

## Robust Data-Driven Inference in the Regression-Discontinuity Design

Sebastian Calonico  
University of Michigan  
Ann Arbor, MI  
calonico@umich.edu

Matias D. Cattaneo  
University of Michigan  
Ann Arbor, MI  
cattaneo@umich.edu

Rocio Titiunik  
University of Michigan  
Ann Arbor, MI  
titiunik@umich.edu

**Abstract.** This article introduces three Stata commands useful to conduct “robust” data-driven (i.e., fully automatic) statistical inference in the regression-discontinuity (RD) design. First, we present the command `rdrobust` that implements the more robust bias-corrected confidence intervals for sharp RD treatment effects proposed in Calonico, Cattaneo, and Titiunik (2012). This command also implements other conventional nonparametric RD treatment-effect point estimators and confidence intervals. Second, we describe the companion command `rdbwselect`, which implements several bandwidths selectors proposed in the RD literature. Third, we introduce the command `rdbinselect`, which implements a novel data-driven optimal length choice of evenly-spaced bins employed to approximate the underlying regression functions by local sample means, and discuss how this command may be used to construct commonly employed RD plots in empirical applications.

**Keywords:** `st0001`, `rdrobust`, `rdbwselect`, `rdbinselect`, regression discontinuity, treatment effects, local polynomials, bias correction.

This draft: November 20, 2012

## 1 Introduction

The regression-discontinuity (RD) design is by now a well-established and widely used empirical strategy in empirical work. In this design, units receive treatment based on whether their value of an observed covariate is above or below a known cutoff. The key feature of the design is that the probability of receiving the treatment conditional on this covariate jumps discontinuously at the cutoff, inducing variation in treatment assignment that is assumed to be unrelated to potential confounders. Due to its local non-parametric nature, RD treatment effects are usually constructed using local-polynomial nonparametric regression, and statistical inference is based on large-sample approximations. For review and further discussion see, among others, Hahn et al. (2001), Porter (2003), Lee (2008), Cook (2008), Imbens and Lemieux (2008), van der Klaauw (2008), Lee and Lemieux (2010), Dinardo and Lee (2011) and Imbens and Kalyanaraman (2012).

In this article we discuss data-driven, local-polynomial based, robust inference procedures in the RD design. We introduce three main commands that together offer an array of data-driven nonparametric inference procedures, including point estimators and confidence intervals, useful for RD empirical applications. First, our main command `rdrobust` implements the more robust (to “large” bandwidth choices) bias-corrected inference procedure proposed by Calonico, Cattaneo, and Titiunik (2012). This command also offers an implementation of other classical inference procedures employing local-polynomial regression, as suggested in the recent econometrics literature (see references given in the previous paragraph).

To construct any of these nonparametric estimators and confidence intervals, one or more choices of bandwidth are needed. Our second command `rdbwselect`, which is used by `rdrobust`, provides a few alternative data-driven bandwidth selectors based on the recent work of Imbens and Kalyanaraman (2012) and Calonico et al. (2012). This command may be used as a stand-alone command in RD applications, although its main purpose is to provide fully data-driven choices to be used by `rdrobust`.

Finally, our third command `rdbinselect`, offers a data-driven optimal length choice of equally-spaced bins, which are useful to approximate the regression function by local sample averages of the outcome variable. This optimal choice is based on an integrated mean-square error expansion of the appropriate estimators, as derived in Cattaneo and Farrell (2012). We discuss how these binned sample-means, and hence the bin-length choice, are used to construct the familiar plots commonly found in RD applications. Employing this optimal choice, this companion command offers a fully automatic way of construct useful RD plots, as discussed in more detail in the upcoming sections.

The rest of this article is organized as follows. Section 2 provides an heuristic review of all the methods implemented in our commands. Sections 3, 4 and 5, describe in detail the syntax of `rdrobust`, `rdbwselect` and `rdbinselect`, respectively. We offer an empirical illustration of our commands in Section 6, where we estimate the so-called incumbency advantage using the data for the U.S. Senate constructed by Cattaneo, Frandsen, and Titiunik (2012). Section 7 describes possible future extensions of our commands.

## 2 Review of Methods

This section offers an overview of the methods implemented in our commands `rdrobust`, `rdbwselect` and `rdbinselect`. To avoid distractions and technicalities, regularity conditions and technical discussions underlying these methods are not presented herein (but may be found in the references given). For recent comprehensive reviews on classical inference approaches in the RD design, including comprehensive lists of empirical examples, see Cook (2008), Imbens and Lemieux (2008), van der Klaauw (2008), Lee and Lemieux (2010), Dinardo and Lee (2011), and references therein. Our discussion focuses on those approaches employing local-polynomial nonparametric estimators with data-driven bandwidth selectors and bias-correction techniques, also following the recent results in Imbens and Kalyanaraman (2012, IK hereafter) and Calonico, Cattaneo, and Titiunik (2012, CCT hereafter).

### 2.1 Setup and Notation

We focus on large-sample inference for the “average treatment effect” in the sharp RD design. We assume  $\{(Y_i, X_i)' : i = 1, 2, \dots, n\}$  is an observed random sample from a large population. For each unit  $i$ , the scalar random variable  $Y_i$  denotes the outcome of interest, while the scalar regressor  $X_i$  is the so-called “running variable” or “score” which determines treatment assignment based on whether it exceeds a known cutoff.

We adopt the potential outcome framework commonly employed in the treatment effects literature (e.g., Imbens and Wooldridge (2009)). Let  $\{(Y_i(0), Y_i(1), X_i)' : i = 1, 2, \dots, n\}$  be a random sample from  $(Y(0), Y(1), X)'$ , where  $Y(1)$  and  $Y(0)$  denote the potential outcomes with and without treatment, respectively, and treatment assignment is determined by the known rule: unit  $i$  is assigned treatment if  $X_i > \bar{x}$  and not assigned treatment if  $X_i < \bar{x}$ , for some known fixed value  $\bar{x} \in \mathbb{R}$ . Therefore, the observed outcome is

$$Y_i = \begin{cases} Y_i(0) & \text{if } X_i < \bar{x} \\ Y_i(1) & \text{if } X_i > \bar{x} \end{cases}.$$

We discuss and implement several data-driven inference procedures for the (sharp) average treatment effect at the threshold, which is given by

$$\tau = \mathbb{E}[Y_i(1) - Y_i(0) | X_i = \bar{x}].$$

This popular estimand in the RD literature is nonparametrically identifiable under mild continuity conditions (Hahn et al. (2001)). Specifically,

$$\tau = \mu_+ - \mu_-,$$

with

$$\mu_+ = \lim_{x \downarrow \bar{x}} \mu(x), \quad \mu_- = \lim_{x \uparrow \bar{x}} \mu(x), \quad \mu(x) = \mathbb{E}[Y_i | X_i = x],$$

where here, and elsewhere in the paper, we drop the evaluation point of functions whenever possible to simplify notation.

Following Hahn et al. (2001) and Porter (2003), a popular estimator of  $\tau$  is constructed using kernel-based local polynomials at either side of the threshold. These regression estimators are particularly well-suited for inference in the RD design because of their good properties at the boundary of the support of the regression function (see Fan and Gijbels (1996) and Cheng et al. (1997) for details). The local polynomial RD estimator of order  $p$  is

$$\hat{\tau}_p(h_n) = \hat{\mu}_{+,p}(h_n) - \hat{\mu}_{-,p}(h_n),$$

where  $\hat{\mu}_{+,p}(h_n)$  and  $\hat{\mu}_{-,p}(h_n)$  denote the intercept (at  $\bar{x}$ ) of a weighted  $p$ -th order polynomial regression for treated and control units, respectively. More precisely,

$$\hat{\mu}_{+,p}(h_n) = \mathbf{e}'_0 \hat{\boldsymbol{\beta}}_{+,p}(h_n) \quad \text{and} \quad \hat{\mu}_{-,p}(h_n) = \mathbf{e}'_0 \hat{\boldsymbol{\beta}}_{-,p}(h_n),$$

with

$$\hat{\boldsymbol{\beta}}_{+,p}(h_n) = \arg \min_{\boldsymbol{\beta} \in \mathbb{R}^{p+1}} \sum_{i=1}^n \mathbf{1}(X_i > \bar{x}) (Y_i - \mathbf{r}_p(X_i - \bar{x})' \boldsymbol{\beta})^2 K_{h_n}(X_i - \bar{x}),$$

$$\hat{\boldsymbol{\beta}}_{-,p}(h_n) = \arg \min_{\boldsymbol{\beta} \in \mathbb{R}^{p+1}} \sum_{i=1}^n \mathbf{1}(X_i < \bar{x}) (Y_i - \mathbf{r}_p(X_i - \bar{x})' \boldsymbol{\beta})^2 K_{h_n}(X_i - \bar{x}),$$

where  $\mathbf{r}_p(x) = (1, x, \dots, x^p)'$ ,  $e_\nu$  is the conformable  $(\nu + 1)$ -th unit vector (e.g.,  $\mathbf{e}_1 = (0, 1, 0)'$  if  $p = 2$ ),  $K_h(u) = K(u/h)/h$  with  $K(\cdot)$  a kernel function,  $h_n$  is a positive bandwidth sequence, and  $\mathbf{1}(\cdot)$  denotes the indicator function.

Under simple regularity conditions, and assuming the bandwidth  $h_n$  vanishes at an appropriate rate, local polynomial estimators are well known to satisfy

$$\hat{\boldsymbol{\beta}}_{+,p}(h_n) \rightarrow_p \boldsymbol{\beta}_{+,p} \quad \text{and} \quad \hat{\boldsymbol{\beta}}_{-,p}(h_n) \rightarrow_p \boldsymbol{\beta}_{-,p},$$

with

$$\boldsymbol{\beta}_{+,p} = \left[ \mu_+, \mu_+^{(1)}, \mu_+^{(2)}/2, \dots, \mu_+^{(p)}/p! \right]', \quad \boldsymbol{\beta}_{-,p} = \left[ \mu_-, \mu_-^{(1)}, \mu_-^{(2)}/2, \dots, \mu_-^{(p)}/p! \right]',$$

$$\mu_+^{(s)} = \lim_{x \downarrow \bar{x}} \frac{\partial^s}{\partial x^s} \mu_+(x), \quad \mu_+(x) = \mathbb{E}[Y(1)|X_i = x],$$

$$\mu_-^{(s)} = \lim_{x \uparrow \bar{x}} \frac{\partial^s}{\partial x^s} \mu_-(x), \quad \mu_-(x) = \mathbb{E}[Y(0)|X_i = x],$$

$s = 1, 2, \dots, p$ , thereby offering a family of consistent estimators of  $\tau$ . Among these possible estimators, the local-linear RD estimator  $\hat{\tau}_1(h_n)$  is perhaps the preferred and most common choice in practice.

