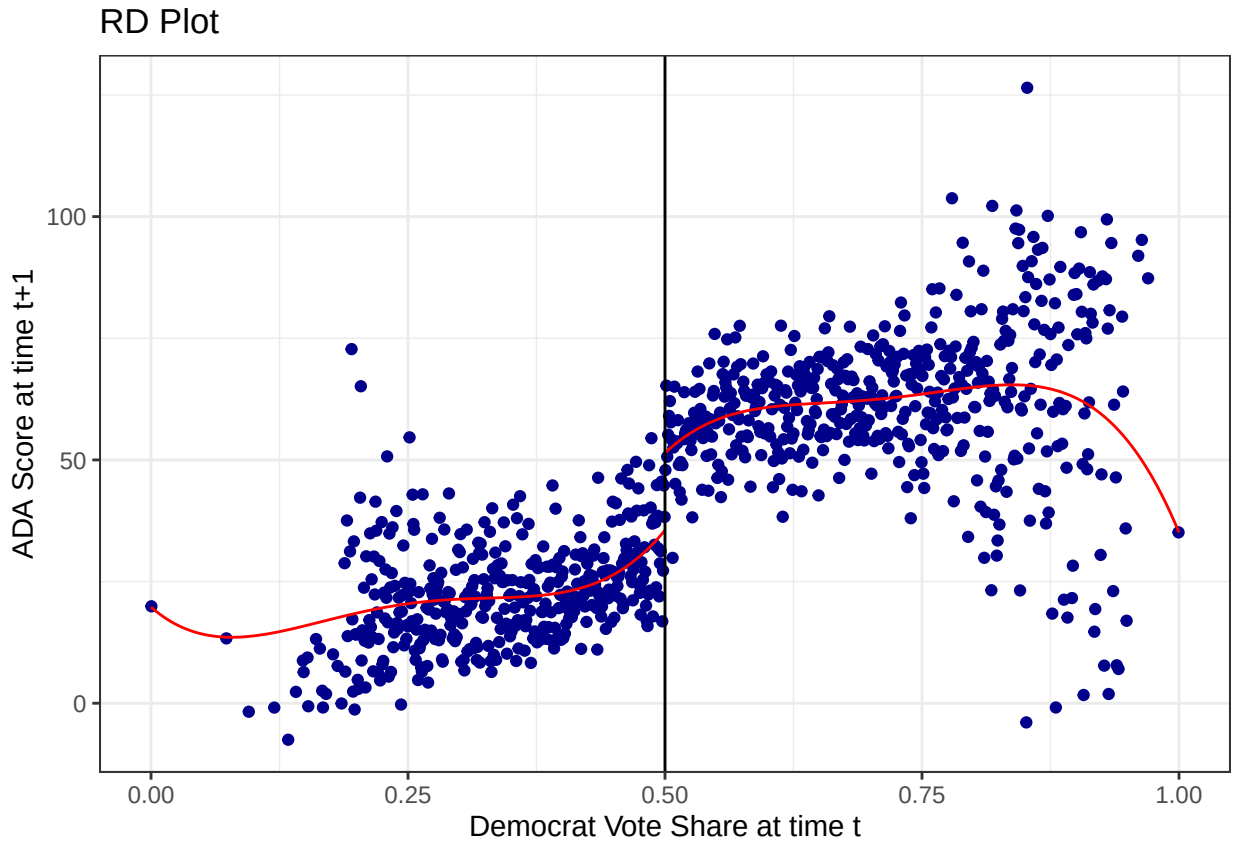
Solution

```r
# Load Data
enricoall4 <- read_dta("enricoall4.dta")

# Fit RDD Model specifying clustered standard errors for ADA as DV
lmb.fit <- rdrobust(enricoall4$realada,
                    enricoall4$lagdemvoteshare,
                    cluster=enricoall4$clusterid,
                    c=0.5)
```

