

rddensity: Manipulation Testing Based on Density Discontinuity in R

Matias D. Cattaneo*

Michael Jansson[†]

Xinwei Ma[‡]

July 3, 2017

Abstract

This document gives a brief discussion of three R functions implementing automatic manipulation tests based on the results in [Cattaneo, Jansson, and Ma \[2017a\]](#). These functions have the same syntax, and offer the same functionalities, as our companion **Stata** commands described in [Cattaneo, Jansson, and Ma \[2017b\]](#). The first function, `rddensity()`, implements manipulation tests based on a novel local polynomial density estimation technique that avoids pre-binning of the data (improving size properties) and allows for restrictions on other features of the model (improving power properties). The second function, `rdbwdensity()`, implements several bandwidth selectors specifically tailored for the manipulation tests discussed herein. The third function, `rdplotdensity()`, constructs density plot near the cutoff.

Keywords: `rddensity`, `rdbwdensity`, `rdplotdensity`, falsification test, manipulation test, regression discontinuity.

*Department of Economics, Department of Statistics, University of Michigan.

[†]Department of Economics, UC Berkeley and *CREATES*.

[‡]Department of Economics, Department of Statistics, University of Michigan.

Contents

1	Introduction	1
2	Notation and Methods Overview	1
2.1	Setup and Notation.....	1
2.2	Unrestricted Testing.....	2
2.3	Restricted Testing.....	3
2.4	Standard Errors	3
2.5	Bandwidth Selection	4
3	Illustration of Methods.....	6

1 Introduction

This document gives a brief discussion of three R functions implementing automatic manipulation tests based on the results in Cattaneo, Jansson, and Ma [2017a]. These functions have the same syntax, and offer the same functionalities, as our companion **Stata** commands described in Cattaneo, Jansson, and Ma [2017b]. The first function, `rddensity()`, implements manipulation tests based on a novel local polynomial density estimation technique that avoids pre-binning of the data (improving size properties) and allows for restrictions on other features of the model (improving power properties). The second function, `rdbwdensity()`, implements several bandwidth selectors specifically tailored for the manipulation tests discussed herein. The third function, `rdpltdensity()`, constructs density plot near the cutoff.

To avoid repetition, please refer to our companion papers Cattaneo, Jansson, and Ma [2017a,b] for further discussion on theoretical, methodological and practical aspects of the R functions discussed herein. Details on syntax of the functions can be found in the [reference manual](#).

For related **Stata** and R packages useful for analysis of RD designs, visit:

<https://sites.google.com/site/rdpackages>

2 Notation and Methods Overview

This section summarize the main notation used throughout the document, and also offers a very short overview of the methods implemented in our functions `rddensity()` and `rdbwdensity()`. We follow closely the notation and discussion given in Cattaneo, Jansson, and Ma [2017b], which in turn relies on results given in Cattaneo, Jansson, and Ma [2017a] (hereafter CJM) and its supplemental appendix.

2.1 Setup and Notation

We assume that $\{X_i : i = 1, 2, \dots, n\}$ is a random sample from the random variable X with cumulative distribution function (c.d.f.) and probability density function (p.d.f.) given, respectively, by $F(x)$ and $f(x)$. The random variable X_i denotes the score, index or running variable of unit i in the sample. Each unit is assigned to control or treatment depending on whether their observed index exceeds a known cutoff denoted by $\bar{x}$. That is, group or “treatment” assignment is given by:

$$\begin{array}{ll} \text{unit } i \text{ assigned to control group} & \text{if } X_i < \bar{x} \\ \text{unit } i \text{ assigned to treatment group} & \text{if } X_i \geq \bar{x} \end{array}$$

where the cutoff point $\bar{x}$ is known and, of course, we assume enough observations for each group are available (e.g., $f(x) > 0$ near $\bar{x}$ and the sample is large enough).

A manipulation test in this context is a hypothesis test on the continuity of the density $f(\cdot)$

at the cutoff point $\bar{x}$. Formally, we are interested in the testing problem:

$$H_0 : \lim_{x \uparrow \bar{x}} f(x) = \lim_{x \downarrow \bar{x}} f(x) \quad \text{vs} \quad H_1 : \lim_{x \uparrow \bar{x}} f(x) \neq \lim_{x \downarrow \bar{x}} f(x).$$

We construct a test statistic for this hypothesis testing problem employing local polynomial density estimator based on the c.d.f. of the observed sample.

