

STAT 209 Weeks 5 Regression Discontinuity Designs

Example from Package 'rdd'

Maintainer Drew Dimmery <drewd@nyu.edu> Title Regression Discontinuity Estimation
 Description This package provides the tools to undertake estimation in Regression Discontinuity Designs. Both sharp and fuzzy designs are supported. Estimation is accomplished using local linear R version 3.0.1 (2013-05-16) -- "Good Sport"

```
> install.packages("rdd")
> library(rdd)
```

install and load

```
## Artificial data example from p.9 rdd manual
> x<-runif(1000,-1,1)
> cov<-rnorm(1000) # extra auxiliary variable
> y<-3+2*x+3*cov+10*(x>=0)+rnorm(1000)
# example builds in a treatment effect of 10 points for those selected on x (x>0)
# story? students with high (or higher) ability selected for enriched instruction
```

selection var
form outcome

run the rdd function just using X

```
> RDestimate(y~x)
Call: RDestimate(formula = y ~ x)
Coefficients:
```

local ATE

	LATE	Half-BW	Double-BW
	9.821	9.599	10.005

```
> summary(RDestimate(y~x))
Call: RDestimate(formula = y ~ x)
Type: sharp
Estimates:
```

	Bandwidth	Observations	Estimate	Std. Error	z value	Pr(> z)	
LATE	0.7613	721	<u>9.821</u>	0.5264	18.65	1.157e-77	***
Half-BW	0.3807	348	<u>9.599</u>	0.7705	12.46	1.258e-35	***
Double-BW	1.5227	1000	<u>10.005</u>	0.4134	24.20	2.001e-129	***

F-statistics:

	F	Num. DoF	Denom. DoF	p
LATE	785.6	3	717	0
Half-BW	341.9	3	344	0
Double-BW	1260.4	3	996	0

```
> plot(RDestimate(y~x))
```

#compare with our simple ancova approach

```
> rubin = lm(y ~ (x>0) + x)
> summary(rubin)
Call: lm(formula = y ~ (x > 0) + x)
Coefficients:
```

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	3.1168	0.2227	13.998	< 2e-16 ***
x > 0TRUE	<u>10.0795</u>	<u>0.4038</u>	24.960	< 2e-16 ***
x	<u>1.9951</u>	<u>0.3371</u>	5.919	4.45e-09 ***

Residual standard error: 3.046 on 997 degrees of freedom
 Multiple R-squared: 0.8017, Adjusted R-squared: 0.
 F-statistic: 2015 on 2 and 997 DF, p-value: < 2.2e-16

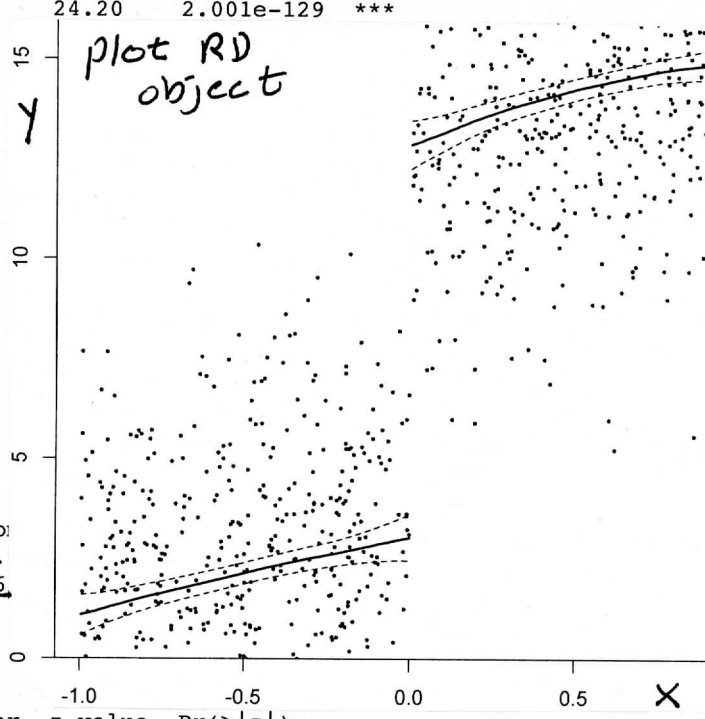
Use extra info (w=cov)

```
> summary(RDestimate(y~x | cov))
Call: RDestimate(formula = y ~ x | cov) Type: sharp
Estimates:
```