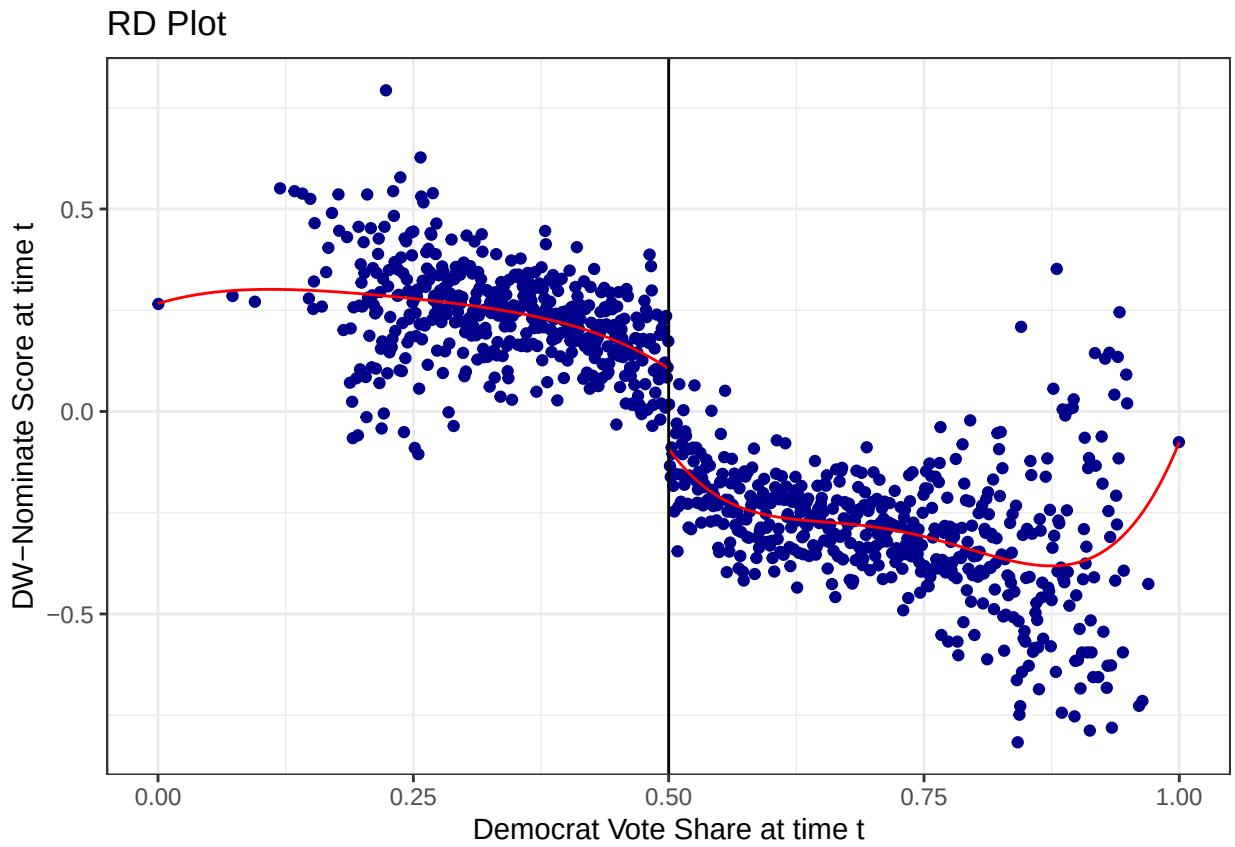
```
# Plot RDD
rdplot(enricoall4$realada,
       enricoall4$lagdemvoteshare,
       x.label="Democrat Vote Share at time t",
       y.label="ADA Score at time t+1",
       c=0.5)
```

```
## [1] "Mass points detected in the running variable."
```




```
# Fit RDD Model specifying clustered standard errors for DW-Nom as DV
lmb.fit2 <- rdrobust(enricoall4$dwnom1,
                     enricoall4$lagdemvoteshare,
                     cluster=enricoall4$clusterid,
                     c=0.5)
# Plot RDD
rdplot(enricoall4$dwnom1,
         enricoall4$lagdemvoteshare,
       x.label="Democrat Vote Share at time t",
       y.label="DW-Nominate Score at time t",
         c=0.5)
```

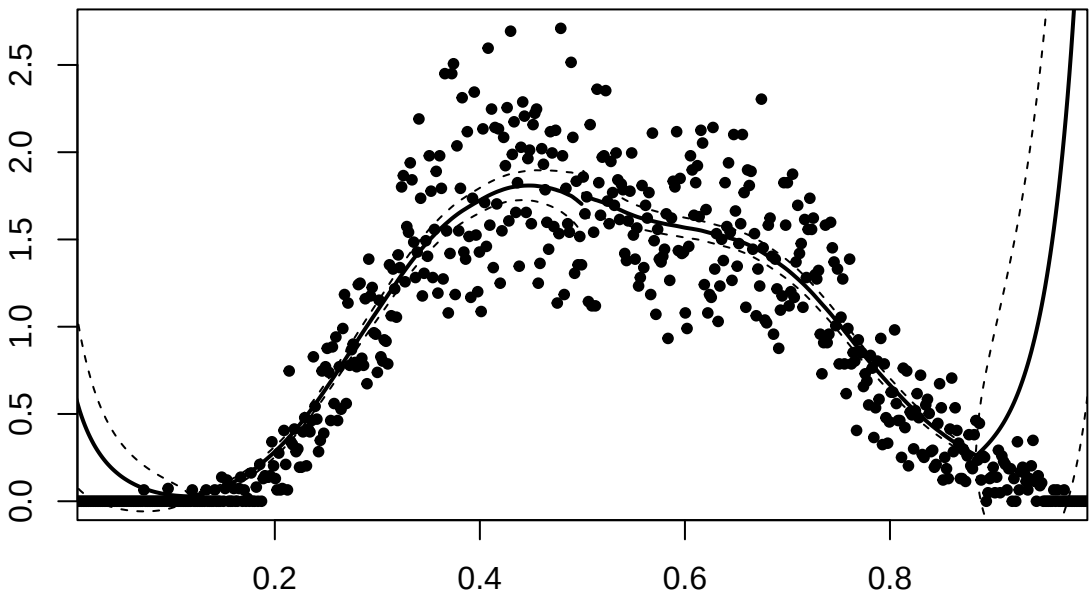
```
## [1] "Mass points detected in the running variable."
```



Yes, we have successfully replicated their results found in the paper.

McCrary Test

```
#McCrary Test
DCdensity(enricoall4$lagdemvoteshare, 0.5)
```




```
## [1] 0.1309416
```

The McCrary Test does not show sharp jumps at around the threshold (c=0.5), but it does alert us that there are heavy tails in the distribution of the running variable. The heavy tails come from Republican strongholds (Democrats have no chance of winning) and Democrat strongholds (Democrats always win). You can also see this above with the lack of observations around the cutpoint. This does lead to some concerns about the robustness of the RD effect we found, and particularly in the sudden reversal of RD trends at the upper tail of the running variable. Substantively this makes sense - if Democrats always win a district, then there's no point to adjust their policy predispositions to election results!