

`rdmulti`: RD Designs with multiple cutoffs or multiple scores*

Matias D. Cattaneo[†]

Rocio Titiunik[‡]

Gonzalo Vazquez-Bare[§]

December 30, 2020

Abstract

We introduce the **R** (and **Stata**) package `rdmulti`, which includes three commands (`rdmc`, `rdmcplot`, `rdms`) for analyzing Regression Discontinuity (RD) designs with multiple cutoffs or multiple scores. The command `rdmc` applies to non-cumulative and cumulative multi-cutoff RD settings. It calculates pooled and cutoff-specific RD treatment effects, and provides robust bias-corrected inference procedures. Post estimation and inference is allowed. The command `rdmcplot` offers RD plots for multi-cutoff settings. Finally, the command `rdms` concerns multi-score settings, covering in particular cumulative cutoffs and two running variables contexts. It also calculates pooled and cutoff-specific RD treatment effects, provides robust bias-corrected inference procedures, and allows for post-estimation estimation and inference. These commands employ the **Stata** (and **R**) package `rdrobust` for plotting, estimation, and inference. Companion **R** functions with the same syntax and capabilities are provided.

Keywords: regression discontinuity designs, quasi-experimental techniques, causal inference, multiple cutoffs, multiple scores.

*We thank Sebastian Calonico and Nicolas Idrobo for helpful comments and discussions.

[†]Department of Operations Research and Financial Engineering, Princeton University.

[‡]Department of Politics, Princeton University.

[§]Department of Economics, UC Santa Barbara.

Contents

1	Introduction	1
2	Illustration of Methods.....	1
2.1	Multiple non-cumulative cutoffs using <code>rdmc</code>	1
2.2	Cumulative cutoffs using <code>rdms</code>	5
2.3	Bivariate score using <code>rdms</code>	7

1 Introduction

This article illustrates the R package `rdmulti`, which analyzes RD designs with multiple cutoffs or scores. The functions in this package have the same syntax and offer the same functionalities as our companion `Stata` commands described in Cattaneo, Titiunik, and Vazquez-Bare [forthcoming].

For brevity, we focus exclusively on software implementation issues. Extensive discussion and details on methodological and practical aspects can be found in Cattaneo, Keele, Titiunik, and Vazquez-Bare [2016] and Cattaneo, Idrobo, and Titiunik [2018a,b].

For help on the functions' syntax and related issues, please refer to [reference manual](#). For related `Stata` and R packages useful for analysis of RD designs, visit:

<https://rdpackages.github.io/>

2 Illustration of Methods

We illustrate how to implement the functions described above using simulated data. This section replicates, as close as possible, section 5 of Cattaneo, Titiunik, and Vazquez-Bare [forthcoming].

First, to install the `rdmulti` package, type:

```
install.packages("rdmulti")
```

2.1 Multiple non-cumulative cutoffs using `rdmc`

We begin by illustrating `rdms` using a simulated dataset, `simdata_multic`. In this dataset, `y` is the outcome variable, `x` is the running variable and `c` is a variable indicating the cutoff that each unit in the sample faces. As shown below, there are four different cutoffs, each with the sample size.

```
> data <- read.csv('simdata_multic.csv')
> Y <- data$y
> X <- data$x
> C <- data$c
```

We begin by illustrating `rdmc` using a simulated dataset, `simdata_multic`. In this dataset, `y` is the outcome variable, `x` is the running variable and `c` is a variable indicating the cutoff that each unit in the sample faces. As shown below, there are four different cutoffs, each with the sample size.

```
> aux <- rdmc(Y,X,C)
```

```
=====
Cutoff      Coef.   P-value      95% CI      hl      hr      Nh      Weight
```

33.00	484.831	0.000	421.184	552.531	14.662	14.662	289	0.540
66.00	297.981	0.000	220.348	362.268	11.952	11.952	246	0.460

Weighted	398.915	0.000	348.742	445.141	.	.	535	.
Pooled	436.400	0.001	179.336	676.631	13.684	13.684	550	.
=====								

The output shows the cutoff-specific estimate at each cutoff, together with the corresponding robust bias-corrected p-value, 95 percent robust confidence interval and sample size at each cutoff, and two “global” estimates. The first one is a weighted average of the cutoff specific estimates using the estimated weights described in [Cattaneo, Titiunik, and Vazquez-Bare \[forthcoming\]](#). These estimated weights are shown in the last column. The second one is the pooled estimate obtained by normalizing the running variable. While these two estimators converge to the same population parameter, they can differ in finite samples as seen in this example. In this example, the effect is statistically insignificant at the lowest cutoff, with evidence of a positive treatment effect at the remaining three cutoffs. Given the distribution of the running variable and the sample size at each cutoff, the weighted and pooled estimates give approximately the same weight to each cutoff.