## 2.2 Overview of Upcoming Discussion

In the following sections we review some of the salient asymptotic properties of RD treatment-effect estimators  $\hat{\tau}_p(h_n)$ , which are based on local-polynomial nonparametric estimators. Our discussion has the following outline:

(1) We assess some of the main properties of  $\hat{\tau}_p(h_n)$  as point estimators in the first part of Section 2.3. Specifically, we discuss a (conditional) mean-square error (MSE) expansion of  $\hat{\tau}_p(h_n)$ , which highlights their variance and bias properties. We also use this expansion to summarize some bandwidth selection approaches tailored to minimize the leading terms in the asymptotic MSE expansion. In addition, for completeness, we briefly discuss an alternative bandwidth selection approach based on cross-validation (CV).

(2) The rest of Section 2.3 discusses the construction of asymptotically valid confidence intervals based on  $\hat{\tau}_p(h_n)$  for the sharp mean treatment-effect  $\tau$ . In particular, we discuss two distinct approaches: one based on “undersmoothing” and the other based on “bias-correction”.

The first approach, which is arguably the most commonly used in practice, ignores the (potentially large) bias of the estimator, and constructs confidence intervals of the form:

$$\text{CI}_{\alpha,n} = \left[ \hat{\tau}_p(h_n) \pm \Phi_{1-\alpha/2}^{-1} \sqrt{\hat{v}_n} \right],$$

where  $\Phi_{\alpha}^{-1}$  denotes the appropriate quantile of the Gaussian distribution (e.g., 1.96 for  $\alpha = .95$ ) and  $\hat{v}_n$  denotes an appropriate choice of standard-error estimator. This approach is theoretically justified only if the (smoothing) leading bias of the RD estimator is “small”, which from a theoretical perspective requires some form of “undersmoothing”; that is, choosing a “smaller” bandwidth than the MSE-optimal one. In practice, choosing such a bandwidth may be difficult, so researchers typically employ the same bandwidth used to construct the RD estimator  $\hat{\tau}_p(h_n)$ , thereby ignoring the potential effects of the leading bias on the performance of these confidence intervals.

A second approach to construct confidence intervals is to employ bias-correction. This approach is conventional in the nonparametrics literature, although it is not commonly used in empirical work, partially because it is usually regarded as having inferior finite-sample properties. The resulting confidence intervals in this case take the form:

$$\text{CI}_{\alpha,n}^{\text{bc}} = \left[ (\hat{\tau}_p(h_n) - \hat{\mathbf{b}}_n) \pm \Phi_{1-\alpha/2}^{-1} \sqrt{\hat{v}_n} \right],$$

where here the only addition is the bias-estimate  $\hat{\mathbf{b}}_n$ , which is introduced with the explicit goal of removing the potentially large effects of the unknown leading bias of the RD estimator  $\hat{\tau}_p(h_n)$ . This second approach to construct confidence intervals justifies (theoretically) the use of MSE-optimal bandwidth choices when constructing the estimators.

(3) The results in CCT offer alternative confidence intervals based on bias-corrected local polynomials, which take the form:

$$\text{CI}_{\alpha,n}^{\text{rbc}} = \left[ (\hat{\tau}_p(h_n) - \hat{\mathbf{b}}_n) \pm \Phi_{1-\alpha/2}^{-1} \sqrt{\hat{v}_n^{\text{bc}}} \right],$$

where the key difference between the “conventional” confidence intervals ( $\text{CI}^{\text{bc}}$ ) and these alternative confidence intervals ( $\text{CI}^{\text{rbc}}$ ) is the presence of a different standard-error estimator, denoted here by  $\hat{v}_n^{\text{bc}}$ . This new standard-error formula is theoretically

derived by employing an alternative asymptotic approximation to the bias-corrected RD estimator  $\hat{\tau}_p(h_n) - \mathbf{b}_n$ . The resulting confidence intervals enjoy some nice theoretical properties and, as we discuss in more detail below, allow for the use of MSE-optimal bandwidth choices while offering very good finite-sample performance. Section 2.4 offers an heuristic discussion of these results, which are the main motivation for the development of our Stata package.

(4) Finally, in Section 2.5, we discuss a fully automatic (i.e., data-driven) approach to construct the RD plots commonly employed in empirical work. Specifically, our approach exploits a result in Cattaneo and Farrell (2012) to derive an optimal choice of bin length for a partitioning scheme that is used to construct local sample-means useful to approximate the underlying regression functions for control and treatment units. As we further discuss below, these sample means are usually plotted together with estimated global polynomials as a way to summarize the RD design.

## 2.3 Conventional RD Inference

### Point Estimators

Under appropriate regularity conditions, the treatment-effect estimator  $\hat{\tau}_p(h_n)$  admits the following mean-square error (MSE) expansion. Let  $\mathbf{X}_n = (X_1, X_2, \dots, X_n)'$ .

$$\text{MSE}_p(h_n) = \mathbb{E} [(\hat{\tau}_p(h_n) - \tau)^2 | \mathbf{X}_n] \approx h_n^{2(p+1)} \mathbf{B}_{n,p}^2 + \frac{1}{nh_n} \mathbf{V}_{n,p}$$

with

$$\mathbf{B}_{n,p} \rightarrow_p \mathbf{B}_p \quad \text{and} \quad \mathbf{V}_{n,p} \rightarrow_p \mathbf{V}_p,$$

where the exact form of  $\mathbf{B}_{n,p}$  and  $\mathbf{V}_{n,p}$ , and their asymptotic counterparts, may be found at this levels of generality in, for example, CCT. Here  $\mathbf{B}_{n,p}$  and  $\mathbf{V}_{n,p}$  represent, respectively, the leading asymptotic bias and the asymptotic variance of  $\hat{\tau}_p(h_n)$ . It follows that this treatment-effect estimator will be consistent if  $h_n \rightarrow 0$  and  $nh_n \rightarrow \infty$ . Moreover, the point estimator  $\hat{\tau}_p(h_n)$  will be optimal in an asymptotic MSE sense if the bandwidth  $h_n$  is chosen so that

$$h_{p,n}^* = \left( \frac{\mathbf{V}_p}{2(p+1)\mathbf{B}_p^2} \right) n^{-1/(2p+3)},$$

whenever  $\mathbf{B}_p \neq 0$ . As pointed out by IK, this last assumption may be restrictive because  $\mathbf{B}_p \propto \mu_+^{(p+1)} - \mu_-^{(p+1)}$  may be (close to) zero in some applications. (See also IK for a recent review on bandwidth selection in the RD design.)

IK employ this reasoning to provide a data-driven, asymptotically MSE-optimal, RD treatment-effect estimator. Specifically, they propose a more “robust” consistent bandwidth estimator of the form

$$\hat{h}_{\text{IK},n,p} = \left( \frac{\hat{\mathbf{V}}_{\text{IK},p}}{2(p+1)\hat{\mathbf{B}}_{\text{IK},p}^2 + \hat{\mathbf{R}}_{\text{IK},p}} \right) n^{-1/(2p+3)},$$

where the additional (regularization) term  $\hat{R}_p$  is introduced to avoid small denominators in samples of moderate size. Here  $\hat{B}_p$  and  $\hat{V}_p$  (and  $\hat{R}_p$ ) are nonparametric consistent estimators of their respective population counterparts, which require the choice of preliminary bandwidths, generically denoted by  $b_n$  herein. IK provides a simple, direct implementation approach for  $p = 1$  where these estimators ( $\hat{B}_p$ ,  $\hat{V}_p$ ,  $\hat{R}_p$ ) are consistent for their population counterparts, but the preliminary bandwidths used in their construction are not optimally chosen. As a consequence,  $\hat{h}_{\text{IK},n,p}$  may be viewed as a nonparametric first-generation plug-in rule (see, e.g., Wand and Jones (1995)), sometimes denoted by DPI-1.

Although not the main focus of their paper, but motivated by the work of IK, CCT propose an alternative, second-generation plug-in bandwidth selection approach. Specifically, CCT propose a second-order direct plug-in rule (DPI-2)

$$\hat{h}_{\text{CCT},n,p} = \left( \frac{\hat{V}_{\text{CCT},p}}{2(p+1)\hat{B}_{\text{CCT},p}^2 + \hat{R}_{\text{CCT},p}} \right) n^{-1/(2p+3)},$$

where this alternative bandwidth estimator has two distinct features relative to  $\hat{h}_{\text{IK},n,p}$ . First, not only the estimators  $\hat{V}_{\text{CCT},p}$  and  $\hat{B}_{\text{CCT},p}$  (and  $\hat{R}_{\text{CCT},p}$ ) are consistent for their population counterparts, but also the preliminary bandwidths used in their constructions are also consistent estimators of the corresponding population MSE-optimal bandwidths. In this sense,  $\hat{h}_{\text{CCT},n,p}$  is a DPI-2.

Second, motivated by finite-sample performance considerations, CCT construct an alternative estimator of  $V_p$  (denoted by  $\hat{V}_{\text{CCT},p}$  above), which does not require an additional choice of bandwidth for its construction. It relies instead on a fixed-matches nearest-neighbor-based “estimate” of the residuals, following the work of Abadie and Imbens (2006). Below this construction, as well as other more traditional approaches, are further discussed because the term  $V_p$  plays a crucial role when constructing confidence intervals for  $\tau$ .

