Tutorial 3: Regression Discontinuity Design

Alexander Wintzéus*

February 8, 2024

In this tutorial, we will discuss Regression Discontinuity Design (RDD). To introduce the implementation of related methods in `Stata`, we will rely on the empirical application in Lee (2008). As before, the exercises are collected in the first part of this document. Theoretical concepts are reviewed in the second part.

1 Application

One of the most striking facts of congressional politics in the U.S. is the consistently high rate of electoral success of incumbents. In any given election year, the incumbent party in a given congressional district will most likely win. Between 1948 and 1998 this re-election rate was about 90%.

As noted by Lee (2008), the casual observer is tempted to interpret this figure as evidence that there is an *incumbency advantage* – that is, elected House Representatives may be using their privileges and resources of office to gain an unfair advantage over potential challengers. They seem to be suggesting that there may be a causal influence of winning an election on the probability that a candidate will run for office again and win the next election.

It is, however, well-known that a simple comparison of incumbent and non-incumbent electoral outcomes does not represent anything about a potential incumbency advantage. Indeed, incumbents are, by definition, politicians who were successful in the previous election. If what makes these politicians successful is to some extent persistent over time, they should be expected to be more successful in a subsequent election.

To estimate the incumbency advantage in the U.S., defined as the overall causal impact of being the current incumbent party in a congressional district on the votes obtained in the district's election, Lee (2008) exploits the RDD inherent in the U.S. congressional electoral system. That is, being the incumbent party in a congressional district is a deterministic function of that party's vote shares in the prior election.

*KU Leuven, Department of Economics. alexander.wintzeus@kuleuven.be

To fix ideas, let Z denote the *running variable*. In the application, this variable will correspond to the Democratic vote share margin of victory in election year t. It is defined as the difference in the vote share between the Democratic party nominee and the strongest opponent – virtually always a nominee of the Republican party. The treatment variable D is a dummy indicating whether the Democrat has won the election in year t, and hence, has become the incumbent party for a subsequent election. Clearly, the treatment D will equal one if and only if the running variable Z exceeds zero. Lee (2008) analyzes four different outcomes Y, one of which we will consider here. In particular, the Democratic vote share in election year $t + 1$.

Load the dataset `Lee2008-JoE.dta` into `Stata` and get to know its contents by using the `browse` and `describe` commands.

1. One of the most powerful aspects of Regression Discontinuity (RD) is the ability to present results graphically. As such, making clear and compelling graphs is paramount.

   (a) Plot the empirical relationship between `demsharenext` (Democratic vote share in next election year) and the running variable `difdemshare` (Democratic vote share margin in current election year) using the `scatter` command. Is this graph at all informative about the possible discontinuity in the outcome at $Z = 0$?

   (b) Create a dummy variable `window` using the `generate` command that indicates whether observations are within a 0.25 distance from $Z = 0$. Plot the empirical relationship between `demsharenext` and `difdemshare` using the `scatter` command for those observations for which `window` equals one. Make use of the `xline` option to draw a vertical line at $Z = 0$. Based on this graph, does there seem to be a discontinuity in the outcome at the cutoff? What else can we learn from this graph?

   (c) Create a variable containing equally-spaced bins for `difdemshare` using the `egen` command with the `cut` function. Use the `at` option to communicate to `Stata` that the bins should be of width 0.005 and should cover the range of the running variable. Can you explain why this variable will contain missing values for some observations?

   (d) Compute the mean of `demsharenext` and `difdemshare` within each bin using the `collapse` command. Provide the lower bounds of the bins stored in the designated variable via the `by` option. Afterwards, plot the empirical relationship between the binned versions of `demsharenext` and `difdemshare` using the `scatter` command for those observations for which `window` equals one. Are you more or less convinced about the existence of a discontinuity in the outcome at the cutoff?

   (e) Repeat the previous two steps, but considering, in turn, bins of width 0.001 and 0.05. What can we learn from this exercise in regard to the choice of bins?

2. We now turn to estimating the size of the treatment effect. That is, what is the causal impact of being the incumbent party on the vote shares obtained in the subsequent election?

In a Regression Discontinuity Design (RDD) setting, estimation may be particularly cumbersome. We will implement both parametric and nonparametric estimation techniques in `Stata`. In this question, we will discuss parametric estimation. Nonparametric techniques are discussed in the next question.

(a) Consider the following parametric model:

$$Y_i = \beta_0 + \beta_1 D_i + \beta_2 Z_i + \beta_3 Z_i^2 + \beta_4 D_i \times Z_i + \beta_5 D_i \times Z_i^2 + \varepsilon_i. \tag{1}$$

Create a dummy variable `demwin` indicating whether the Democratic nominee is the current incumbent party. Estimate equation (1) by OLS on the full set of observations. You can use the `robust` option. What is the treatment effect? What do the results suggest about the size of the treatment effect?