All the results in the above display are calculated using `rdrobust`. The user can specify options for `rdrobust` to calculate both the pooled estimates using the option `pooled_opt`. For instance, the syntax below specifies a bandwidth of 20 and a local quadratic polynomial for the pooled estimand. By default, `rdmc` omits the output from `rdrobust` when estimating the effects. The output from the pooled effect estimation can be displayed using the option `verbose`, which we use below to show how the options are passed to `rdrobust`.

```
> aux <- rdmc(Y,X,C,pooled_opt=paste('h=20','p=2',sep=','),verbose=TRUE)
```

```
Call: rdrobust
```

```
Number of Obs.      2000
BW type             Manual
Kernel              Triangular
VCE method          NN
```

```
Number of Obs.      968      1032
Eff. Number of Obs. 409      416
Order est. (p)       2        2
Order bias (q)       3        3
BW est. (h)          20.000    20.000
BW bias (b)          20.000    20.000
rho (h/b)            1.000     1.000
Unique Obs.          968      1032
```

=====				
Method	Coef.	Std. Err.	z	P> z
				[95% C.I.]
=====				

```

Conventional  437.042  129.798  3.367  0.001  [182.643 , 691.441]
Robust        -      -      3.012  0.003  [185.618 , 877.381]
=====

```

```

=====
Cutoff      Coef.  P-value      95% CI      hl      hr      Nh  Weight
=====
33.00      484.831  0.000      421.184  552.531  14.662  14.662  289  0.540
66.00      297.981  0.000      220.348  362.268  11.952  11.952  246  0.460
-----
Weighted    398.915  0.000      348.742  445.141      .      .    535      .
Pooled      437.042  0.003      185.618  877.381  20.000  20.000  825      .
=====

```

The following syntax shows how to manually change options for the cutoff-specific estimates.

```
> aux <- rdmc(Y,X,C,h=c(11,10))
```

```

=====
Cutoff      Coef.  P-value      95% CI      hl      hr      Nh  Weight
=====
33.00      495.429  0.000      368.129  563.215  11.000  11.000  207  0.498
66.00      303.769  0.000      220.396  403.320  10.000  10.000  209  0.502
-----
Weighted    399.138  0.000      321.557  455.233      .      .    416      .
Pooled      436.400  0.001      179.336  676.631  13.684  13.684  550      .
=====

```

The command `rdmcplot` jointly plots the estimated regression functions at each cutoff. The basic syntax is the following:

```
> aux <- rdmcplot(Y,X,C)
```

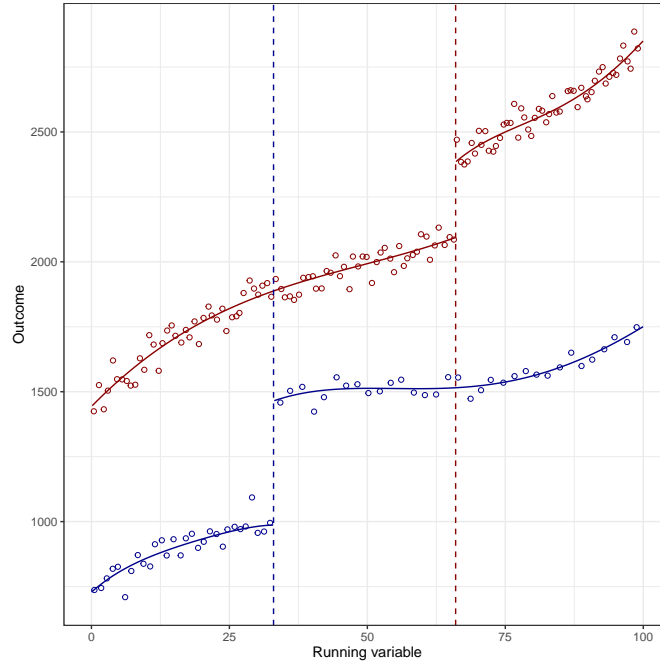


Figure 1. Multiple RD plot.

Finally, the plot can be restricted to a bandwidth using the option `h` and to use a polynomial of a specified order using the option `p`, as shown below. This option allows the user to plot the linear fit and estimated treatment effects at each cutoff.

```
aux <- rdmcpplot(Y,X,C,h=c(11,12),pvec=c(1,1))
```