	Bandwidth	Observations	Estimate	Std. Error	z value	Pr(> z)	
LATE	0.7613	721	<u>9.954</u>	0.1831	54.36	0.000e+00	***
Half-BW	0.3807	348	<u>10.005</u>	0.2837	35.27	1.621e-272	***
Double-BW	1.5227	1000	<u>9.989</u>	0.1389	71.94	0.000e+00	***

```
> rubincov = lm(y ~ (x>0) + x + cov)
> summary(rubincov)
Call: lm(formula = y ~ (x > 0) + x + cov)
Coefficients:
```

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	3.04203	0.07480	40.67	<2e-16 ***
x > 0TRUE	<u>9.99862</u>	<u>0.13567</u>	73.70	<2e-16 ***
x	<u>1.99487</u>	<u>0.11323</u>	17.62	<2e-16 ***
cov	2.97875	0.03365	88.53	<2e-16 ***



```

Example from Package 'rdd'
Maintainer Drew Dimmery <drewd@nyu.edu> Title Regression Discontinuity Estimation
Description This package provides the tools to undertake estimation in Regression Discontinuity
Designs. Both sharp and fuzzy designs are supported. Estimation is accomplished using local linear
R version 3.0.1 (2013-05-16) -- "Good Sport"
> install.packages("rdd")
> library(rdd)
      ## Artificial data example from p.9 rdd manual
> x<-runif(1000,-1,1)
> cov<-rnorm(1000) # extra auxiliary variable
> y<-3+2*x+3*cov+10*(x>=0)+rnorm(1000)
# example builds in a treatment effect of 10 points for those selected on x (x>0)
# story? students with high (or higher) ability selected for enriched instruction

# run the rdd function just using X
> RDestimate(y~x)
Call: RDestimate(formula = y ~ x)
Coefficients:
      LATE      Half-BW      Double-BW
    9.821      9.599      10.005
> summary(RDestimate(y~x))
Call: RDestimate(formula = y ~ x)
Type: sharp
Estimates:
      Bandwidth  Observations  Estimate  Std. Error  z value  Pr(>|z|)
LATE      0.7613           721         9.821     0.5264     18.65    1.157e-77 ***
Half-BW   0.3807           348         9.599     0.7705     12.46    1.258e-35 ***
Double-BW 1.5227           1000        10.005     0.4134     24.20    2.001e-129 ***
---
F-statistics:
      F          Num. DoF  Denom. DoF  p
LATE      785.6      3          717        0
Half-BW   341.9      3          344        0
Double-BW 1260.4     3          996        0
> plot(RDestimate(y~x))

#compare with our simple ancova approach
> rubin = lm(y ~ (x>0) + x)
> summary(rubin)
Call: lm(formula = y ~ (x > 0) + x)
Coefficients:
      Estimate Std. Error t value Pr(>|t|)
(Intercept)  3.1168    0.2227  13.998 < 2e-16 ***
x > 0TRUE    10.0795    0.4038  24.960 < 2e-16 ***
x            1.9951    0.3371   5.919 4.45e-09 ***
---
Residual standard error: 3.046 on 997 degrees of freedom
Multiple R-squared: 0.8017, Adjusted R-squared: 0.8013
F-statistic: 2015 on 2 and 997 DF, p-value: < 2.2e-16

> summary(RDestimate(y~x| cov))
Call: RDestimate(formula = y ~ x | cov) Type: sharp
Estimates:
      Bandwidth  Observations  Estimate  Std. Error  z value  Pr(>|z|)
LATE      0.7613           721         9.954     0.1831     54.36    0.000e+00 ***
Half-BW   0.3807           348         10.005     0.2837     35.27    1.621e-272 ***
Double-BW 1.5227           1000         9.989     0.1389     71.94    0.000e+00 ***
---
> rubincov = lm(y ~ (x>0) + x +cov)
> summary(rubincov)
Call: lm(formula = y ~ (x > 0) + x + cov)
Coefficients:
      Estimate Std. Error t value Pr(>|t|)
(Intercept)  3.04203    0.07480   40.67 <2e-16 ***
x > 0TRUE    9.99862    0.13567   73.70 <2e-16 ***
x            1.99487    0.11323   17.62 <2e-16 ***

```

