Extrapolation of Treatment Effect in Sharp Regression Discontinuity Design: A Double Machine Learning Approach

Nan Ge

May 20, 2019

Abstract

Regression discontinuity (RD) design is one of the most credible identification strategy in the absence of a randomized experiment. However, canonical RD parameter identifies a local average treatment effect, and has very limited external validity. Angrist and Rokkanen [2015] proposed a set of assumptions and method to extrapolate the RD effects. In this paper, we improved the method by combining it with a double machine learning method based on Chernozhukov et al. [2018].

1 Methods

Angrist and Rokkanen [2015] has proved that when conditional independence assumtpion (CIA) holds, it is safe to extrapolate local average treatment effect (LATE) to average treatment effect (ATE). In section 1.1, I list out the sufficient conditions for CIA assumption. In section 1.2, I'll lay out the plan for our extrapolation. Finally, section 1.3 describes a detailed data generating process for simulation.

1.1 Assumption

Lemma: Assume that running variable can be modeled as: $R_i = g(x_i, \varepsilon_i)$, where $g(x_i, \varepsilon_i)$ is a measurable function, and assume that $\varepsilon_i(Y_i(d), x_i)$, then, CIA holds: $E[Y_i(d)|R_i, x_i] = E[Y_i(d)|x_i]$. Intuitively, the lemma says that if information in the running variable could be fully explained by the covariates we have, we could safely estimate the ATE by adding in the covariates.

1.2 Extrapolation Method

We'll leverage previous Lemma to select covariates that make CIA holds. Suppose that we have: the outcome variable Y_i, the running variable R_i, binary treatment rule D_i, and a large number of covariates x_i that satisfies the lemma in our data.

Our proposed method is: First, split the sample into two folds: train set and test set. Second, use train set data, regress Y_i on x_i given $D_i = j$, where $j \in \{0, 1\}$, select covariates with coefficients that are not zero by LASSO, say, x_{1i}. Third, use train set data, regress R_i and x_i, select covariates with coefficients that are not zero by LASSO, say, x_{2i}. Fourth, combine the selected covariates from these two steps to generate conditional set: z_i. Use test set to estimate $E[Y_i(d)|z_i]$ by estimating $E[Y_i|z_i, D_i = d]$, where $d \in \{0, 1\}$. Finally, Use test set to compute extrapolated ATE by: $E[Y_i(1) - Y_i(0)] = E[E[Y_i(1) - Y_i(0)|z_i]]$.

Notice that step two to four is simply the double LASSO method. In order to compare our method's performance with Angrist and Rokkanen's (AR heceforth), I also used AR's method, and AR with LASSO on each simulated dataset. In an effort to do asymptotic analysis, I also computed the bootstrapped error and the 95% confidence interval by resampling the dataset 100 times.

1.3 Data Generating Process

The DGP is divided into four cases: single cutoff with similar parameters, single cutoff with different parameters, multiple cutoffs of the same type, multiple cutoffs with different types. I'll introduce them one by one.

1.3.1 Single cutoff with similar parameters

$$y_i(0) = 0.8 + 18.3x_{1i} - 13.3x_{2i} - 15.6x_{3i} + \varepsilon_{0i}$$
$$y_i(1) = 1.5 + 23.5x_{1i} + 21.6 * x_{2i} + 22.6 * x_{3i} + \varepsilon_{1i}$$
$$r_i = 1.6 + 6.8 * x_{1i} + 8.3 * x_{2i} + 7.6 * x_{3i} + u_i$$
$$D_i = 1\{r_i > 0\}$$

We have covaraites: x_{4i}up until x_{100i}.

Notice that the coefficients for x_1, x_2 and x_3 are in the same scale in this DGP. I plotted the potential outcomes over running variable in **Figure 1**.

1.3.2 Single cutoff with different parameters

$$y_i(0) = 0.8 + 2.3x_{1i} - 6.3x_{2i} - 15.6x_{3i} + \varepsilon_{0i}$$
$$y_i(1) = 1.5 + 23.5x_{1i} + 1.6 * x_{2i} + 2.6 * x_{3i} + \varepsilon_{1i}$$
$$r_i = 1.6 + 6.8 * x_{1i} + 3.3 * x_{2i} + 3.6 * x_{3i} + u_i$$
$$D_i = 1\{r_i > 0\}$$

Again, we have covaraites: x_{4i}up until x_{100i}.

The biggest difference between case 1 and case 2 is that the coefficients for x_1, x_2 and x_3 are in different scales in DGP2. This will bring a difference to the performance of our feature selecting methods.

