ECON675 – Assignment 6

Anirudh Yadav

December 3, 2018

Contents

1 Continuity-based identification in SRD designs

1.1 RD treatment effect with a single cutoff

We have

$$\tau_{\texttt{SRD}} = \lim_{\epsilon \to 0^+} \mathbb{E}[Y_i | \tilde{X}_i = \epsilon] - \lim_{\epsilon \to 0^+} \mathbb{E}[Y_i | \tilde{X}_i = -\epsilon]$$

Using the definition $\tilde{X}_i = X_i - C_i$ and the single cutoff assumption, $\mathbb{P}[C_i = c] = 1$, gives

$$\begin{aligned}
\tau_{\texttt{SRD}} &= \lim_{\epsilon \to 0^+} \mathbb{E}[Y_i | X_i = c + \epsilon] - \lim_{\epsilon \to 0^+} \mathbb{E}[Y_i | X_i = c - \epsilon] \\
&= \lim_{\epsilon \to 0^+} \mathbb{E}[Y_{1i}(c) T_i + Y_{0i}(c)(1 - T_i) | X_i = c + \epsilon] - \lim_{\epsilon \to 0^+} \mathbb{E}[Y_{1i}(c) T_i + Y_{0i}(c)(1 - T_i) | X_i = c - \epsilon] \\
&= \lim_{\epsilon \to 0^+} \mathbb{E}[Y_{1i}(c) | X_i = c + \epsilon] - \lim_{\epsilon \to 0^+} \mathbb{E}[Y_{0i}(c) | X_i = c - \epsilon], \text{ using } \mathbf{S3} \\
&= \mathbb{E}[Y_{1i}(c) - Y_{0i}(c) | X_i = c],
\end{aligned}$$

as required.

1.2 RD treatment effect with multiple cutoffs

Now, with multiple cutoffs

$$\tau_{\texttt{SRD}} = \lim_{\epsilon \to 0^+} \mathbb{E}[Y_i | \tilde{X}_i = \epsilon] - \lim_{\epsilon \to 0^+} \mathbb{E}[Y_i | \tilde{X}_i = -\epsilon]$$

Focusing on the first term on the RHS:

$$\begin{aligned}
\lim_{\epsilon \to 0^+} \mathbb{E}[Y_i | \tilde{X}_i = \epsilon] &= \lim_{\epsilon \to 0^+} \mathbb{E}[Y_{1i}(C_i) T_i + Y_{0i}(C_i)(1 - T_i) | X_i = C_i + \epsilon] \\
&= \mathbb{E}[Y_{1i}(C_i) | X_i = C_i] \text{ using } \mathbf{S3}, \\
&= \sum_{c \in \mathcal{C}} \mathbb{E}[Y_{1i}(C_i) | X_i = c, C_i = c] \mathbb{P}[X_i = c, C_i = c], \text{ since } C_i \text{ is a discrete r.v}, \\
&= \sum_{c \in \mathcal{C}} \mathbb{E}[Y_{1i}(C_i) | X_i = c, C_i = c] \frac{f_{X|C}(c|c) \mathbb{P}[C_i = c]}{\sum_{c \in \mathcal{C}} f_{X|C}(c|c) \mathbb{P}[C_i = c]}.
\end{aligned}$$

An analogous derivation shows that

$$\lim_{\epsilon \to 0^+} \mathbb{E}[Y_i | \tilde{X}_i = -\epsilon] = \sum_{c \in \mathcal{C}} \mathbb{E}[Y_{0i}(C_i) | X_i = c, C_i = c] \frac{f_{X|C}(c|c) \mathbb{P}[C_i = c]}{\sum_{c \in \mathcal{C}} f_{X|C}(c|c) \mathbb{P}[C_i = c]},$$

which proves the desired result. In this case, $\tau_{\texttt{SRD}}$ is just a weighted average of the RD treatment effects for the different cutoff levels.

1.3 RD treatment effect with individual heterogeneity

First consider the case with a single cutoff, $\mathbb{P}[C_i = c] = 1$. Suppose that W_i is a continuous random variable, with marginal cdf $G(\cdot)$, that affects i's potential outcomes and is iid across observations. Now,

$$\begin{aligned}\tau_{\texttt{SRD}} &= \lim_{\epsilon \to 0^+} \mathbb{E}[y_1(w_i)|X_i = c + \epsilon] - \lim_{\epsilon \to 0^+} \mathbb{E}[y_0(w_i)|X_i = c - \epsilon] \\ &= \mathbb{E}[y_1(w_i) - y_0(w_i)|X_i = c] \\ &= \int (y_1(w) - y_0(w)) g_{W|X}(w)|_{X=c}\end{aligned}$$