The class of manipulation test statistics implemented in `rddensity()` take the form:

$$T_p(h_n) = \frac{\hat{f}_{+,p}(h_n) - \hat{f}_{-,p}(h_n)}{\hat{V}_p(h_n)}, \quad \hat{V}_p^2(h_n) = \hat{V}[\hat{f}_{+,p}(h_n) - \hat{f}_{-,p}(h_n)],$$

where $T_p(h_n) \stackrel{a}{\sim} \mathcal{N}(0, 1)$ under appropriate assumptions. This statistic may be constructed in several different ways, as we discuss in more detail below. Two main ingredients to construct the test statistic are (i) the local polynomial density estimators $\hat{f}_{+,p}(h_n)$ and $\hat{f}_{-,p}(h_n)$, and (ii) the corresponding standard error estimator $\hat{V}_p(h_n)$. These estimators depend on the choice of polynomial order p , the choice of kernel function $K(\cdot)$ and the restrictions imposed in the model, among other possibilities. The standard error formulas $\hat{V}_p(h_n)$ could be based on either an asymptotic plugin or jackknife approach, and its specific form will depend on whether additional restrictions are imposed to the model. Another crucial ingredient is of course the choice of bandwidth h_n , which determines which observations near the cutoff $\bar{x}$ are used for estimation and inference. This choice can either be specified by the user or estimated using the available data. Our functions allow, when possible, for different bandwidth choices on either side of the cutoff $\bar{x}$. A common bandwidth on both sides of the cutoff is always possible.

2.2 Unrestricted Testing

In unrestricted testing, the manipulation test becomes a standard two-sample problem where the estimators $\hat{f}_{+,p}(h_n)$ and $\hat{f}_{-,p}(h_n)$ are unrelated. Thus, the standard errors formula reduces to $\hat{V}_p^2(h_n) = \hat{V}[\hat{f}_{+,p}(h_n) - \hat{f}_{-,p}(h_n)] = \hat{V}[\hat{f}_{+,p}(h_n)] + \hat{V}[\hat{f}_{-,p}(h_n)]$. To be more concrete, the density estimators take the form:

$$\hat{f}_{-,p}(h_n) = \mathbf{e}'_1 \hat{\boldsymbol{\beta}}_{-,p}(h_n) \quad \text{and} \quad \hat{f}_{+,p}(h_n) = \mathbf{e}'_1 \hat{\boldsymbol{\beta}}_{+,p}(h_n),$$

with

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

where $\hat{F}(X_i)$ denotes the (leave-one-out) c.d.f. estimator for the full sample, $\mathbf{r}_p(x) = (1, x, \dots, x^p)'$, $\mathbf{e}_1 = (0, 1, 0, 0, \dots, 0)' \in \mathbb{R}^{p+1}$ is the second unit vector, $K_h(u) = K(u/h)/h$ with $K(\cdot)$ a kernel

function, h_n is a positive bandwidth sequence, and $\mathbf{1}(\cdot)$ denotes the indicator function. Employing local polynomial techniques, it may be shown that $\hat{\beta}_{-,p}(h_n)$ and $\hat{\beta}_{+,p}(h_n)$ roughly approximate $(F_-, f_-, \frac{1}{2!}f_-^{(1)}, \dots, \frac{1}{(p+1)!}f_-^{(p)})$ and $(F_+, f_+, \frac{1}{2!}f_+^{(1)}, \dots, \frac{1}{(p+1)!}f_+^{(p)})$, respectively, where we employ the notation:

$$f_-^{(s)} = \lim_{x \uparrow \bar{x}} \frac{\partial^s}{\partial x^s} f(x) \quad \text{and} \quad f_+^{(s)} = \lim_{x \downarrow \bar{x}} \frac{\partial^s}{\partial x^s} f(x),$$

$s = 1, 2, \dots, p$. Thus, the second elements of $\hat{\beta}_{-,p}(h_n)$ and $\hat{\beta}_{+,p}(h_n)$ are used to construct consistent, boundary-corrected density estimators at the cutoff point $\bar{x}$ entering the numerator of the manipulation test statistic. Notice that this estimation approach avoids pre-binning of the data and may be constructed using simple and easy-to-interpret kernels $K(\cdot)$ (e.g., uniform or triangular).