(b) Create a dummy variable `bandwidth` indicating whether observations are within a 0.10 distance from $Z = 0$. Estimate equation (1) by OLS on the restricted set of observations. What do the results suggest about the size of the treatment effect? Do you think the proposed parametric model provides a good fit within the chosen bandwidth?

3. Nonparametric estimation techniques allow for the data to inform us about the shape of the conditional mean function rather than imposing a specific functional form. In what follows, we will rely on *kernel-weighted local polynomial smoothing* to nonparametrically estimate the conditional mean function and the treatment effect. This technique can be implemented in `Stata` with the `lpoly` command. It is useful to consult its documentation using the `help` command before proceeding.

(a) Use the `lpoly` command to nonparametrically estimate the conditional mean function on each side of the cutoff. Set the kernel weighting function to the triangular kernel with the `kernel` option. Use the `bwidth` option to set the bandwidth to 0.1. Allow for polynomials of degree two by providing the `degree` option. Save the grid points and local predictions in designated variables by supplying the `gen` option with chosen variable names. Finally, provide the `nograph` option to let `Stata` suppress graphical output.

(b) Plot the grid points and local predictions on each side of the cutoff in a single graph using the `twoway scatter` command. Make use of the `xline` option to draw a vertical line at $Z = 0$. Does it matter that we are performing the estimation using the full set of observations?

(c) Create a variable `cutoff` using the `generate` command that is simply equal to zero for the first observation; this can be achieved by adding `in 1` at the end. Now redo the estimation on both sides of the cutoff, but supply the additional option `at(cutoff)`. The treatment effect is then simply given by the difference in the local predictions – these are stored in the first row under the designated variables. Compare the estimate to the ones obtained with the parametric approach.

4. A common critique with respect to nonparametric techniques is that we are replacing one set of assumptions with another set of assumptions: What bandwidth should we choose? What degree of polynomial for the local approximations? Importantly, it is not as straightforward to obtain standard errors for our estimate of the treatment effect as with the parametric approach. Much of the work on nonparametric methods in RDD has focused on finding optimal ways of choosing bandwidths and estimating standard errors. All these features are present in the command `rdrobust` described in Calonico, Cattaneo, Farrell, and Titiunik (2017).

   (a) Look at the `help` file of the `rdrobust` command. Estimate the treatment effect using the `rdrobust` command. You can use similar options as with `lpoly`. However, let the command choose the optimal bandwidth. In addition, supply the `all` option. Discuss the results.

   (b) Look at the `help` file of the `rdplot` command. Use `rdplot` to create a binned scatter plot of the relationship between `demsharenext` and `difdemshare` including a second-order polynomial fit with optimally chosen bandwidth.

5. There is more to learn about Regression Discontinuity Designs. In particular, we have not touched upon testing of assumptions underlying RDD. There are a number of different kinds of tests that exist, such as balance tests and bunching or manipulations tests. The companion command `rddensity` is able to implement some of these. I leave it to you to explore this.

2 Theoretical summary

In this section, we will briefly summarize the main theoretical concepts regarding Regression Discontinuity Design (RDD). We will primarily focus on sharp RDD and related nonparametric estimation techniques.

2.1 Sharp RDD

Assume we have a binary treatment D_i. A key feature of RDD is that there is a continuous variable Z_i that determines who gets treated.[1] This variable is referred to as the *running variable* or *forcing variable*. In a sharp design, the value of the running variable Z_i completely determines treatment assignment. If the value of Z_i exceeds some cutoff c, then unit i is treated.[2] In other words, treatment D_i is a deterministic function of Z_i.

To estimate treatment effects in this particular setting, we rely on one crucial assumption:

1. **Continuity assumption:** The conditional mean functions $E[Y_i(1)|Z_i = z]$ and $E[Y_i(0)|Z_i = z]$ are continuous in z.

Under this assumption, one can identify the following Conditional Average Treatment Effect (CATE):

$$\begin{aligned}
\tau_{CATE} &= E[Y_i(1) - Y_i(0)|Z_i = c] && (2)\\
&= E[Y_i(1)|Z_i = c] - E[Y_i(0)|Z_i = c] && (3)\\
&= \lim_{z \downarrow c} E[Y_i|D_i = 1, Z_i = z] - \lim_{z \uparrow c} E[Y_i|D_i = 0, Z_i = z] && (4)\\
&= \lim_{z \downarrow c} E[Y_i|Z_i = z] - \lim_{z \uparrow c} E[Y_i|Z_i = z] && (5)
\end{aligned}$$

where the final equality follows from the fact that all units above the cutoff are treated and all units below are not. From a more design-based perspective, identification of treatment effects is based on the running variable being like a *randomizer* around the cutoff. If units could for some other, possibly unobserved reason, shift their value of the running variable so as to get treated, then there is some form of self-selection and identification breaks down. In the case treatment is desirable, this may be a real issue.