Now, using Bayes' rule

$$g_{W|X}(w) = \frac{f(X|W)g(w)}{f(X)}$$

where $f(\cdot)$ is the marginal density of the running variable X_i. Thus, for the single cutoff case

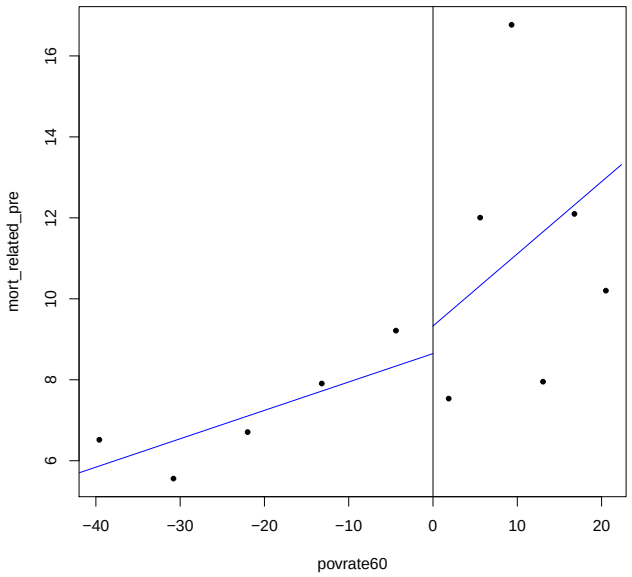
$$\tau_{\texttt{SRD}} = \int (y_1(w) - y_0(w)) \frac{f_{X|W}(c)}{f(c)} g(w) dw$$

which is the result presented by Lee (2008). Then, you can extend the multiple cutoff case to get the desired result.

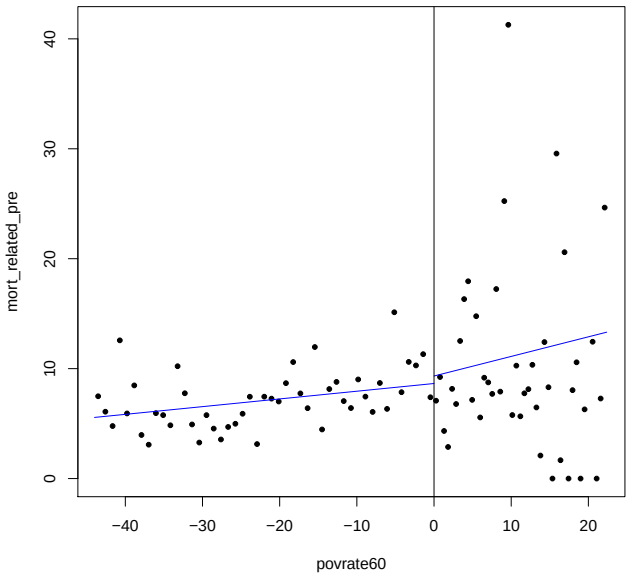
2 The effect of Head Start on child mortality

2.1 RD plots and falsification tests

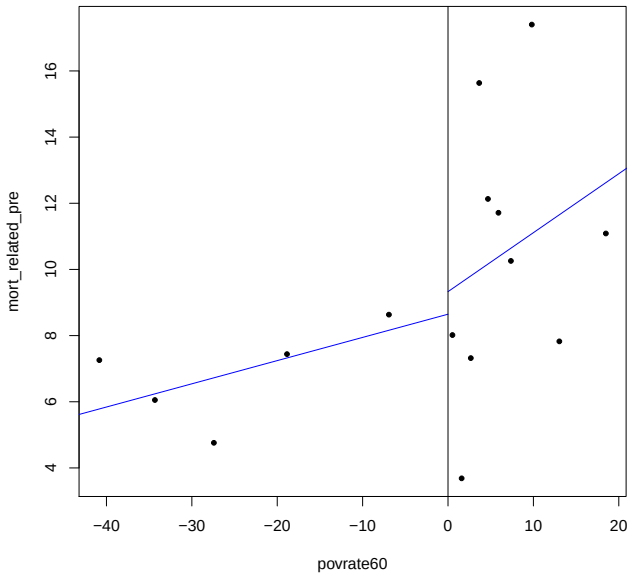
Figure 1: RD Plots of Pre-intervention Mortality Rates Using Different Binning Procedures