2.3 Restricted Testing

A restricted manipulation test is constructed by solving the above weighted (local) least-squares problem with the additional restrictions ensuring that all but the second element in β are equal in both groups. It follows that this restricted problem can be represented as a single regression problem:

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

where $\mathbf{r}_p^R(x) = (1, x \cdot \mathbf{1}(x < \bar{x}), x \cdot \mathbf{1}(x \geq \bar{x}), x^2, x^3, \dots, x^p)'$. In words, this estimation approach incorporates the restrictions ensuring that the estimated c.d.f. as well as the estimated higher-order derivatives are equal on both sides of the cutoff point $\bar{x}$. This problem involves estimating only $p + 2$ parameters, rather than $2p + 2$ parameters as it is the case for the unrestricted method discussed above. The main advantage of imposing these restrictions is related to power improvements, provided the restrictions are indeed satisfied in the underlying data generating process.

Therefore, a restricted manipulation test employs the density estimators:

$$\hat{f}_{-,p}^R(h_n) = \mathbf{e}_1' \hat{\beta}_p^R(h_n) \quad \text{and} \quad \hat{f}_{+,p}^R(h_n) = \mathbf{e}_2' \hat{\beta}_p^R(h_n),$$

that is, the density estimators from the left and from the right of the cutoff point $\bar{x}$ are given by the second and third element of in the least-squares vector $\hat{\beta}_{+,p}^R(h_n)$; $\mathbf{e}_2 = (0, 0, 1, 0, 0, \dots, 0)' \in \mathbb{R}^{p+1}$ denotes the third unit vector. Furthermore, the standard errors formula is different because now cross-restrictions are incorporated in the estimation procedure, leading to a different asymptotic variance for $(\hat{f}_{-,p}(h_n), \hat{f}_{+,p}(h_n))$ and, in consequence, for $\hat{f}_{+,p}(h_n) - \hat{f}_{-,p}(h_n)$ as well. This is exactly the source of the power gains of a restricted manipulation test relative to an unrestricted one. In this case, the standard errors formula $\hat{V}_p^2(h_n) = \hat{\mathbb{V}}[\hat{f}_{+,p}^R(h_n) - \hat{f}_{-,p}^R(h_n)] \neq \hat{\mathbb{V}}[\hat{f}_{+,p}(h_n)] + \hat{\mathbb{V}}[\hat{f}_{-,p}(h_n)]$.

2.4 Standard Errors

The asymptotic variance entering the denominator of the manipulation test statistic $T_p(h_n)$ are different depending on whether an unrestricted or a restricted model is used. Our functions

`rddensity()` and `rdbwdensity()` allow for both cases. In addition, for each of these cases, the functions provide two distinct consistent standard errors estimators: (i) a plugin estimator based on the asymptotic variance of the numerator of $T_p(h_n)$, and (ii) a jackknife estimator based on the leading term of an expansion of asymptotic variance of the numerator of $T_p(h_n)$.

The plugin estimator is faster because it essentially requires no additional estimation beyond the quantities entering the numerator of $T_p(h_n)$, but it relies on asymptotic approximations. On the other hand, the jackknife estimator is slower because it requires additional estimation and looping over the data, but according to simulation evidence in CJM it appears to provide a more accurate approximation to the finite-sample variability of the numerator of $T_p(h_n)$ in both cases (unrestricted model and restricted model). Therefore, our implementations employ the jackknife standard errors estimator by default, but we also offer the plugin estimator for cases involving relatively large sample sizes.

2.5 Bandwidth Selection

The function `rddensity()` either requires specifying bandwidths for estimation, or employs the function `rdbwdensity()` to construct data-driven bandwidth choices specifically tailored for the manipulation tests discussion in this article. In this subsection we describe briefly the data-driven implementations provided in `rdbwdensity()` for automatic bandwidth selection.

In the unrestricted model, the user has the option to specify two distinct bandwidths: h_1 for left estimation and h_r for right estimation. Of course, one such choice may be equal bandwidths: $h = h_1 = h_r$. In the restricted model, however, only a common bandwidth h can be specified because estimation is done jointly by construction. For each of these cases, whenever possible, CJM develops three distinct approaches to select the bandwidth(s) employing the mean squared error (MSE) criterion function, generically denoted by $\text{MSE}[\hat{\theta}] = \mathbb{E}[(\hat{\theta} - \theta)^2]$, where $\hat{\theta}$ denotes some estimator and θ denotes its target estimand.