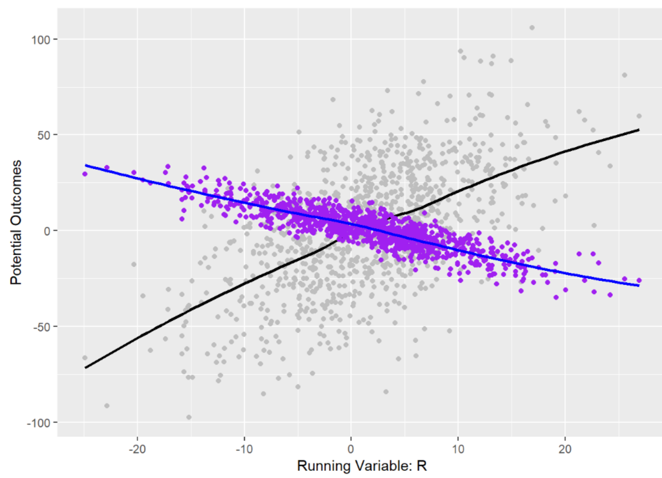

Figure 1: Potential Outcomes over Running Variable (Case 1)

1.3.3 Multiple cutoffs of the same type

$$y_i(0c) = 5.65 + 19.6x_{1i} - 3.6x_{2i} - 5.5x_{3i} + \varepsilon_{0i}$$
$$y_i(1c) = 11.2 + 2.5x_{1i} + 16.9 * x_{2i} + 3.5 * x_{3i} + \varepsilon_{1i}$$
$$r_i = 1.6 + 5.6 * x_{1i} + 8.3 * x_{2i} + 6.6 * x_{3i} + u_i$$

There are two cutoffs: 2 and 5.

The individuals are assigned into two groups suitable for different cutoffs. The scenario is mostly common in education or labor economics, where different groups of people apply to different treatment rules.

1.3.4 Multiple cutoffs with different types

$$y_{i0}(1) = 0.65 + 9.6x_{1i} - 3.6x_{2i} - 3.5x_{3i} + \varepsilon_{0i}$$
$$y_{i1}(1) = 1.2 + 15.5x_{1i} + 3.9 * x_{2i} + 4.5 * x_{3i} + \varepsilon_{1i}$$
$$y_{i0}(2) = 0.5 + 8.6x_{4i} - 3.6x_{5i} - 5.5x_{6i} + 6.3x_{7i} + \varepsilon_{0i}$$
$$y_{i1}(2) = 1.3 + 19.9x_{4i} + 8.8x_{5i} + 3.6x_{6i} + 8.8x_{7i} + \varepsilon_{1i}$$
$$r_{i1} = 1.6 + 5.6 * x_{1i} + 8.3 * x_{2i} + 6.6 * x_{3i} + u_i$$
$$r_{i2} = 2.3 + 6.5 * x_{4i} + 8.5 * x_{5i} + 5.3 * x_{6i} + 7.2x_{7i} + u_i$$

There are two cutoffs: 0 and 5.

In case 3, individuals belong to different groups, but the potential outcomes are generated by the same set of covariates. However, in case 4, potential outcomes in different groups are generated by different set of covariates.

2 Results

2.1 Point Estimation

2.1.1 Single cutoff with similar parameters

Extrapolation results for case 1 are reported in Table 1. The three columns report results using AR's method, AR with LASSO, and the proposed double machine learning method respectively. The above panel reports results from a simulated dataset with 1000 observations, and the second panel has 2000 observations. The first row of each panel reports mean difference in extrapolation. The second row reports mean of absolute extrapolation error. Two observations could be made here. First, the double machine learning method performs better than AR's method. Second, double machine learning is comparatively better in point estimate when sample size is small.

Table 1: Simulation Result for DGP 1 with Single Cutoff

Methodology	AR + Split	AR + LASSO	AR + DML
Extrapolation Error	0.5159	0.04276	-0.08795
Abs(Extrapolation Error)	1.905466	2.06358	1.44159
MSE of Extrapolation Error	10.30503	6.20552	3.141086
N	1000	1000	1000
Extrapolation Error	0.1624	0.164	0.1639
Abs(Extrapolation Error)	0.97363	0.972988	0.972924
MSE of Extrapolation Error	1.487618	1.483372	1.483115
N	2000	2000	2000

2.1.2 Single cutoff with different parameters

Extrapolation results for case 2 are reported in Table 2. When coefficients are in different scale, double machine learning method could achieve much better performance. The intuition is that, if we only do LASSO once, it's very likely that the covariate with small covariate will be left out. Double LASSO could help to select the right covariates.

2.1.3 Multiple cutoffs of the same type

Extrapolation results for case 3 are reported in Table 3. Double machine learning method performs much better than AR or AR with LASSO in the setting of multiple cutoffs, even with large number of observations. Plus, the method is flexible enough to be applied to scenarios where there are more than two cutoffs.