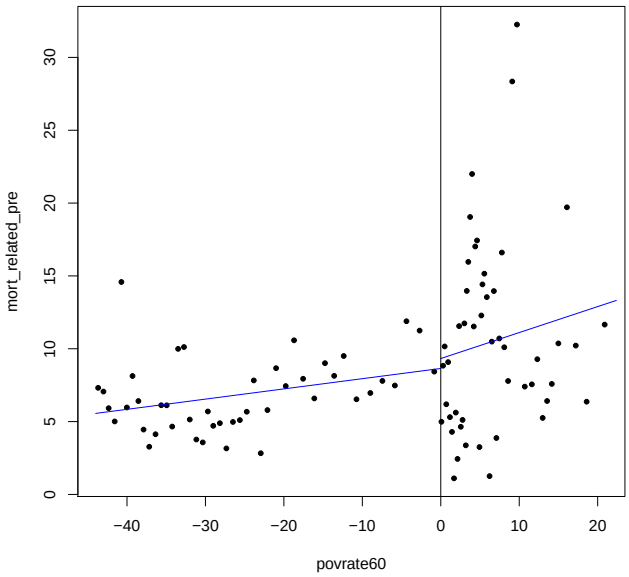
(a) Evenly-spaced, IMSE optimal



(b) Evenly-spaced, variance mimicking



(c) Quantile-spaced, IMSE optimal



(d) Quantile-spaced, variance mimicking

Figure 1 shows the RD plots of `mort_related_pre` using different binning procedures, as required. For each binning method, there is clearly no evidence of a negative discontinuity at the cutoff poverty rate. (In fact, there seems to be a small positive jump in pre-inervention mortality rates at the cutoff.) If there was a negative jump in pre-intervention mortality rates at the cutoff this would potentially falsify the proposed RD design because it would provide strong evidence that the counties assigned to Head Start treatment had systematically lower mortality rates prior to the intervention.

We can also conduct formal falsification tests. First, I conduct exact binomial tests for small windows around the cutoff. The basic idea is that if there is no systematic sorting, then the number of observations just above or below the cutoff should be pretty close to random, and thus should follow a binomial distribution. Table 1 shows the results for the exact binomial test (with probability of success set to 0.5) for a few small windows near the cutoff (replicating Table 1 in Cattaneo, *et al.* (2017)). Clearly, we cannot reject the null that the number of observations just above and below the cutoff are random.

Table 1: Binomial tests

	h	N_W^-	N_W^+	p-value
1	0.3	9	10	1.000
2	0.5	18	16	0.864
3	0.7	24	22	0.883
4	0.9	32	27	0.603
5	1.1	43	33	0.302
6	1.3	51	38	0.203

Next, I test for a discontinuity in the density of the running variable at the cutoff, as in Cattaneo, *et al.* (2017), using the `rddensity` package in `R`. I compute the density test p-value using the package defaults, which specifies a local-quadratic polynomial estimator with triangular kernel and jackknife standard errors. I get $p = 0.639$, implying that we cannot reject the null that the density of the running variable is continuous at the cutoff.

2.2 Global and flexible parametric methods – BAD!

2.2.1 Constant treatment effect

I estimate the following global polynomial regression models

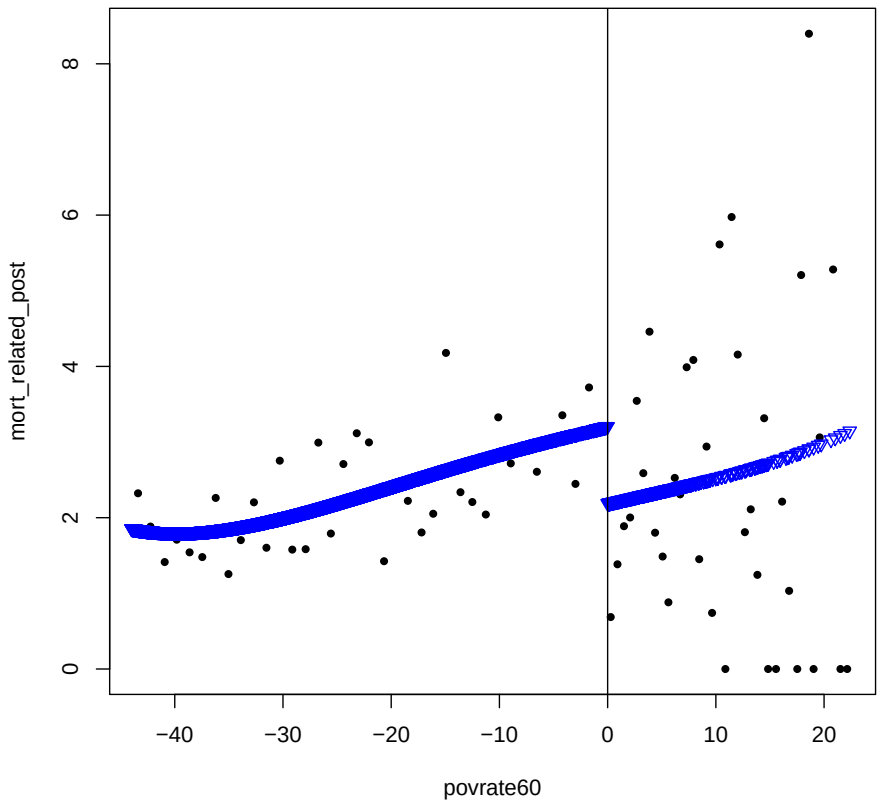
$$y_i = \alpha_0 + \tau_{\texttt{SRD}} t_i + \sum_{k=1}^{p} \beta_k x_i^k + \epsilon_i$$