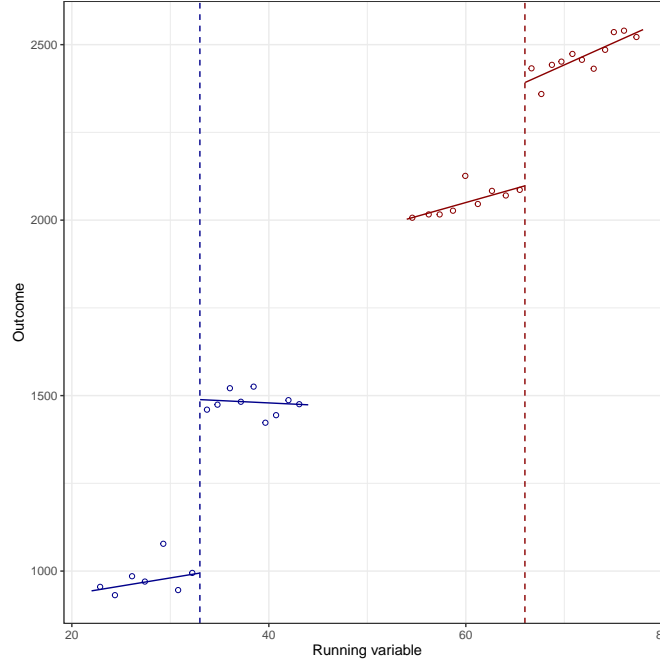


Figure 2. Multiple RD plot.

2.2 Cumulative cutoffs using rdms

We now illustrate the use of `rdms` for cumulative cutoffs using the simulated dataset `simdata_cumul`. In this dataset, the running variable ranges from 0 to 100, and units with running variable below 33 receive a certain treatment level d_1 whereas units with running variable above 66 receive another treatment level d_2 . In this setting, the cutoffs are indicated as a short variable in the dataset, where each row indicates a cutoff.

```
> aux <- rdmcplot(Y,X,C)
> data <- read.csv('simdata_cumul.csv')
> Y <- data$y
> X <- data$x
> cvec <- c(data$c[1],data$c[2])
```

The syntax for cumulative cutoffs is similar to `rdmc`. The user specifies the outcome variable, the running variable and the cutoffs as follows:

```
> aux <- rdms(Y,X,cvec)
```

Cutoff	Coef.	P-value	95% CI		hl	hr	Nh
33.00	395.492	0.000	363.761	423.856	15.109	15.109	286
66.00	342.872	0.000	315.952	373.959	12.221	12.221	265

Options like the bandwidth, the bandwidth and the kernel for each cutoff-specific effect can be specified as shown below.

```
> aux <- rdms(Y,X,cvec,h=c(11,8),kernel=c('uniform','triangular'))
```

Cutoff	Coef.	P-value	95% CI		hl	hr	Nh
33.00	394.470	0.000	351.649	438.725	11.000	11.000	215
66.00	342.505	0.000	301.556	375.947	8.000	8.000	166

Without further information, the `rdms` command could be using any observation above the cutoff 33 to estimate the effect of the first treatment level d_1 . This implies that some observations in the range $[66, 100]$ are used. But these observations receive the second treatment level, d_2 . This feature can result in inconsistent estimators for τ_1 . To avoid this problem, the user can specify the range of observations to be used around each cutoff. In this case, we can restrict the range at the first cutoff (33) to go from 0 to 65.5, to ensure that no observations above 66 are used, and the range at the second cutoff (66) to go from 33.5 to 100. This can be done as follows.

```
> aux <- rdms(Y,X,cvec,range = matrix(c(0,33.5,65.5,100),ncol=2))
```

Cutoff	Coef.	P-value	95% CI		hl	hr	Nh
33.00	394.698	0.000	356.124	430.445	10.960	10.960	214
66.00	342.180	0.000	312.205	372.042	11.176	11.176	246

The pooled estimate can be obtained using `rdmc`. For this, we need to assign each unit in the sample a value for the cutoff. One possibility is to assign each unit to the closest cutoff. For this, we generate a variable named `cutoff` that equals 33 for units with score below 49.5 (the middle point between 33 and 66), and equals 66 for units above 49.5.

```
> cutoff <- cvec[1]*(X<=49.5) + cvec[2]*(X>49.5)
> aux <- rdmc(Y,X,cutoff)
```

Cutoff	Coef.	P-value	95% CI		hl	hr	Nh	Weight
33.00	389.528	0.000	332.937	443.686	6.261	6.261	119	0.531
66.00	341.015	0.000	300.389	377.332	5.042	5.042	105	0.469
Weighted	366.788	0.000	330.626	399.637	.	.	224	.
Pooled	363.968	0.000	180.111	551.784	8.137	8.137	333	.