The main bandwidth  $h_{n,p}$  may be chosen in other ways. A popular alternative is to employ cross-validation, as done for example by Ludwig and Miller (2007). As discussed in IK, one such bandwidth selection approach may be described as follows:

$$\hat{h}_{\text{CV},n} = \arg \min_{h>0} \text{CV}_\delta(h), \quad \text{CV}_\delta(h) = \sum_{i=1}^n \mathbf{1}(X_{-,[\delta]} \leq X_i \leq X_{+,[\delta]}) (Y_i - \hat{\mu}_p(X_i; h))^2,$$

where

$$\hat{\mu}(x; h) = \begin{cases} \mathbf{e}'_0 \hat{\beta}_{+,p}(x, h) & \text{if } x > \bar{x} \\ \mathbf{e}'_0 \hat{\beta}_{-,p}(x, h) & \text{if } x < \bar{x} \end{cases},$$

$$\hat{\beta}_{+,p}(x, h) = \arg \min_{\beta \in \mathbb{R}^{p+1}} \sum_{i=1}^n \mathbf{1}(X_i > x) (Y_i - \mathbf{r}_p(X_i - x)' \beta)^2 K_h(X_i - x),$$

$$\hat{\beta}_{-,p}(x, h) = \arg \min_{\beta \in \mathbb{R}^{p+1}} \sum_{i=1}^n \mathbf{1}(X_i < x) (Y_i - \mathbf{r}_p(X_i - x)' \beta)^2 K_h(X_i - x),$$

and, for  $\delta \in (0, 1)$ ,  $X_{-,[\delta]}$  and  $X_{+,[\delta]}$  denote the  $\delta$ -th quantile of  $\{X_i : X_i < \bar{x}\}$  and  $\{X_i : X_i > \bar{x}\}$ , respectively. Our bandwidth selection command also implements this approach for completeness.

To summarize, the results discussed so far justify three data-driven RD treatment-effect point estimators:

$$\hat{\tau}_p(\hat{h}_{\text{IK},n,p}), \quad \hat{\tau}_p(\hat{h}_{\text{CCT},n,p}), \quad \hat{\tau}_p(\hat{h}_{\text{CV},n,p}).$$

Under appropriate conditions, these estimators may be interpreted as consistent and (asymptotically) MSE-optimal point estimators of  $\tau$ .

### Confidence Intervals: Asymptotic Distribution

Conventional confidence intervals accompanying the point estimators discussed above rely on the following distributional approximation. Under appropriate regularity conditions and rate-restrictions on the bandwidth sequence  $h_n \rightarrow 0$ ,

$$\sqrt{nh_n} \left( \hat{\tau}_p(h_n) - \tau - h_n^{(p+1)} \mathbf{B}_{n,p} \right) \rightarrow_d \mathcal{N}(0, \mathbf{V}_p), \quad \mathbf{V}_p \propto \frac{\sigma_+^2 + \sigma_-^2}{f},$$

where  $\sigma_+^2 = \lim_{x \rightarrow 0^+} \sigma^2(x)$  and  $\sigma_-^2 = \lim_{x \rightarrow 0^-} \sigma^2(x)$  with  $\sigma^2(x) = \mathbb{V}[Y_i | X_i = x]$ , and  $f(x)$  is the density of  $X$ . Therefore, an infeasible asymptotic  $100\alpha$ -percent confidence interval for  $\tau$  is given by

$$\text{CI}_\alpha^*(h_n) = \left[ \left( \hat{\tau}_p(h_n) - h_n^{p+1} \mathbf{B}_{n,p} \right) \pm \Phi_{1-\alpha/2}^{-1} \sqrt{\frac{\mathbf{V}_p}{nh_n}} \right],$$

where  $\Phi_\alpha^{-1}$  denotes the  $\alpha$ -quantile of the Gaussian distribution. To implement this confidence intervals in practice, we need to handle the leading bias ( $\mathbf{B}_{n,p}$ ) and the variance ( $\mathbf{V}_p$ ) of the RD estimator because they involve unknown quantities. We discuss these and related practical issues in the following subsections.

### Confidence Intervals: Standard-Error Estimators

The asymptotic variance is handled by replacing  $\mathbf{V}_p$  with a consistent estimator. A natural approach employs the conditional (on  $\mathbf{X}_n$ ) variance of  $\hat{\tau}_p(h_n)$  as a starting point because, as mentioned above,  $\mathbf{V}_{n,p} \rightarrow_p \mathbf{V}_p$ . Here we have

$$\mathbf{V}_{n,p} = \mathbb{V}[\hat{\tau}_p(h_n) | \mathbf{X}_n] = \mathbf{V}_{+,n,p} + \mathbf{V}_{-,n,p},$$

with

$$\mathbf{V}_{+,n,p} = \mathbf{e}_0' \Gamma_{+,n,p}^{-1} \mathbf{X}_{+,n,p} \mathbf{W}_{+,n,p} \Sigma \mathbf{W}_{+,n,p} \mathbf{X}_{+,n,p} \Gamma_{+,n,p}^{-1} \mathbf{e}_0,$$

$$\mathbf{V}_{-,n,p} = \mathbf{e}_0' \Gamma_{-,n,p}^{-1} \mathbf{X}_{-,n,p} \mathbf{W}_{-,n,p} \Sigma \mathbf{W}_{-,n,p} \mathbf{X}_{-,n,p} \Gamma_{-,n,p}^{-1} \mathbf{e}_0,$$

where the exact form of these matrices are given in, for example, CCT. Importantly, the only matrix including unknown quantities is

$$\Sigma = \begin{bmatrix} \sigma^2(X_1) & 0 & \cdots & 0 \\ 0 & \sigma^2(X_2) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \sigma^2(X_n) \end{bmatrix} = \mathbb{E}[\boldsymbol{\varepsilon}\boldsymbol{\varepsilon}'|\mathbf{X}_n],$$

where  $\boldsymbol{\varepsilon} = (\varepsilon_1, \varepsilon_2, \dots, \varepsilon_n)'$  with  $\varepsilon_i = Y_i - \mu(X_i)$ .

The “sandwich” structure of  $\mathbf{V}_{n,p}$  arises naturally from the weighted least-squares form of local polynomials, resembling the usual heteroskedasticity-robust standard-error formula in linear regression models. As a consequence, and just like in linear models, implementing these standard-errors only requires an estimator of  $\Sigma$ , which in turn reduces the problem to plugging-in an estimator of  $\sigma^2(X_i) = \mathbb{E}[\varepsilon_i^2|X_i]$ , for control and treatment units separately. In this article we consider two approaches to construct such estimators: “plug-in estimated errors” and “fixed-matches errors”. Both approaches construct an estimator of  $\mathbf{V}_{n,p}$  by removing the conditional expectation in  $\Sigma$  and replacing  $\varepsilon_i$  by some “estimator” of it.

*Plug-in Estimated Errors.* This approach follows the standard linear models logic underlying local polynomial estimators, and therefore replaces  $\varepsilon_i$  by

$$\tilde{\varepsilon}_{+,i} = Y_i - \hat{\mu}_{+,p}(X_i; c_n) \quad \text{and} \quad \tilde{\varepsilon}_{-,i} = Y_i - \hat{\mu}_{-,p}(X_i; c_n),$$

respectively, for treated and control units as appropriate. Here the bandwidth employed is denoted by  $c_n$ , which in practice is usually chosen to be  $c_n = h_n$ , even though this choice may not be theoretically justified and could lead to poor finite-sample performance of the estimators. The resulting standard-error formula becomes the familiar Huber-Eicker-White standard-error estimator, which is robust to heteroskedasticity of unknown form. We denote this estimator by

$$\check{\mathbf{V}}_{n,p} = \check{\mathbf{V}}_{+,n,p} + \check{\mathbf{V}}_{-,n,p},$$

where  $\check{\mathbf{V}}_{+,n,p}$  and  $\check{\mathbf{V}}_{-,n,p}$  employ, respectively,  $\tilde{\varepsilon}_{+,i}$  and  $\tilde{\varepsilon}_{-,i}$  in their construction.

*Fixed-matches Errors.* The previous construction is intuitive and easy-to-implement, but requires an additional choice of bandwidth to construct the estimates of the residuals. Employing the same bandwidth choice used to construct the RD treatment-effect estimator may not lead to a standard-error estimator with good finite-sample properties. As an alternative, CCT propose a standard-error estimator employing a different construction for the residuals, motivated by the recent work of Abadie and Imbens (2006). This estimator is constructed using a simple fixed-matches “estimator” for the residuals, denoted by  $\hat{\varepsilon}_{+,i}$  and  $\hat{\varepsilon}_{-,i}$ , which are unbiased but inconsistent. Nonetheless, the resulting standard-error estimators are shown to be consistent under appropriate

regularity conditions. See CCT for details, which are not reproduced here to conserve space. We denote this estimator by

$$\hat{V}_{n,p} = \hat{V}_{+,n,p} + \hat{V}_{-,n,p},$$

where  $\hat{V}_{+,n,p}$  and  $\hat{V}_{-,n,p}$  employ, respectively, the fixed-matches estimators  $\hat{\varepsilon}_{+,i}$  and  $\hat{\varepsilon}_{-,i}$  in their construction.

In summary, two alternative standard-error estimators are (i) the plug-in estimated residuals estimator  $\check{V}_{n,p}$  and (ii) the fixed-matches estimator  $\hat{V}_{n,p}$ , both satisfying

$$\check{V}_{n,p} \rightarrow_p V_p \quad \text{and} \quad \hat{V}_{n,p} \rightarrow_p V_p.$$

For implementation of either of these estimators we replace the bandwidth  $h_n$  by the same estimated bandwidth employed in the treatment-effect estimator  $\hat{\tau}_p(h_n)$  as appropriate.

### **Confidence Intervals: Asymptotic Bias**

There are two main approaches to handle the leading bias ( $B_{n,p}$ ) present in the infeasible confidence intervals  $CI_\alpha^*(h_n)$ : “undersmoothing” (alternatively, assuming the bias is “small”), or “bias correction”. We briefly discuss each of these ideas here.

*Undersmoothing.* The first approach is to “undersmooth” the estimator, that is, to choose a “small” enough bandwidth so the bias is negligible. Theoretically, this approach simply requires selecting a bandwidth sequence  $h_n \rightarrow 0$  such that

$$\sqrt{nh_n} (\hat{\tau}_p(h_n) - \tau - h_n^{p+1} B_{n,p}) = \sqrt{nh_n} (\hat{\tau}_p(h_n) - \tau) + o_p(1) \rightarrow_d \mathcal{N}(0, V_p),$$

which is justified whenever the bandwidth  $h_n$  vanishes fast enough. In practice, however, this procedure may be difficult to implement because most bandwidth selectors, such as  $h_{p,n}^*$ , will not satisfy the conditions required for undersmoothing. This fact implies that most empirical bandwidth selectors could in principle lead to a non-negligible leading bias in the distributional approximation, which in turn will bias the associated confidence intervals. Simulation evidence highlighting this potential drawback of the undersmoothing/small-bias approach is provided in CCT.

Nonetheless, in applications it is common to simply ignore the leading bias, proceeding as if  $B_{n,p} \approx 0$ . This approach is justified by either assuming the bias is “small”, or by shrinking the bandwidth choice by some ad-hoc factor (i.e., undersmoothing). We give the exact formula of the resulting confidence intervals justified by this approach in the next subsection.