For the specific context considered herein, we need to consider four MSE criterion functions. These results can be briefly summarized as follows:

$$\text{MSE}[\hat{\theta}(h_n)] \approx \text{AMSE}[\hat{\theta}(h_n)],$$

where

$$\text{AMSE}[\hat{\theta}(h_n)] = h_n^{p+1} B_p(\theta) + h_n^{p+2} B_{p+1}(\theta) + \frac{1}{nh_n} V_p(\theta),$$

with, for the unrestricted model,

$$\hat{\theta}(h) \text{ representing one of } \{\hat{f}_{-,p}(h); \hat{f}_{+,p}(h); \hat{f}_{+,p}(h) - \hat{f}_{-,p}(h); \hat{f}_{+,p}(h) + \hat{f}_{-,p}(h)\}$$

while, for the restricted model,

$$\hat{\theta}(h) \text{ representing one of } \{\hat{f}_{+,p}^R(h) - \hat{f}_{-,p}^R(h); \hat{f}_{+,p}^R(h) + \hat{f}_{-,p}^R(h)\},$$

and, of course,

$$\theta \text{ representing one of } \{f_-; f_+; f_+ - f_-; f_+ + f_-\},$$

as appropriate according to the choice of $\hat{\theta}(h)$. Crucially, for each combination of estimator $\hat{\theta}(h_n)$ and estimand θ , the corresponding bias constants $(B_p(\theta), B_{p+1}(\theta))$ and variance constant $V_p(\theta)$ are different. In all cases, as it is usually the case in nonparametric problems, these constants involve features of both the data generating process as well as the nonparametric estimator.

Given a choice of estimator and estimand, and provided preliminary estimates of the leading asymptotic constants in the associated MSE expansion are available, it is straightforward to construct a plug-in bandwidth selector. In our implementations, we consider the five alternative plug-in rules for bandwidth selection mentioned above, which employ simple rule-of-thumbs to approximate the leading constants in the MSE expansions.

Specifically, `rdwbdensity()` allows for the following alternative bandwidth selectors. We do not introduce additional notation to reflect the estimation of the leading AMSE constants only to ease the exposition, but our implementations are automatic as they rely on preliminary rule-of-thumb estimators to approximate those constants.

- *Unrestricted model with different bandwidths*: when no restrictions are imposed on the model and (h_1, h_r) are allowed to be different, the bandwidths are chosen to minimize the AMSE of the density estimators separately, that is,

$$\hat{h}_{1,p} = \min_h \text{AMSE}[\hat{f}_{-,p}(h)] \quad \text{and} \quad \hat{h}_{r,p} = \min_h \text{AMSE}[\hat{f}_{+,p}(h)].$$

- *Unrestricted model with equal bandwidths*: when no restrictions are imposed on the model but (h_1, h_r) are forced to be equal, the common bandwidth may be chosen in two distinct ways.

(1) Difference of densities:

$$\hat{h}_{\text{diff},p} = \min_h \text{AMSE}[\hat{f}_{+,p}(h) - \hat{f}_{-,p}(h)].$$

(2) Sum of densities:

$$\hat{h}_{\text{sum},p} = \min_h \text{AMSE}[\hat{f}_{+,p}(h) + \hat{f}_{-,p}(h)].$$

- *Restricted model*: when the restrictions are imposed on the model, then $h = h_1 = h_r$ by construction. In this case, the common bandwidth may be chosen in two distinct ways as well.

(1) Difference of densities:

$$\hat{h}_{\text{diff},p}^{\text{R}} = \min_h \text{AMSE}[\hat{f}_{+,p}^{\text{R}}(h) - \hat{f}_{-,p}^{\text{R}}(h)].$$

(2) Sum of densities:

$$\hat{h}_{\text{sum},p}^{\text{R}} = \min_h \text{AMSE}[\hat{f}_{+,p}^{\text{R}}(h) + \hat{f}_{-,p}^{\text{R}}(h)].$$