Estimation of treatment effects may also be particularly difficult for a number of reasons. First, there may be some direct relationship between the running variable Z_i and the outcome Y_i which we have to control for. As we generally do not know the exact relationship, controlling flexibly for the effect of Z_i requires substantially more from the data. Second, we need to estimate counterfactual means *exactly* at $Z_i = c$. However, there may not be a lot of observations for which this is true,

[1]Note that some RDD studies have exploited *discrete* running variables for identification. However, for the purposes of this summary, we will abstract from this possibility.

[2]Often, the running variable is normalized with respect to the cutoff, so that the cutoff corresponds to $Z_i = 0$.

if any. Estimation of treatment effects will thus almost necessarily rely on extrapolation.

In other words, if we get the functional relationship between outcome and running variable wrong, we may not be able to correctly estimate treatments effects. To limit the influence of parametric assumptions on the treatment effect estimates, we will discuss nonparametric estimation techniques in the following section. Nevertheless, extrapolation may still remain somewhat of an issue.

2.2 Kernel-weighted local polynomial regression

Recall that we are interested in τ_{CATE} at the cutoff. This is nothing else than a conditional mean function $E[Y_i(1) - Y_i(0)|Z_i = z]$ for a particular value of z. In general, this conditional mean will be a nonlinear function in z.

Local polynomial regression exploits the fact that the possibly complex conditional mean function can be locally approximated by a polynomial of degree p. For example, local *linear* regression tries to approximate the conditional mean function at a given point by fitting a line through that point. The logic behind local polynomial regression thus closely corresponds to that of locally approximating complex functions by Taylor expansions.

Suppose that we want to approximate the conditional mean function at the point z_0. To this end, we can consider fitting a kernel-weighted polynomial of degree p around z_0:

$$\min_{\{\beta_d\}_{d=0}^{p}} \sum_{i=1}^{n} \big(Y_i - \sum_{d=0}^{p} \beta_d (Z_i - z_0)^d\big)^2 K_h(Z_i - z_0) \tag{6}$$

where $K_h(u)$ denotes the kernel weighting function for a given bandwidth h. This weighting function ensures that only values within the given bandwidth around the point z_0 are given a positive weight in the regression. If one would weight all observations within the bandwidth equally (a uniform kernel), this would correspond to the traditional local polynomial regression approach. However, choosing any other kernel that puts more weight on observations closer to z_0 corresponds to what is known as kernel-weighted local polynomial regression.

The predicted value of the outcome $\hat{Y}_i$ at z_0 will be our local *nonparametric* estimate of the conditional mean function.[3] Note that it will simply equal $\hat{\beta}_0$. One can now easily see that we can nonparametrically estimate the treatment effect by implementing this method around the cutoff c – once using only data above the cutoff and once using only data below. The difference in the obtained estimates provides our estimate for τ_{CATE}.

[3] Calling the estimate *semiparametric* would be more appropriate. Conditional on the chosen bandwidth, the local polynomial regression approach really is a *parametric* method.

2.3 Fuzzy RDD

In a fuzzy RDD, the relationship between the treatment variable D_i and the running variable Z_i is not deterministic. Conditional on Z_i, we can think of D_i as a random variable for which we know that $E[D_i|Z_i]$ changes discontinuously at $Z_i = c$. So while the running variable is a predictor of who gets treated at the cutoff, it does not completely determine treatment assignment.

As a result, fuzzy RDD is simply a special case of IV. Exceeding the cutoff discontinuously changes the probability of receiving the treatment. If treatment matters, exceeding the cutoff will affect outcomes. However, since not all units that exceed the cutoff end up getting treated, the jump in outcomes at the cutoff needs to be rescaled by the jump in the probability of receiving the treatment at the cutoff; which resembles standard IV approaches.

References

Calonico, S., Cattaneo, M. D., Farrell, M. H., & Titiunik, R. (2017). rdrobust: Software for regression discontinuity designs. *The Stata Journal*, *17*(2), 372-404.

Lee, D. S. (2008). Randomized experiments from non-random selection in U.S. house elections. *Journal of Econometrics*, *142*(2), 675-697.