*Bias Correction.* The second approach to deal with the leading bias in the distributional approximation is to bias-correct the estimator, that is, to construct an estimator of  $B_{n,p}$ , which is then subtracted off to eliminate this leading bias.

A simple way to implement this idea is to use a higher-order local polynomial to estimate the unknown derivatives present in the leading bias term; recall that  $\mathbf{B}_p \propto \mu_+^{(p+1)} - \mu_-^{(p+1)}$ . For example,  $\mu_+^{(p+1)}$  can be estimated by using a  $q$ -th order local polynomial ( $q \geq p+1$ ) with pilot bandwidth  $b_n$ , leading to the estimator  $\hat{\mu}_+^{(p+1)} = e'_{p+1} \hat{\mathbf{B}}_{+,q}(b_n)$ . This is the approach we employed herein to construct a bias-estimate  $\hat{\mathbf{B}}_{n,p,q}$ , which depends on a preliminary bandwidth choice  $b_n$ . The resulting bias-corrected estimator is

$$\hat{\tau}_{p,q}^{\text{bc}}(h_n, b_n) = \hat{\tau}_p(h_n) - h_n^{p+1} \hat{\mathbf{B}}_{n,p,q}.$$

Using this bias-corrected estimator, and imposing appropriate regularity conditions and bandwidth restrictions, we obtain:

$$\sqrt{nh_n} (\hat{\tau}_{p,q}^{\text{bc}}(h_n, b_n) - \tau) = \underbrace{\sqrt{nh_n} (\hat{\tau}_p(h_n) - \tau - h_n^{p+1} \mathbf{B}_{n,p})}_{\rightarrow_d \mathcal{N}(0, \mathbf{V}_p)} - \underbrace{\sqrt{nh_n} h_n^{p+1} (\hat{\mathbf{B}}_{n,p,q} - \mathbf{B}_{n,p})}_{\rightarrow_p 0},$$

which is valid if

$$\frac{h_n}{b_n} \rightarrow 0.$$

This result immediately justify bias-corrected confidence intervals, where the unknown bias in  $\text{CI}_\alpha^*(h_n)$  is replaced by the bias-estimate  $\hat{\mathbf{B}}_{n,p,q}$ . The exact formula of the resulting confidence intervals is also given in the upcoming subsection.

To implement this approach a choice of pilot bandwidth  $b_n$  is now also required. As discussed above, bandwidth choices may be constructed using asymptotic MSE expansions for the appropriate estimators. This is the approach followed by CCT, who propose the following pilot bandwidth optimal choice: a MSE-optimal choice of  $b_n$  for the bias-correction estimator  $\hat{\mathbf{B}}_{n,p,q}$  is given by

$$b_{q,n}^* = \left( \frac{(2p+3) \mathbf{V}_{q,p}}{2(q-p) \mathbf{B}_{q,p}^2} \right) n^{-1/(2q+3)},$$

where  $\mathbf{V}_{q,p}$  and  $\mathbf{B}_{q,p}$  are the corresponding variance and leading bias terms arising from the MSE expansion used. (Note this choice is not necessarily optimal for  $\hat{\tau}_{p,q}^{\text{bc}}(h_n, b_n)$ .) CCT discuss a relative simple implementation procedure of  $b_{q,n}^*$ , leading to the data-driven estimator:

$$\hat{b}_{\text{CCT},n,q} = \left( \frac{(2p+3) \hat{\mathbf{V}}_{\text{CCT},q,p}}{2(q-p) \hat{\mathbf{B}}_{\text{CCT},q,p}^2 + \hat{\mathbf{R}}_{\text{CCT},q,p}} \right) n^{-1/(2p+3)},$$

where the exact form of the estimators  $\hat{\mathbf{V}}_{\text{CCT},q,p}$ ,  $\hat{\mathbf{B}}_{\text{CCT},q,p}$  and  $\hat{\mathbf{R}}_{\text{CCT},q,p}$  are given in CCT but not reproduced here to conserve space. In our implementation we consider this pilot bandwidth estimator as well as similar estimators constructed following the underlying logic in IK. The latter choice is denoted by  $\hat{b}_{\text{IK},n,q}$ .

### Confidence Intervals: Summary of Classical Approaches

The results discussed so far suggest the following data-driven RD treatment-effect confidence intervals:

- *Undersmoothing / Small-Bias:*

- *Plug-in estimated errors:*  $\check{C}I_\alpha(\hat{h}_{IK,n})$ ,  $\check{C}I_\alpha(\hat{h}_{CCT,n})$  and  $\check{C}I_\alpha(\hat{h}_{CV,n})$ , where

$$\check{C}I_\alpha(h_n) = \left[ \hat{\tau}_p(h_n) \pm \Phi_{1-\alpha/2}^{-1} \sqrt{\frac{\check{V}_p}{nh_n}} \right].$$

- *Fixed-matches errors:*  $\hat{C}I_\alpha(\hat{h}_{IK,n})$ ,  $\hat{C}I_\alpha(\hat{h}_{CCT,n})$  and  $\hat{C}I_\alpha(\hat{h}_{CV,n})$ , where

$$\hat{C}I_\alpha(h_n) = \left[ \hat{\tau}_p(h_n) \pm \Phi_{1-\alpha/2}^{-1} \sqrt{\frac{\hat{V}_p}{nh_n}} \right].$$

- *Bias-Correction:*

- *Plug-in estimated errors:*  $\check{C}I_\alpha^{\text{bc}}(\hat{h}_n, \hat{b}_n)$ , where

$$\check{C}I_\alpha^{\text{bc}}(h_n, b_n) = \left[ \left( \hat{\tau}_p(h_n) - h_n^{p+1} \hat{B}_{n,p,q} \right) \pm \Phi_{1-\alpha/2}^{-1} \sqrt{\frac{\check{V}_p}{nh_n}} \right].$$

- *Fixed-matches errors:*  $\hat{C}I_\alpha^{\text{bc}}(\hat{h}_n, \hat{b}_n)$ , where

$$\hat{C}I_\alpha^{\text{bc}}(h_n, b_n) = \left[ \left( \hat{\tau}_p(h_n) - h_n^{p+1} \hat{B}_{n,p,q} \right) \pm \Phi_{1-\alpha/2}^{-1} \sqrt{\frac{\hat{V}_p}{nh_n}} \right].$$

Here  $\hat{h}_n \in \{\hat{h}_{IK,n}, \hat{h}_{CCT,n}, \hat{h}_{CV,n}\}$  and  $\hat{b}_n \in \{\hat{b}_{IK,n}, \hat{b}_{CCT,n}\}$ , for example.

## 2.4 Robust RD Inference

The classical confidence intervals discussed above may have some unappealing properties that could seriously affect their performance in empirical work. The confidence intervals  $\check{C}I_\alpha(h_n)$  and  $\hat{C}I_\alpha(h_n)$  require undersmoothing (or, alternatively, a “small” bias), leading to potentially large coverage distortions otherwise. The bias-corrected confidence intervals  $\check{C}I_\alpha^{\text{bc}}(h_n, b_n)$  and  $\hat{C}I_\alpha^{\text{bc}}(h_n, b_n)$ , while theoretically justified, are usually regarded as having poor performance in empirical settings, also leading to potentially large coverage distortions in applications. Monte Carlo evidence showing some of these potential pitfalls is given in CCT.

Alternative, more robust confidence intervals may be constructed using bias-corrected RD treatment-effect estimators as a starting point. Intuitively, these estimators do not perform well in finite-samples because the bias-estimate introduces additional variability to  $\hat{\tau}_{p,q}^{\text{bc}}(h_n, b_n) = \hat{\tau}_p(h_n) - h_n^{p+1} \hat{\mathbf{B}}_{n,p,q}$ , which is not accounted for when forming the associated confidence intervals (e.g.,  $\check{\mathbf{C}}\mathbf{I}_\alpha^{\text{bc}}(h_n, b_n)$  and  $\hat{\mathbf{C}}\mathbf{I}_\alpha^{\text{bc}}(h_n, b_n)$ ).

Following this reasoning, CCT propose an alternative asymptotic approximation for  $\hat{\tau}_{p,q}^{\text{bc}}(h_n, b_n)$  which is heuristically given by the observation that

$$\sqrt{nh_n} (\hat{\tau}_{p,q}^{\text{bc}}(h_n, b_n) - \tau) = \underbrace{\sqrt{nh_n} (\hat{\tau}_p(h_n) - \tau - h_n^{p+1} \mathbf{B}_{n,p})}_{\rightarrow_d \mathcal{N}(0, \mathbf{V}_p)} - \underbrace{\sqrt{nh_n} h_n^{p+1} (\hat{\mathbf{B}}_{n,p,q} - \mathbf{B}_{n,p})}_{\rightarrow_d \mathcal{N}(0, \mathbf{V}_{q,p}(c))}$$

provided that (appropriate regularity conditions hold and)

$$\frac{h_n}{b_n} \rightarrow \rho \in (0, \infty).$$

Here  $\mathbf{V}_{q,p}(\rho)$  could be interpreted as the contribution of the bias-correction to the variability of the bias-corrected estimator. (It can be shown that  $\mathbf{V}_{q,p}(0) = 0$ .)

Under weaker conditions than those typically imposed in the results summarized in the previous subsections, CCT show that

$$\sqrt{nh_n} (\hat{\tau}_{p,q}^{\text{bc}}(h_n, b_n) - \tau) \rightarrow_d \mathcal{N}(0, \mathbf{V}_{p,q}^{\text{bc}}(\rho)),$$

where  $\mathbf{V}_{p,q}^{\text{bc}}(\rho)$  is the asymptotic variance for the bias-corrected estimator, which is different than the usual one. Indeed, it can be shown that if  $\rho \downarrow 0$  then  $\mathbf{V}_{p,q}^{\text{bc}}(\rho) \rightarrow \mathbf{V}_p$ , but in general  $\mathbf{V}_{p,q}^{\text{bc}}(\rho) > \mathbf{V}_p$ , under appropriate conditions.

More generally, it can be shown that under appropriate conditions,

$$\frac{\hat{\tau}_{p,q}^{\text{bc}}(h_n, b_n) - \tau}{\sqrt{\mathbf{V}_{n,q,p}^{\text{bc}}(h_n, b_n)}} \rightarrow_d \mathcal{N}(0, 1),$$

where the exact formula for  $\mathbf{V}_{n,q,p}^{\text{bc}}(h_n, b_n)$  is given in CCT. Intuitively, this variance formula is constructed to account for both the variability of the original RD treatment-effect estimator ( $\hat{\tau}_p(h_n)$ ) and the variability of the bias-correction term ( $\hat{\mathbf{B}}_{n,p,q}$ ) in the distributional approximation of the studentized statistic.