All the bandwidth selectors above have close form solutions, and their specific decay rate (as a function of the sample size) depend on the specific choices. This is an important point, because of $B_p(\theta)$ or $B_{p+1}(\theta)$ may be zero depending on the choice of θ and p . For example, if $\theta = f_+ - f_-$ then $B_p(\theta) \propto f_+^{(p)} - f_-^{(p)}$ when $p = 2$ but $B_p(\theta) \propto f_+^{(p)} + f_-^{(p)}$ when $p = 3$, which implies that $B_p(\theta) = 0$ under the plausible assumption that $f_+^{(p)} = f_-^{(p)}$. We also implement two simple “regularization” approaches that avoid this type of potential degeneracies:

- *Unrestricted model with different bandwidths:*

$$\hat{h}_{1,\text{comb},p} = \text{median}\{\hat{h}_{1,p}, \hat{h}_{\text{diff},p}, \hat{h}_{\text{sum},p}\}$$

$$\hat{h}_{r,\text{comb},p} = \text{median}\{\hat{h}_{r,p}, \hat{h}_{\text{diff},p}, \hat{h}_{\text{sum},p}\}.$$

- *Unrestricted model with equal bandwidths:*

$$\hat{h}_{\text{comb},p} = \min\{\hat{h}_{\text{diff},p}, \hat{h}_{\text{sum},p}\}.$$

- *Restricted model:*

$$\hat{h}_{\text{comb},p}^R = \min\{\hat{h}_{\text{diff},p}^R, \hat{h}_{\text{sum},p}^R\}.$$

3 Illustration of Methods

We illustrate the functionalities of our R functions employing *exactly* the same examples as in [Cattaneo, Jansson, and Ma \[2017b\]](#). We refer the reader to that paper for details regarding the dataset and empirical application. The dataset is `rddenisty_senate.csv`, and the variable of interest is *margin* (ranging from -100 to 100). The cutoff is $\bar{x} = 0$.

First we load the database and present summary statistics:

```
> data("rddensity_senate")
> summary(margin)
   Min.   1st Qu.   Median     Mean  3rd Qu.    Max.
-100.000 -12.210    2.166    7.171   22.770   100.000
```

The dataset has a total of 1,390 observations, with an average Democratic party’s margin of victory of about 7 percentage points.

We now conduct a Manipulation Test using the command `rddensity()` with default options.

```
> summary(rddensity(X = margin))
```

```
RD Manipulation Test using local polynomial density estimation.
```

```
Number of obs =      1390
Model =          unrestricted
Kernel =         triangular
BW method =      comb
```

```

VCE method =          jackknife

Cutoff c = 0          Left of c          Right of c
Number of obs        640                750
Eff. Number of obs   408                460
Order est. (p)       2                  2
Order bias (q)       3                  3
BW est. (h)          19.841             27.119

Method               T                  P > |T|
Robust               -0.8753            0.3814

```

The output of the command `rddensity()` contains a variety of useful information. The first part gives basic summary statistics on the data being used, separate for control ($X_i < \bar{x}$) and treatment units ($X_i \geq \bar{x}$). This part also reports the value of the bandwidth(s) chosen. Second, the middle part includes general information regarding the overall sample size and implementation choices of the manipulation test. Finally, the lower part reports the results from implementing the manipulation test. In this first execution, the test statistic is constructed using a $q = 3$ polynomial, with different bandwidths chosen for an unrestricted model with polynomial order $p = 2$. Specifically, the bandwidth choices are: $(\hat{h}_{1,\text{comb},p}, \hat{h}_{\text{r},\text{comb},p}) = (19.841, 27.119)$, leading to effective sample sizes of $N_- = 408$ and $N_+ = 460$ for control and treatment groups, respectively. The final manipulation test is $T_q(\hat{h}_{1,\text{comb},p}, \hat{h}_{\text{r},\text{comb},p}) = -0.8753$, with a p-value of 0.3814. Therefore, in this application there is no statistical evidence of systematic manipulation of the running variable.

To further illustrate some of the capabilities of our main function `rddensity()` we consider a few additional runs. We can obtain another test statistic, [Conventional](#), by the following:

```

> summary(rddensity(X = margin, all=TRUE))

RD Manipulation Test using local polynomial density estimation.

Number of obs =      1390
Model =          unrestricted
Kernel =        triangular
BW method =      comb
VCE method =     jackknife

Cutoff c = 0          Left of c          Right of c
Number of obs        640                750
Eff. Number of obs   408                460
Order est. (p)       2                  2
Order bias (q)       3                  3
BW est. (h)          19.841             27.119

Method               T                  P > |T|
Conventional         -1.6506            0.0988
Robust               -0.8753            0.3814

```

The first statistic, labeled `Conventional` does not have bias correction or under-smoothing, and may exhibit some asymptotic bias and is therefore not recommended in general. It is included here only for completeness.