Finally, we can use the variable `cutoff` to plot the regression functions using `rdmcplot`.

```
aux <- rdmcplot(Y,X,cutoff,col_bins=rep('darkblue',2),col_xline=rep('darkblue',2),nopoly=TRUE)
```

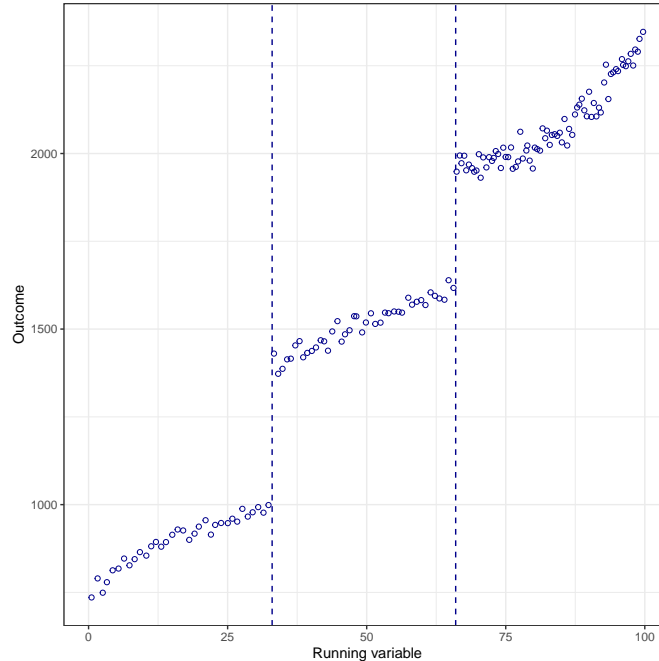


Figure 3. Multiple RD plot.

2.3 Bivariate score using `rdms`

We now illustrate the use of `rdms` to analyze RD designs with two running variables using the simulated dataset `simdata_multis`. In this dataset, there are two running variables, `x1` and `x2`, ranging between 0 and 100, and units receive the treatment when $x1 \leq 50$ and $x2 \leq 50$. We look at three cutoffs on the boundary: (25,50), (50,50) and (50,25).

```
> data <- read.csv('simdata_multis.csv')
> Y <- data$y
> X1 <- data$x1
> X2 <- data$x2
> zvar <- data$t
> cvec <- c(data$c1[1],data$c1[2],data$c1[3])
> cvec2 <- c(data$c2[1],data$c2[2],data$c2[3])
```

The basic syntax is the following:

```
> aux <- rdms(Y,X1,cvec,X2,zvar,cvec2)
```

```
=====
```

Cutoff	Coef.	P-value	95% CI		hl	hr	Nh
(25.00,50.00)	243.842	0.111	-50.929	491.180	11.122	11.122	42
(50.00,50.00)	578.691	0.000	410.825	764.877	13.831	13.831	47
(50.00,25.00)	722.444	0.000	451.493	1060.149	10.834	10.834	38

Information to estimate each cutoff-specific estimate can be provided as illustrated before. For instance, to specify cutoff-specific bandwidths:

```
> aux <- rdms(Y,X1,cvec,X2,zvar,cvec2,h=c(15,13,17))
```

Cutoff	Coef.	P-value	95% CI		hl	hr	Nh
(25.00,50.00)	336.121	0.233	-119.352	491.365	15.000	15.000	87
(50.00,50.00)	583.047	0.000	501.940	1101.236	13.000	13.000	42
(50.00,25.00)	620.692	0.000	464.920	1159.992	17.000	17.000	86

Finally, the `xnorm` option allows the user to specify the normalized running variable to calculate a pooled estimate. In this case, we define the normalized running variable as the closest perpendicular distance to the boundary defined by the treatment assignment.

```
> xnorm <- apply(cbind(abs(50-X1),abs(50-X2)),1,min)*(2*zvar-1)
> aux <- rdms(Y,X1,cvec,X2,zvar,cvec2,xnorm=xnorm)
```

Cutoff	Coef.	P-value	95% CI		hl	hr	Nh
(25.00,50.00)	243.842	0.111	-50.929	491.180	11.122	11.122	42
(50.00,50.00)	578.691	0.000	410.825	764.877	13.831	13.831	47
(50.00,25.00)	722.444	0.000	451.493	1060.149	10.834	10.834	38
Pooled	447.017	0.000	389.330	496.852	12.726	12.726	433

References

- Matias D. Cattaneo, Luke Keele, Rocio Titiunik, and Gonzalo Vazquez-Bare. Interpreting regression discontinuity designs with multiple cutoffs. *Journal of Politics*, 78(4):1229–1248, 2016.
- Matias D. Cattaneo, Nicolás Idrobo, and Rocío Titiunik. *A Practical Introduction to Regression Discontinuity Designs: Foundations*. Cambridge Elements: Quantitative and Computational Methods for Social Science, Cambridge University Press, 2018a.

Matias D. Cattaneo, Nicolás Idrobo, and Rocío Titiunik. *A Practical Introduction to Regression Discontinuity Designs: Extensions*. Cambridge Elements: Quantitative and Computational Methods for Social Science, Cambridge University Press, 2018b.

Matias D. Cattaneo, Rocio Titiunik, and Gonzalo Vazquez-Bare. Analysis of regression discontinuity designs with multiple cutoffs or scores. *Stata Journal*, forthcoming.