These more general distributional approximation leads to the following data-driven robust confidence intervals:

- *Robust Bias-Correction:*

– *Plug-in estimated errors:*  $\check{\mathbf{C}}\mathbf{I}_\alpha^{\text{rbc}}(\hat{h}_n, \hat{b}_n)$ , where

$$\hat{\mathbf{C}}\mathbf{I}_\alpha^{\text{rbc}}(h_n, b_n) = \left[ \left( \hat{\tau}_p(h_n) - h_n^{p+1} \hat{\mathbf{B}}_{n,p,q} \right) \pm \Phi_{1-\alpha/2}^{-1} \sqrt{\check{\mathbf{V}}_{n,q,p}^{\text{bc}}(h_n, b_n)} \right].$$

- Fixed-matches errors:  $\hat{C}l_\alpha^{\text{bc}}(\hat{h}_n, \hat{b}_n)$ , where

$$\hat{C}l_\alpha^{\text{rbc}}(h_n, b_n) = \left[ \left( \hat{\tau}_p(h_n) - h_n^{p+1} \hat{B}_{n,p,q} \right) \pm \Phi_{1-\alpha/2}^{-1} \sqrt{\hat{V}_{n,q,p}^{\text{bc}}(h_n, b_n)} \right].$$

As above, for example,  $\hat{h}_n \in \{\hat{h}_{\text{IK},n}, \hat{h}_{\text{CCT},n}, \hat{h}_{\text{CV},n}\}$  and  $\hat{b}_n \in \{\hat{b}_{\text{IK},n}, \hat{b}_{\text{CCT},n}, \hat{b}_{\text{CV},n}\}$ . In addition, the exact formulas for the constructions  $\check{V}_{n,q,p}^{\text{bc}}(h_n, b_n)$  and  $\hat{V}_{n,q,p}^{\text{bc}}(h_n, b_n)$  are given in CCT.

## 2.5 Procedures Implemented for RD Inference

The commands `rdbwselect` and `rdrobust` implement the following procedures.

- `rdbwselect` implements three bandwidth selectors for  $h_{n,p}^*$ :
  - $\hat{h}_{\text{IK},n,p}$ : IK implementation for  $p$ -th order local polynomial estimator.
  - $\hat{h}_{\text{CCT},n,p}$ : CCT implementation for  $p$ -th order local polynomial estimator. **This is the default in `rdbwselect`.**
  - $\hat{h}_{\text{CV},n,p}$ : CV implementation for  $p$ -th order local polynomial estimator.
- `rdbwselect` implements two bandwidth selectors for  $b_{n,q}^*$ :  $\hat{b}_{\text{IK},n,q}$  and  $\hat{b}_{\text{CCT},n,q}$ .
  - $\hat{b}_{\text{IK},n,p}$ : IK-analogue implementation for  $p$ -th order local polynomial estimator.
  - $\hat{b}_{\text{CCT},n,p}$ : CCT implementation for  $p$ -th order local polynomial estimator. **This is the default in `rdbwselect`.**
- `rdrobust` implements two point estimators for  $\tau$ :
  - $\hat{\tau}_p(h_n)$ :  $p$ -th order local polynomial estimator. **This is the default in `rdrobust`.**
  - $\hat{\tau}_{p,q}^{\text{bc}}(h_n, b_n)$ :  $p$ -th order local polynomial estimator with  $q$ -th order local polynomial bias-correction.
- `rdrobust` implements six confidence intervals for  $\tau$ :
  - $\check{C}l_\alpha(h_n)$ : no-bias-correction with plug-in estimated errors for standard-errors formula.

- $\hat{CI}_\alpha(h_n)$ : no-bias-correction with fixed-matches errors for standard-errors formula.
- $\check{CI}_\alpha^{bc}(h_n, b_n)$ : bias-correction with plug-in estimated errors for standard-errors formula.
- $\hat{CI}_\alpha^{bc}(h_n, b_n)$ : bias-correction with fixed-matches errors for standard-errors formula.
- $\check{CI}_\alpha^{rbc}(h_n, b_n)$ : bias-correction with plug-in estimated errors for robust standard-errors formula.
- $\hat{CI}_\alpha^{rbc}(h_n, b_n)$ : bias-correction with fixed-matches errors for robust standard-errors formula. **This is the default in `rdrobust`.**

All the details on the syntax of `rdrobust` and `rdbwselect` are given in Section 3 and Section 4, respectively. Section 6 includes a complete empirical example that illustrate these methods and commands using real data. Further details on implementation of the bandwidth selectors are given in IK and CCT.

## 2.6 Data-Driven RD Plots

The main aspects of the sharp RD design may be summarized in an easy-to-interpret figure, showing how an estimated regression function behaves for control ( $X_i < \bar{x}$ ) and treated ( $X_i \geq \bar{x}$ ) units relative to some summary of the actual data. This common RD plot gives an idea of overall fit, while also exhibits graphically the sharp RD estimate. In most empirical applications, this figure is constructed using “dots” for local sample means over non-overlapping bins partitioning a restricted support of  $X$  together with two estimated smooth “global” polynomial regression curves. The binned means are usually included to capture the behavior of the “cloud of points”, while the two global-polynomial estimates for control and treatment units are meant to show a flexible approximation of  $\mu_-(x)$  and  $\mu_+(x)$ . Figure 1 gives an example of this kind of plot using the data of Lee (2008).

This exploratory graphical approach employs two main ingredients. First, polynomial regression curves are estimated to flexibly approximate the population conditional mean functions for control and treated units, usually over a large but restricted support of the running variable. Formally, these estimates are a  $p$ -th order “global” polynomials given by

$$\hat{\mu}_{-,p,1}(x) = \mathbf{r}_p(x)' \hat{\gamma}_{-,p,1}(x_l) \quad \text{and} \quad \hat{\mu}_{+,p,1}(x) = \mathbf{r}_p(x)' \hat{\gamma}_{+,p,1}(x_u),$$

with

$$\hat{\gamma}_{-,p,k}(x_l) = \arg \min_{\gamma \in \mathbb{R}^{p+1}} \sum_{i=1}^n \mathbf{1}(x_l < X_i < \bar{x}) (Y_i^k - \mathbf{r}_p(X_i)' \gamma)^2,$$

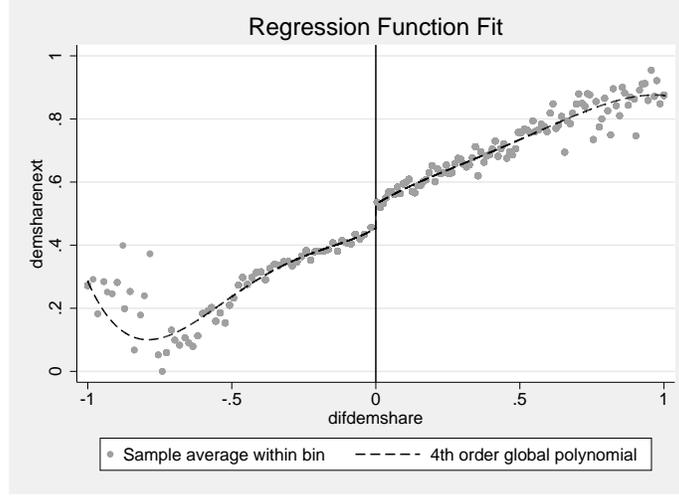


Figure 1: Binned sample-means and global-polynomials fit (dataset of Lee (2008)).

$$\hat{\gamma}_{+,p,k}(x_u) = \arg \min_{\gamma \in \mathbb{R}^{p+1}} \sum_{i=1}^n \mathbb{1}(\bar{x} < X_i < x_u) (Y_i^k - r_p(X_i)' \gamma)^2,$$

where  $\mathbf{r}_p(x) = (1, x, \dots, x^p)'$  and  $k = 1, 2$ . Here  $x_l$  and  $x_u$  are the lower and upper limits on the support of the running variable, and satisfy  $x_l < \bar{x} < x_u$ . In words,  $\hat{\mu}_{-,p,1}(x)$  and  $\hat{\mu}_{+,p,1}(x)$  are  $p$ -th order global polynomials over the supports  $(x_l, \bar{x})$  and  $(\bar{x}, x_u)$  of control and treated units. This ingredient of the figure is easily implementable in practice, common choices being  $p = 4$  and  $p = 5$ .

The second part of the figure requires constructing sample means over nonoverlapping regions of the support of the running variable  $X$ , for control and treatment units separately. These sample means are meant to provide an approximation of the population regression functions as well, but they also give a sense of dispersion of the data around them. To implement these estimators, we construct two evenly-spaced partitions for control units and treatment units separately:

$$\mathcal{P}_{-,n} = \bigcup_{j=1}^{J_{-,n}} \mathcal{P}_{-,j,n} = (x_l, \bar{x}) \quad \text{and} \quad \mathcal{P}_{+,n} = \bigcup_{j=1}^{J_{+,n}} \mathcal{P}_{+,j,n} = (\bar{x}, x_u),$$

where

$$\mathcal{P}_{-,j,n} = \left( \bar{x} - \frac{j(\bar{x} - x_l)}{J_{-,n}}, \bar{x} - \frac{(j-1)(\bar{x} - x_l)}{J_{-,n}} \right), \quad \text{for } j = 1, 2, \dots, J_{-,n},$$

$$\mathcal{P}_{+,j,n} = \left( \bar{x} + \frac{(j-1)(x_u - \bar{x})}{J_{+,n}}, \bar{x} + \frac{j(x_u - \bar{x})}{J_{+,n}} \right), \quad \text{for } j = 1, 2, \dots, J_{+,n}.$$

Thus,  $\mathcal{P}_n$  forms a (disjoint) equally-spaced partition of  $(x_l, x_u)$  centered at  $\bar{x}$ , which is assumed to become finer as the sample size grows (i.e.,  $J_{-,n} \rightarrow \infty$  and  $J_{+,n} \rightarrow \infty$ ). To implement this binning in practice, it is necessary to select the common length of the bins, denoted here by  $(\bar{x} - x_l)/J_{-,n}$  and  $(x_u - \bar{x})/J_{+,n}$ .