The following output showcases some other features of `rddensity()`. Here we conduct a manipulation test using the restricted model and plug-in standard errors:

```
> summary(rddensity(X = margin, fitselect="restricted", vce="plugin"))
```

```
RD Manipulation Test using local polynomial density estimation.
```

Number of obs =	1390	
Model =	restricted	
Kernel =	triangular	
BW method =	comb	
VCE method =	plugin	
Cutoff c = 0	Left of c	Right of c
Number of obs	640	750
Eff. Number of obs	396	362
Order est. (p)	2	2
Order bias (q)	3	3
BW est. (h)	18.753	18.753
Method	T	P > T
Robust	-1.4768	0.1397

In this case, because the restricted model is being used, a common bandwidth for both control and treatment units is selected. This estimated bandwidth is 18.753, and employing now the plug-in standard errors estimator (instead of the jackknife method, as used before). This empirical finding shows that we continue to not reject the null hypothesis of no manipulation (p-value is 0.1397), even when the restricted model is used, which provides further empirical evidence in favor of the validity of the RD design in this application.

To visualize, we also provide a function, `rdplotdensity()`, which performs density plot. To obtain the default manipulation test plot, all it is needed is to pass both the `rddensity` estimation result and the original data to the function:

```
> rdd <- rddensity(X = margin)
> rdplotdensity(rdd, margin)
```

The resulting default plot is given in Figure 1. The basic manipulation test plot can be improved using user-specified options. For example, the following command changes its legends and general appearance. The result is given in Figure 2

```
> rdplotdensity(rdd, margin, lcol = c("black", "black"), xlabel = "margin",
+               plotRange = c(-50, 50), plotN = 100)
```

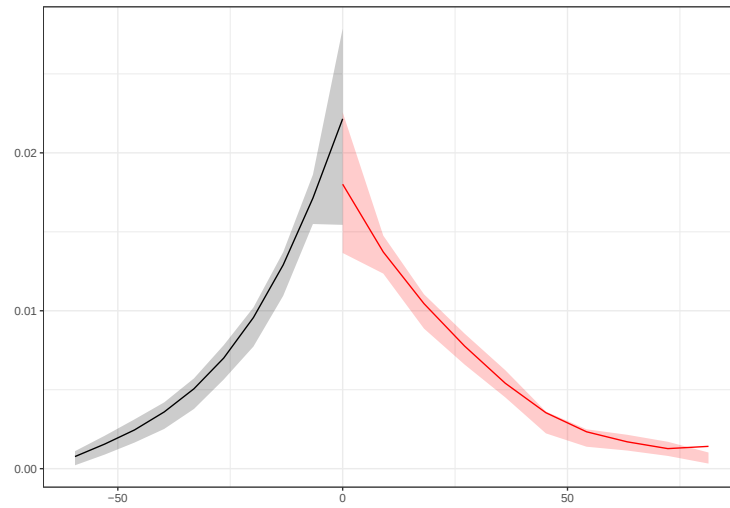


Figure 1. Manipulation Test Plot (default options).

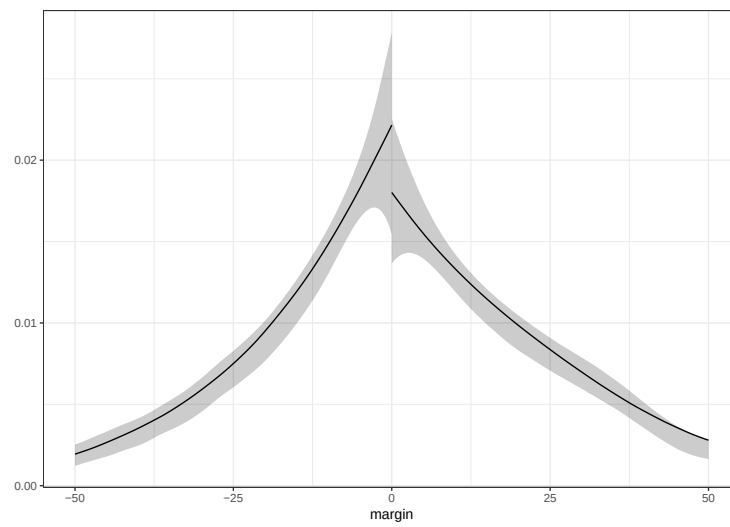


Figure 2. Manipulation Test Plot (with user options).