2.1.4 Multiple cutoffs with different types

Finally, extrapolation results for case 4 are reported in Table 4. When there are multiple cutoffs, and potential outcomes are generated differently, double machine

Table 2: Simulation Result for DGP 2 with Single Cutoff

Methodology	AR + Split	AR + LASSO	AR + DML
Extrapolation Error	0.1665	3.7291	0.01781
Abs(Extrapolation Error)	0.799501	3.7404	0.674
MSE of Extrapolation Error	1.072785	15.16516	0.693618
N	1000	1000	1000
Extrapolation Error	0.07641	3.8978	0.07721
Abs(Extrapolation Error)	0.559227	3.991	0.560756
MSE of Extrapolation Error	0.4676639	17.1152	0.4679712
N	2000	2000	2000

Table 3: Simulation Result 2: Two Cutoffs that are Different Types

Cutoff 1			
Methodology	AR + Split	AR + LASSO	AR + DML
Extrapolation Error	1.0139	4.293	0.04702
Abs(Extrapolation Error)	1.248002	4.293	0.332591
MSE of Extrapolation Error	7.9508	18.58	0.157128
N	2000	2000	2000
Cutoff 2			
Methodology	AR + Split	AR + LASSO	AR + DML
Extrapolation Error	1.3691	11.978	0.029917
Abs(Extrapolation Error)	1.717539	11.978	0.495368
MSE of Extrapolation Error	10.2483	144.42	0.3658364
N	2000	2000	2000

learning method performs very well at both cutoffs, and is the best among the three methods in Table 4.

2.2 Bootstrapped Error

In order to study the standard deviation of our extrapolation estimation, I generated 100 datasets, resampled the simulated dataset for 100 times, and estimated the ATE for each sampling dataset. For each simulation, I constructed the confidence interval for 100 extrapolation estimates, and compared it with the true ATE. However, the standard deviation is too large that the confidence interval covers the true ATE with nearly 100% for all three methods. However, as we could observe from Figure 2, the standard deviation from AR with LASSO is very large. Therefore, coverage rate might not be a very good measurement.

3 Conclusion

In this paper, I used double machine learning approach to select relevant covariates that make CIA works. The method can be extended to multiple-cutoffs RD designs.

Table 4: Simulation Result 2: Two Cutoffs that are Same Types

Cutoff 1			
Methodology	AR + Split	AR + LASSO	AR + DML
Extrapolation Error	-0.004317	3.619	0.002138
Abs(Extrapolation Error)	0.506142	3.619	0.6932
MSE of Extrapolation Error	0.3580091	13.835	0.714849
N	2000	2000	2000
Cutoff 2			
Methodology	AR + Split	AR + LASSO	AR + DML
Extrapolation Error	0.01658	4.471	-0.1401
Abs(Extrapolation Error)	0.482994	4.471	0.81516
MSE of Extrapolation Error	0.3478324	20.966	0.986248
N	2000	2000	2000

Table 5: Coverage Probability of Bootstrapped Confidence Interval

DGP 1			
Methodology	AR + Split	AR + LASSO	AR + DML
Coverage Probability	1	1	1
DGP 2			
Methodology	AR + Split	AR + LASSO	AR + DML
Coverage Probability	1	0.99	1

The simulation results show that the double machine learning approach performs better than simple approach in Angrist and Rokkanen (2015) in terms of RD treatment effect extrapolation.

References

Joshua D Angrist and Miikka Rokkanen. Wanna get away? regression discontinuity estimation of exam school effects away from the cutoff. *Journal of the American Statistical Association*, 110(512):1331–1344, 2015.

Victor Chernozhukov, Denis Chetverikov, Mert Demirer, Esther Duflo, Christian Hansen, Whitney Newey, and James Robins. Double/debiased machine learning for treatment and structural parameters, 2018.

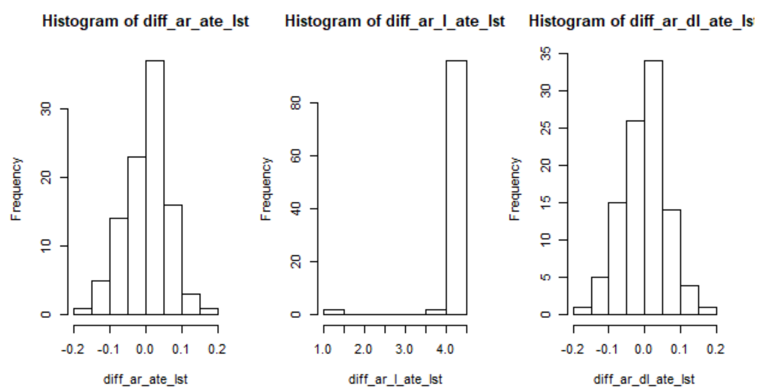

Figure 2: Histogram of Extrapolation Estimates from Case2 (N=2000)