The resulting binned means may be mathematically represented by

$$\begin{aligned}\check{\mu}_-(x; J_{-,n}) &= \sum_{j=1}^{J_{-,n}} \mathbb{1}_{\mathcal{P}_{-,j,n}}(x) \bar{Y}_{-,j}, & \bar{Y}_{-,j} &= \frac{1}{N_{-,j}} \sum_{i=1}^n \mathbb{1}_{\mathcal{P}_{-,j,n}}(X_i) Y_i, \\ \check{\mu}_+(x; J_{+,n}) &= \sum_{j=1}^{J_{+,n}} \mathbb{1}_{\mathcal{P}_{+,j,n}}(x) \bar{Y}_{+,j}, & \bar{Y}_{+,j} &= \frac{1}{N_{+,j}} \sum_{i=1}^n \mathbb{1}_{\mathcal{P}_{+,j,n}}(X_i) Y_i,\end{aligned}$$

with

$$N_{-,j} = \sum_{i=1}^n \mathbb{1}_{\mathcal{P}_{-,j,n}}(X_i), \quad N_{+,j} = \sum_{i=1}^n \mathbb{1}_{\mathcal{P}_{+,j,n}}(X_i), \quad \mathbb{1}_P(x) = \mathbb{1}(x \in P).$$

It follows that these estimators are a simple version of the nonparametric partitioning estimators; see, e.g., Cattaneo and Farrell (2012) and references therein. As a consequence, the asymptotic (integrated) MSE expansion given in (Cattaneo and Farrell 2012, Theorem 3) may be used to derive an optimal (data-driven) choice for  $J_{-,n}$  and  $J_{+,n}$ .

Specifically, under appropriate regularity conditions, we obtain the following expansions:

$$\begin{aligned}\int_{x_l}^{\bar{x}} \mathbb{E}[(\check{\mu}_-(x; J_{-,n}) - \mu_-(x))^2 \mid \mathbf{X}_n] w(x) dx &\approx \frac{J_{-,n}}{n} \mathcal{V}_- + \frac{1}{J_{-,n}^2} \mathcal{B}_-, \\ \int_{\bar{x}}^{x_u} \mathbb{E}[(\check{\mu}_+(x; J_{+,n}) - \mu_+(x))^2 \mid \mathbf{X}_n] w(x) dx &\approx \frac{J_{+,n}}{n} \mathcal{V}_+ + \frac{1}{J_{+,n}^2} \mathcal{B}_+, \end{aligned}$$

with

$$\begin{aligned}\mathcal{V}_- &= \frac{1}{\bar{x} - x_l} \int_{x_l}^{\bar{x}} \frac{\sigma_-^2(x)}{f(x)} w(x) dx, & \mathcal{B}_- &= \frac{(\bar{x} - x_l)^2}{12} \int_{x_l}^{\bar{x}} \left(\mu_-^{(1)}(x)\right)^2 w(x) dx, \\ \mathcal{V}_+ &= \frac{1}{x_u - \bar{x}} \int_{\bar{x}}^{x_u} \frac{\sigma_+^2(x)}{f(x)} w(x) dx, & \mathcal{B}_+ &= \frac{(x_u - \bar{x})^2}{12} \int_{\bar{x}}^{x_u} \left(\mu_+^{(1)}(x)\right)^2 w(x) dx,\end{aligned}$$

and where  $w(x)$  is a weighting function. Thus, the optimal choice of bin lengths for these problems are given by

$$\begin{aligned}J_{-,n}^* &= \left\langle (\mathcal{E}_- n)^{1/3} \right\rangle, & \mathcal{E}_- &= \frac{2 \mathcal{B}_-}{\mathcal{V}_-}, \\ J_{+,n}^* &= \left\langle (\mathcal{E}_+ n)^{1/3} \right\rangle, & \mathcal{E}_+ &= \frac{2 \mathcal{B}_+}{\mathcal{V}_+},\end{aligned}$$

where  $\langle \cdot \rangle$  denotes the nearest integer. A feasible plug-in rule can be easily constructed by using preliminary estimators for the unknown objects in  $\mathcal{E}_+$ , as we discuss next.

### Implementation

We choose  $w(x) = f(x)$ , and propose simple easy-to-implement rule-of-thumb estimates for  $\mathcal{C}_-$  and  $\mathcal{C}_+$ .

*Estimated Bias.* To approximate the leading bias constants, we employ the global polynomials already constructed above. Specifically, we set

$$\hat{\mu}_{-,p,1}^{(1)}(x) = \frac{d}{dx} \hat{\mu}_{-,p,1}(x) = \left( \frac{d}{dx} \mathbf{r}_p(x) \right)' \hat{\gamma}_{-,p,1}(x_l),$$

$$\hat{\mu}_{+,p,1}^{(1)}(x) = \frac{d}{dx} \hat{\mu}_{+,p,1}(x) = \left( \frac{d}{dx} \mathbf{r}_p(x) \right)' \hat{\gamma}_{+,p,1}(x_u),$$

which gives the following data-driven bias estimates:

$$\hat{\mathcal{B}}_- = \frac{(\bar{x} - x_l)^2}{12} \frac{1}{n} \sum_{i=1}^n \mathbb{1}_{(x_l, \bar{x})}(X_i) \hat{\mu}_{-,p,1}^{(1)}(X_i)$$

and

$$\hat{\mathcal{B}}_+ = \frac{(x_u - \bar{x})^2}{12} \frac{1}{n} \sum_{i=1}^n \mathbb{1}_{(\bar{x}, x_u)}(X_i) \hat{\mu}_{+,p,1}^{(1)}(X_i).$$

*Estimated Variance.* We also employ a global polynomial-based approach. Specifically, we compute

$$\hat{\sigma}_{-,p}^2(x) = \hat{\mu}_{-,p,2}(x) - (\hat{\mu}_{-,p,1}(x))^2,$$

$$\hat{\sigma}_{+,p}^2(x) = \hat{\mu}_{+,p,2}(x) - (\hat{\mu}_{+,p,1}(x))^2,$$

which leads to the following data-driven variance estimates:

$$\hat{\mathcal{V}}_- = \frac{1}{\bar{x} - x_l} \int_{x_l}^{\bar{x}} \hat{\sigma}_{-,p}^2(x) dx \quad \text{and} \quad \hat{\mathcal{V}}_+ = \frac{1}{x_u - \bar{x}} \int_{\bar{x}}^{x_u} \hat{\sigma}_{+,p}^2(x) dx,$$

where in practice we approximate the integral using numerical integration methods.

In summary, the proposed data-driven optimal bin-length choices are given by

$$\hat{J}_{-,n} = \left\langle (\hat{\mathcal{C}}_- n)^{1/3} \right\rangle, \quad \hat{\mathcal{C}}_- = \frac{2 \hat{\mathcal{B}}_-}{\hat{\mathcal{V}}_-},$$

and

$$\hat{J}_{+,n} = \left\langle (\hat{\mathcal{C}}_+ n)^{1/3} \right\rangle, \quad \hat{\mathcal{C}}_+ = \frac{2 \hat{\mathcal{B}}_+}{\hat{\mathcal{V}}_+}.$$

Our command `rdbinselect` implement these bin-length selector and also gives the other ingredients required to construct the plots discussed in this section. See Section 5 for syntax details of `rdbinselect`, and Section 6 for an example of this command.

### 3 rdrobust syntax

This section describes the syntax of the command `rdrobust` to conduct point estimation and “robust” inference for mean treatment effects in the context of the sharp RD design.

#### 3.1 Syntax

```
rdrobust depvar runvar [if][in][, c(cutoff)
         p(pvar) q(qvar) kernel(kernelfn)
         h(hvar) b(bvar) rho(rhovar) bwselect(bwmethod) delta(deltavar)
         vce(vcemethod) matches(nummatches) level(level) all ]
```

*depvar* is the dependent variable.

*runvar* is the running variable (a.k.a. score or forcing variable).

*cutoff* is the RD cutoff. Default is 0.

*pvar* is the order of the local-polynomial used to construct the point-estimator. Default is 1 (local-linear).

*qvar* is the order of the local-polynomial used to construct the bias-correction. Default is 2 (local-quadratic).

*kernelfn* is the kernel function used to construct the local-polynomial estimator(s). Default is `triangular`. Others options are `uniform` and `epanechnikov`.

*hvar* allows specifying directly a choice for main bandwidth  $h_n$ . If not specified, this is computed by the companion command `rdbwselect`.

*bvar* allows specifying directly a choice for pilot bandwidth  $b_n$ . If not specified, this is computed by the companion command `rdbwselect`.

*rhovar* sets the value of  $\rho$  so that  $b_n = h_n/\rho$ . If this option is specified, then the choice of  $h_n$  (either provided by the user or estimated by `rdbwselect`) is used together with  $\rho$  to construct  $b_n$ .

*bwmethod* is the method used to estimate the bandwidth(s). Default is `CCT`. Other options are `IK` and `CV`.

*deltavar* sets the quantile that defines the sample used in the cross-validation procedure. This option is used if `bwselect(CV)` is specified. Default is 0.5 (i.e., median of the control and treated samples).

*vcemethod* is the method used to construct estimated residuals used in standard error formula. number of nearest-neighbor matches used for the variance-covariance matrix estimator. Default is `nn` for fixed-matches nearest-neighbor. The other option is `resid` for estimated plug-in residuals.

*nummatches* is the number of nearest-neighbor matches used for the variance-covariance matrix estimator when either `vce(vcmethod)` is not specified or `vce(nn)` is specified. Default is 3 matches.

*level* sets the confidence interval level.

*all* implements three different procedures: conventional RD estimates with conventional standard-errors, bias-corrected RD estimates with conventional standard-errors, and bias-corrected RD estimates with robust standard-errors.

### 3.2 Description

`rdrobust` provides an array of local-polynomial-based inference procedures for mean treatment effects in the sharp RD design.

You must specify the dependent and running variables. This command permits for fully data-driven inference by employing the companion command `rdbwselect`, which may also be used as a stand-alone command. We describe the `rdbwselect` command in the next section.

## 4 rdbwselect syntax

This section describes the syntax of the command `rdbwselect`. This command implements the different bandwidth selection procedures for Local-Polynomial Regression-Discontinuity estimators discussed above.

### 4.1 Syntax

```
rdbwselect depvar runvar [if][in][, c(cutoff)
           p(pvar) q(qvar) kernel(kernelfn)
           bwselect(bwmethod) delta(deltavar) rho(rhovar)
           vce(vcmethod) matches(nummatches) all]
```

*depvar* is the dependent variable.