To close this section, we also report an output from the companion command `rdbwdensity()`. This command was used all along implicitly by `rddensity()`, but here we employ it as a stand-alone command to illustrate some of its features. The default output for the empirical applications is as follows:

```
> summary(rdbwdensity(X = margin))
```

Bandwidth selection for manipulation testing.

Number of obs =	1390		
Model =	unrestricted		
Kernel =	triangular		
VCE method =	jackknife		

Cutoff c = 0	Left of c	Right of c	
Number of obs	640	750	
Min Running var.	-100	0.011	
Max Running var.	-0.079	100	
Order est. (p)	2	2	

Target	Bandwidth	Variance	Bias^2
left density	19.8411	0.109	0
right density	27.5688	0.0853	0
diff. densities	27.1188	0.1944	0
sum densities	19.5312	0.1944	0

The output of `rdbwdensity()` mimics as close as possible the one from `rddensity()`. Notice that the defaults are all the same. The main results are reported in the lower panel, where now the output includes different estimated bandwidth choices. These choices depend on the MSE criterion function (and the model considered, as discussed previously): the first row (labeled “left density”) reports $\hat{h}_{1,p}$, the second row (labeled “right density”) reports $\hat{h}_{r,p}$, the third row (labeled “difference densities”) reports $\hat{h}_{\text{diff},p}$, and the fourth row (labeled “sum densities”) reports $\hat{h}_{\text{sum},p}$. Of course, $\hat{h}_{\text{comb},p}$ may be easily constructed using the above information.

A similar set of results may also be obtained for the restricted model. For this case the command is `rdbwdensity(X = margin, fitselect = "restricted")`, but we do not report these results here to conserve space.

Finally, we briefly illustrate how the two commands can be combined:

```
> tmp <- rdbwdensity(X = margin)
> hr <- tmp$h[2,1]
> summary(rddensity(X = margin, h=c(10,hr)))
```

RD Manipulation Test using local polynomial density estimation.

Number of obs =	1390
Model =	unrestricted

Kernel =	triangular	
BW method =	mannual	
VCE method =	jackknife	
Cutoff c = 0	Left of c	Right of c
Number of obs	640	750
Eff. Number of obs	251	464
Order est. (p)	2	2
Order bias (q)	3	3
BW est. (h)	10	27.569
Method	T	P > T
Robust	-1.0331	0.3016

In this last example, first `rdbwdensity()` is used to estimate the bandwidths quietly, but then the left bandwidth is set manually ($h_1 = 10$) while the right bandwidth is estimated ($h_r = 27.569$) when executing `rddensity()`.

The companion replication file [rddensity_illustration](#) includes the syntax of all the examples discussed above, as well as additional examples not included here to conserve space. These examples are:

1. `summary(rddensity(X = margin, kernel = "uniform"))`
Manipulation testing using uniform kernel.
2. `summary(rddensity(X = margin, bwselect = "diff"))`
Manipulation testing with bandwidth selection based on MSE of difference of densities.
3. `summary(rddensity(X = margin, h = c(10,15)))`
Manipulation testing using user-chosen bandwidths (10, 15).
4. `summary(rddensity(X = margin, p = 2, q = 4))`
Manipulation testing using $p = 2$ and $q = 4$.
5. `summary(rddensity(X = margin, c = 5, all = TRUE))`
Manipulation testing at cutoff 5 with all statistics.
6. `summary(rdbwdensity(X = margin, p = 3, fitselect = "restricted"))`
Bandwidth selection for restricted model with $p = 3$.
7. `summary(rdbwdensity(X = margin, kernel = "uniform", vce = "jackknife"))`
Bandwidth selection with uniform kernel and jackknife variance estimator.

References

- Matias D. Cattaneo, Michael Jansson, and Xinwei Ma. Simple local polynomial density estimators. working paper, University of Michigan, 2017a.
- Matias D. Cattaneo, Michael Jansson, and Xinwei Ma. `rddensity`: Manipulation testing based on density discontinuity. working paper, University of Michigan, 2017b.