*runvar* is the running variable (a.k.a. score or forcing variable).

*cutoff* is the RD cutoff. Default is 0.

*pvar* is the order of the local-polynomial used to construct the point-estimator. Default is 1 (local-linear).

*qvar* is the order of the local-polynomial used to construct the bias-correction. Default is 2 (local-quadratic).

*kernelfn* is the kernel function used to construct the local-polynomial estimator(s). De-

fault is `triangular`. Others options are `uniform` and `epanechnikov`.

`bwmethod` is the method used to estimate the bandwidth(s). Default is `CCT`. Other options are `IK` and `CV`.

`deltavar` sets the quantile that defines the sample used in the cross-validation procedure. This option is used if `bwselect(CV)` is specified. Default is 0.5 (i.e., median of the control and treated samples).

`rho`var sets the value of  $\rho$  so that  $b_n = h_n/\rho$ . If this option is specified, then the estimated  $h_n$  is used together with  $\rho$  to construct  $b_n$ .

`vcmethod` is the method used to construct estimated residuals used in standard error formula. number of nearest-neighbor matches used for the variance-covariance matrix estimator. Default is `nn` for fixed-matches nearest-neighbor. The other option is `resid` for estimated plug-in residuals.

`nummatches` is the number of nearest-neighbor matches used for the variance-covariance matrix estimator when either `vce(vcmethod)` is not specified or `vce(nn)` is specified. Default is 3 matches.

`all` implements all three previous procedures.

## 4.2 Description

`rdbwselect` implements several bandwidth selectors procedures currently available in the literature for the RD design. You must specify the dependent and running variable.

## 5 rdbinselect syntax

This section describes the syntax of the command `rdbinselect`, which implements a bin-length selector useful to construct RD plots of the estimated regression function for control and treated units together with binned-sample-means of the underlying data. This common RD figure gives an idea of overall fit, while also exhibits graphically the (sharp) RD estimate.

### 5.1 Syntax

```
rdbinselect depvar runvar [if][in][, c(cutoff)
             p(pvar) lowerend(xlval) upperend(xuval) scale(scaleval)
             genid(idname) genmeanx(meanxname) genmeany(meanyname) hide]
```

*depvar* is the dependent variable.

*runvar* is the running variable (a.k.a. score or forcing variable).

*cutoff* is the RD cutoff. Default is 0.

*xlval* specifies the minimum value of the running variable used. Default is the minimum observed value in the data.

*xuval* specifies the maximum value of the running variable used. Default is the maximum observed value in the data.

*scaleval* specifies a multiplicative factor to be used with the optimal number of bins selected. Specifically, for the control and treated units, respectively, the number of bins used will be  $\langle scaleval \cdot \hat{J}_{-,n} \rangle$  and  $\langle scaleval \cdot \hat{J}_{+,n} \rangle$ .

*idname* specifies the name of a new generated variable with a unique bin id. Negative natural numbers are assigned to control units, and positive natural numbers are assigned to treated units.

*meanxname* specifies the name of a new generated variable with the sample-mean within bin of the running variable.

*meanyname* specifies the name of a new generated variable with the sample-mean within bin of the outcome variable.

*hide* omits the final RD plot.

## 5.2 Description

`rdbinselect` implement a bin-length selector required to construct plots of the estimated regression function for control ( $X_i < \bar{x}$ ) and treated ( $X_i \geq \bar{x}$ ) units.

You must specify the dependent and running variables.

## 6 Illustration of Methods

We illustrate our Stata commands using an extract of the dataset constructed by Cattaneo, Frandsen, and Titiunik (2012), which studies several measures of incumbency advantage in the U.S. Senate for the period 1914–2010. In particular, we focus here on the (RD) effect of the democratic party winning a U.S. Senate seat on the vote share obtained in the following election for that same seat. This empirical illustration is analogous to the one presented by Lee (2008) for the U.S. House.

The dataset `rdrobust_RDsenate.dta` contains two variables: *vote* and *margin*. The variable *vote* records the vote share of the democratic candidate in the next election, while the variable *margin* records the margin of victory of the democratic candidate in the previous election. We load the database and present basic summary statistics of these two variables:

```
. use rdrobust_RDsenate.dta, clear
. sum vote margin
```

| Variable | Obs  | Mean     | Std. Dev. | Min  | Max |
|----------|------|----------|-----------|------|-----|
| vote     | 1279 | 52.32349 | 17.84619  | 0    | 100 |
| margin   | 1379 | 6.736008 | 33.92505  | -100 | 100 |

This data set has a total of 1,279 complete observations. The variable *margin* ranges from  $-100$  to  $100$ , and records the margin of victory in a given election for a given U.S. Senate seat, while the variable *vote* ranges from  $0$  to  $100$  because it records the outcome of the (two-periods ahead) election for that given seat. Thus, observations for years 2008 and 2010 have missing *vote*. As it is usual in the literature, we exploit the discontinuity at  $0$  on *margin* to employ the RD empirical strategy.

To gain further intuition on the available data, we use `rdbinselect` to construct an automatic plot of the RD design.

```
. rdbinselect vote margin

Number of bins for RD Estimates
-----+-----
Cutoff c = 0 | Left of c   Right of c
-----+-----
Number of obs |      594     685
Poly. Order   |      4       4
Number of bins |     208     233
Bin Length    |    0.480    0.429
```

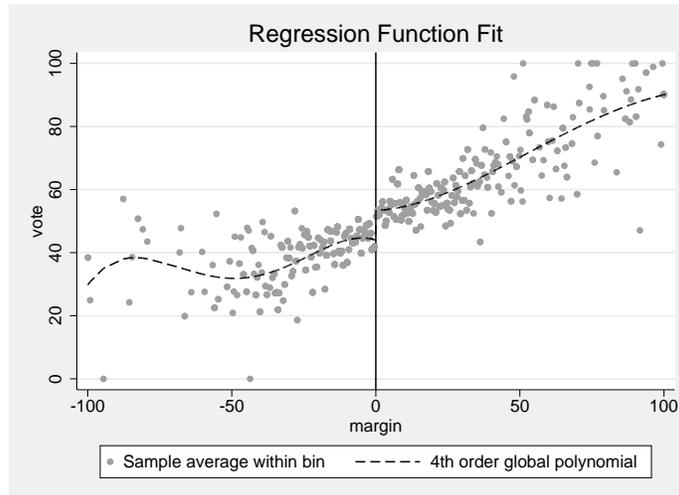


Figure 2: Automatic RD plot.

Figure 2 is constructed using the default options in the command `rdbinselect`. Using the notation introduced above, the number of optimal bins for control and treatment units are  $\hat{J}_{-,n} = 208$  and  $\hat{J}_{+,n} = 233$ , respectively, implying a bin length of roughly half a percent-point. The global polynomial is constructed using a 4-th degree polynomial

( $p = 4$  for  $\hat{\mu}_{-,p,1}(x)$  and  $\hat{\mu}_{+,p,1}(x)$ ). Figure 1 shows the resulting RD plot.

Each bin in Figure 1 contains roughly 3-4 observations, which leads in this case to a dense “cloud of points”. To reduce the overall variability of the plotted points, we may increase the bin-length which is done by decreasing the total number of bins used. This is done using the option `scale(.)` as follows:

```
. rdbinselect vote margin, scale(.5)
Number of bins for RD Estimates
```

| Cutoff $c = 0$ | Left of $c$ | Right of $c$ |
|----------------|-------------|--------------|
| Number of obs  | 594         | 685          |
| Poly. Order    | 4           | 4            |
| Number of bins | 208         | 233          |
| Bin Length     | 0.480       | 0.429        |
| Scale          | 0.500       | 0.500        |
| Number of bins | 104         | 117          |
| Bin Length     | 0.961       | 0.854        |

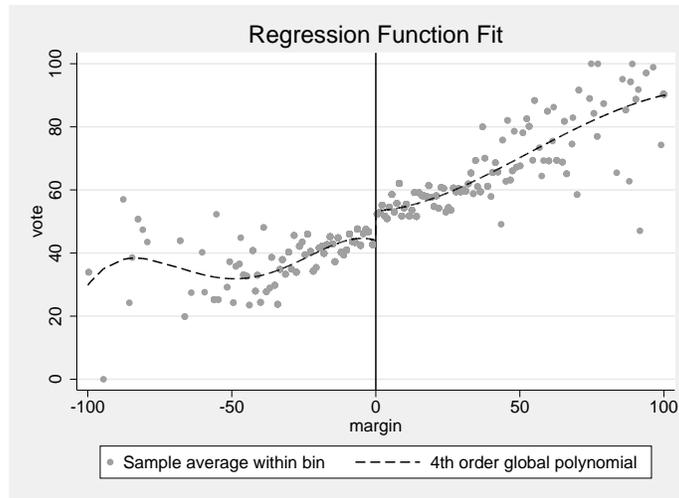


Figure 3: Automatic RD plot, with scaled up optimal bin-length choice.

Next, we conduct a fully data-driven RD treatment-effect estimation. The command `rdrobust` using its default options leads to the following output:

```
. rdrobust vote margin
Preparing data
..... done
Computing EW Selectors
.....done
Computing Variance-Covariance Matrix
.....done
Computing RD Estimates
.....done
```

```

Estimation Completed
Sharp RD Estimates using Local Polynomial Regression

```

| Cutoff c = 0         |  |        | Left of c | Right of c |                 |            |
|----------------------|--|--------|-----------|------------|-----------------|------------|
| Number of obs        |  | 330    |           | 294        | Number of obs = | 1279       |
| Order Loc. Poly. (p) |  | 1      |           | 1          | Rho (h/b) =     | 0.611      |
| Order Bias (q)       |  | 2      |           | 2          | NN Matches =    | 3          |
| BW Loc. Poly. (h)    |  | 15.884 |           | 15.884     | BW Type =       | CCT        |
| BW Bias (b)          |  | 26.015 |           | 26.015     | Kernel Type =   | Triangular |

|        | Loc. Poly. | Robust    |        |       | [Robust   |           |
|--------|------------|-----------|--------|-------|-----------|-----------|
| vote   | Coef.      | Std. Err. | z      | P> z  | 95% Conf. | Interval] |
| margin | 7.43883    | 1.86      | 4.1282 | 0.000 | 4.027283  | 11.308182 |

These results contain a variety of information, which is organized in two panels. The upper panel of the output table contains a summary of the main choices selected to construct the RD treatment-effect estimators, while the lower panel includes the main estimation results. Specifically, using the notation introduced above, this table shows:

1. The total number of observations is 1,279, with effective 330 control and 294 treated units (given the bandwidth  $h_n$  chosen; see below). The estimation is conducted using a local-linear ( $p = 1$ ) estimator with a local-quadratic ( $q = 2$ ) bias-correction estimate, with a triangular kernel. The standard-error estimators are the robust ones proposed by CCT, computed using 3 nearest-neighbors.
2. The bandwidth selection procedure is the one proposed by CCT, leading to

$$\hat{h}_{CCT,n,p} = 15.884 \quad (p = 1) \quad \text{and} \quad \hat{b}_{CCT,n,q} = 26.015 \quad (q = 2).$$

3. The point estimator, robust standard-errors and robust confidence intervals are:

$$\hat{\tau}_p(\hat{h}_{CCT,n,p}) = 7.43883, \quad \hat{V}_{n,q,p}^{bc}(\hat{h}_{CCT,n,p}, \hat{b}_{CCT,n,q}) = (1.86)^2,$$

$$\hat{CI}_\alpha^{rbc}(\hat{h}_{CCT,n,p}, \hat{b}_{CCT,n,q}) = [4.027, 11.308] \quad (\alpha = 0.95).$$

The command `rdrobust` also offers a more detailed output, which includes all the point estimators, standard-errors estimators and confidence intervals discussed in Section 2. These results are retrieved by including the option `all`. The corresponding output is as follows:

```

. rdrobust vote margin, all
Preparing data
..... done
Computing BW Selectors
.....done
Computing Variance-Covariance Matrix
.....done
Computing RD Estimates
.....done
Estimation Completed

Sharp RD Estimates using Local Polynomial Regression

```

| Cutoff c = 0   Left of c |        | Right of c |                          |  |  |  |
|--------------------------|--------|------------|--------------------------|--|--|--|
| Number of obs            | 330    | 294        | Number of obs = 1279     |  |  |  |
| Order Loc. Poly. (p)     | 1      | 1          | Rho (h/b) = 0.611        |  |  |  |
| Order Bias (q)           | 2      | 2          | NN Matches = 3           |  |  |  |
| BW Loc. Poly. (h)        | 15.884 | 15.884     | BW Type = CCT            |  |  |  |
| BW Bias (b)              | 26.015 | 26.015     | Kernel Type = Triangular |  |  |  |

| vote   | Loc. Poly. Coef. | Robust Std. Err. | z      | P> z  | [Robust 95% Conf. Interval] |           |
|--------|------------------|------------------|--------|-------|-----------------------------|-----------|
| margin | 7.43883          | 1.86             | 4.1282 | 0.000 | 4.027283                    | 11.308182 |

All Estimates. Outcome: vote. Running Variable: margin.

| Method         | Coef.    | Std. Err. | z      | P> z  | [95% Conf. Interval] |           |
|----------------|----------|-----------|--------|-------|----------------------|-----------|
| Conventional   | 7.43883  | 1.58      | 4.7118 | 0.000 | 4.344484             | 10.533175 |
| Bias-Corrected | 7.667732 | 1.58      | 4.8568 | 0.000 | 4.573387             | 10.762078 |
| Robust         | 7.667732 | 1.86      | 4.1282 | 0.000 | 4.027283             | 11.308182 |

This detailed output contains an additional table located at the bottom, relative to the default output. Using our notation and the options specified in the case above, this extra table has the following structure:

Table 1: Bottom table structure in output of `rdrobust ... , all`.

| Method         | Coef.                             | Std. Err.                               | z | P> z | [95% Conf. Interval]              |
|----------------|-----------------------------------|---|---|------|-----------------------------------|
| Conventional   | $\hat{\tau}_p(h_n)$               | $\sqrt{\hat{V}_p/nh_n}$                 | . | .    | $\hat{C}I_\alpha(h_n)$            |
| Bias-Corrected | $\hat{\tau}_{p,q}^{bc}(h_n, b_n)$ | $\sqrt{\hat{V}_p/nh_n}$                 | . | .    | $\hat{C}I_\alpha^{bc}(h_n, b_n)$  |
| Robust         | $\hat{\tau}_{p,q}^{bc}(h_n, b_n)$ | $\sqrt{\hat{V}_{n,q,p}^{bc}(h_n, b_n)}$ | . | .    | $\hat{C}I_\alpha^{rbc}(h_n, b_n)$ |

Note: In the output above,  $p = 1$ ,  $q = 2$ ,  $h_n = \hat{h}_{CCT,n,p}$ ,  $b_n = \hat{b}_{CCT,n,q}$ , and  $\alpha = 0.95$ .

Finally, we explore all the bandwidth selection procedures contained in our package. Specifically, we may employ our companion package `rdbwselect` to compare the CCT bandwidth selectors with the IK and CV bandwidth selectors. We have:

```
. rdwbselect vote margin, all
Grid Search: 0% - 11:44:47
Grid Search: 56% - 11:44:50

Bandwidth Estimators for RD Local Polynomial Regression

Number of obs      = 1279
Order Loc. Poly. (p) = 1
Order Bias (q)     = 2
```

| Method | h        | b        | rho      |
|--------|----------|----------|----------|
| CCT    | 15.88415 | 26.0151  | .6105745 |
| IK     | 17.51627 | 15.88309 | 1.102825 |
| CV     | 21.35928 | 21.35928 | 1        |

In this case we employed the option `all`, which computes the three bandwidth selectors briefly discussed above. In the case of  $h_n$  these choices range from 15.88 to

21.35, while in the case of  $b_n$  we obtain  $\hat{b}_{\text{CCT},n,q} = 26.0151$  and  $\hat{b}_{\text{IK},n,q} = 15.88309$  for the CCT and IK methods, respectively. Notice that the option CV is currently not available for derivative estimation.

As discussed above, our commands have many other options. For example, for the main command `rdrobust` we have the following additional examples (output is not provided to conserve space):

1. `rdrobust vote margin, kernel(uniform)`  
Estimation using uniform kernel.
2. `rdrobust vote margin, bwselect(IK)`  
Estimation using the IK bandwidth selectors.
3. `rdrobust vote margin, bwselect(CV)`  
Estimation using the CV bandwidth selector (and  $\rho = 1$ ).
4. `rdrobust vote margin, h(15) rho(0.8)`  
Estimation using  $h_n = 15$  and  $b_n = h_n/\rho = 15/0.8 = 18.75$ .
5. `rdrobust vote margin, p(2) q(4)`  
Estimation using  $p = 2$  and  $q = 4$ .
6. `rdrobust vote margin, vce(resid)`  
Estimation using plug-in residuals estimates in the VCE.

## 7 Conclusion

This article discussed the implementation of several data-driven, local-polynomial based (robust) inference procedures in the RD design. We introduced three main commands that together offer an array of data-driven nonparametric inference methods useful to carry “automatic” empirical work in the context of RD applications. In future versions of our commands we plan to extend our methods to cover fuzzy RD (e.g., Hahn et al. (2001) and Porter (2003)), quantile RD treatment-effects (e.g., Frandsen et al. (2012)), and nonparametric RD specification testing (e.g., McCrary (2008)), among other possibilities.

## 8 References

- Abadie, A., and G. W. Imbens. 2006. Large Sample Properties of Matching Estimators for Average Treatment Effects. *Econometrica* 74(1): 235–267.
- Calonico, S., M. D. Cattaneo, and R. Titiunik. 2012. Robust Nonparametric Bias-Corrected Inference in the Regression Discontinuity Design. Working paper, University of Michigan .

- Cattaneo, M. D., and M. H. Farrell. 2012. Optimal Convergence Rates, Bahadur Representation, and Asymptotic Normality of Partitioning Estimators. Working paper, University of Michigan .
- Cattaneo, M. D., B. Frandsen, and R. Titiunik. 2012. Randomization Inference in the Regression Discontinuity Design to Study the Incumbency Advantage in the U.S. Senate. Working paper, University of Michigan .
- Cheng, M.-Y., J. Fan, and J. S. Marron. 1997. On Automatic Boundary Corrections. *Annals of Statistics* 25(4): 1691–1708.
- Cook, T. D. 2008. “Waiting for Life to Arrive”: A history of the regression-discontinuity design in Psychology, Statistics and Economics. *Journal of Econometrics* 142(2): 636–654.
- Dinardo, J., and D. S. Lee. 2011. Program Evaluation and Research Designs. In *Handbook of Labor Economics*, ed. O. Ashenfelter and D. Card, vol. 4A, 463–536. Elsevier Science B.V.
- Fan, J., and I. Gijbels. 1996. *Local Polynomial Modelling and Its Applications*. New York: Chapman & Hall/CRC.
- Frandsen, B., M. Frölich, and B. Melly. 2012. Quantile treatments effects in the regression discontinuity design. *Journal of Econometrics* 168(2): 382–395.
- Hahn, J., P. Todd, and W. van der Klaauw. 2001. Identification and Estimation of Treatment Effects with a Regression-Discontinuity Design. *Econometrica* 69(1): 201–209.
- Imbens, G., and T. Lemieux. 2008. Regression Discontinuity Designs: A Guide to Practice. *Journal of Econometrics* 142(2): 615–635.
- Imbens, G. W., and K. Kalyanaraman. 2012. Optimal Bandwidth Choice for the Regression Discontinuity Estimator. *Review of Economic Studies* 79(3): 933–959.
- Imbens, G. W., and J. M. Wooldridge. 2009. Recent Developments in the Econometrics of Program Evaluation. *Journal of Economic Literature* 47(1): 5–86.
- Lee, D. S. 2008. Randomized Experiments from Non-random Selection in U.S. House Elections. *Journal of Econometrics* 142(2): 675–697.
- Lee, D. S., and T. Lemieux. 2010. Regression Discontinuity Designs in Economics. *Journal of Economic Literature* 48(2): 281–355.
- Ludwig, J., and D. L. Miller. 2007. Does Head Start Improve Children’s Life Chances? Evidence from a Regression Discontinuity Design. *Quarterly Journal of Economics* 122(1): 159–208.
- McCrary, J. 2008. Manipulation of the running variable in the regression discontinuity design: A density test. *Journal of Econometrics* 142(2): 698–714.

Porter, J. 2003. Estimation in the Regression Discontinuity Model. working paper, University of Wisconsin .

van der Klaauw, W. 2008. Regression-Discontinuity Analysis: A Survey of Recent Developments in Economics. *Labour* 22(2): 219–245.

Wand, M., and M. Jones. 1995. *Kernel Smoothing*. Florida: Chapman & Hall/CRC.