where y_i is the outcome of interest (post-intervention mortality rate) and x_i is the running variable (poverty rate in 1960) and t_i is a treatment dummy equal to 1 if $x_i \geq 0$. Table 2 presents the point estimates of $\tau_{\texttt{SRD}}$ for $p = 3, 4, 5, 6$ and the corresponding robust standard errors. The estimated treatment effects are all negative as expected. I also plot the fitted values for $p = 4$ below.

Table 2: Global Polynomial Fit under Constant Treatment Effect Assumption

	$p = 3$	$p = 4$	$p = 5$	$p = 6$
Point estimate	-1.12	-1.02	-1.66	-1.75
Std. err.	0.59	0.75	0.81	0.86

Figure 2: Fitted Values for 4-th Order Global Polynomial – Constant Effect Assumption



2.2.2 Fully interacted model

Next, I estimate the 'fully-interacted' models

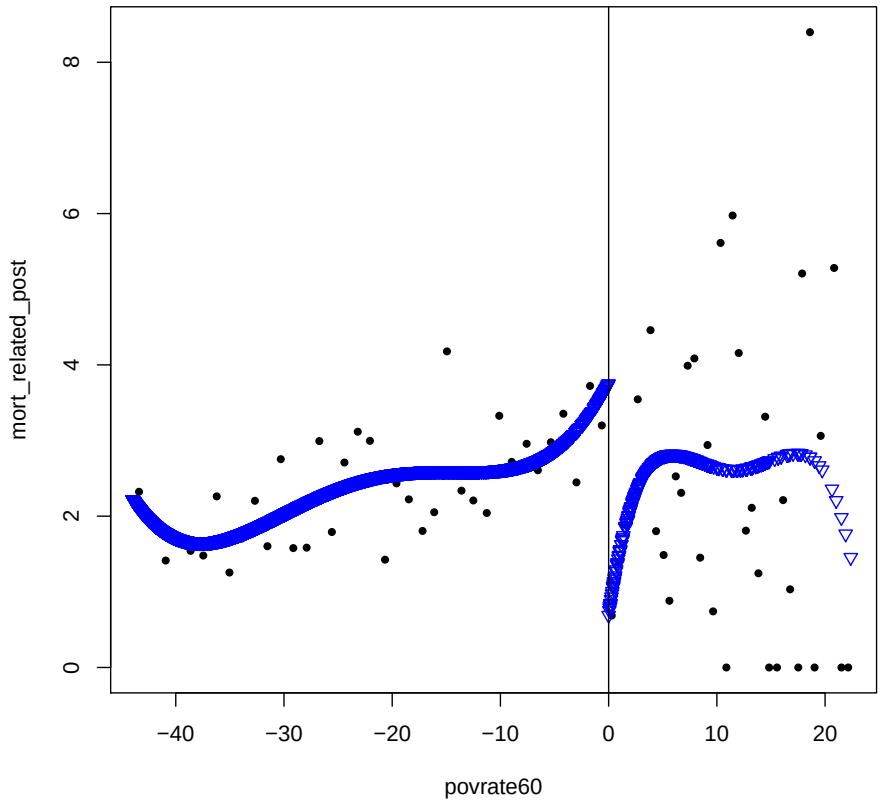
$$y_i = \alpha_0 + \tau_{\texttt{SRD}} t_i + \sum_{k=1}^{p} \beta_k t_i x_i^k + \epsilon_i.$$

These models amount to fitting a global polynomial of order p separately on both sides of the cutoff. Table 3 shows the point estimates of $\tau_{\texttt{SRD}}$ for $p = 3, 4, 5, 6$ and the corresponding robust standard errors.

Table 3: Global Polynomial Fit Separately on Both Sides of the Cutoff

	$p = 3$	$p = 4$	$p = 5$	$p = 6$
Point estimate	−2.02	−3.06	−2.68	−4.12
Std. err.	0.87	1.09	1.29	1.46

Figure 3: Fitted Values for 4-th Order Global Polynomial – Fully Interacted Model



Despite the negative estimated treatment effects, these are bad models. They all use global methods, but we know that RD treatment effects are inherently local to the cutoff. In essence, global methods do not compare 'apples with apples', because counties well below or above the cutoff could be different in systematically important ways.

2.3 Robust local polynomial methods

2.3.1 Baseline estimates

Table 4 shows the MSE-optimal RD point estimators and robust confidence intervals using local constant, linear and quadratic estimators (implemented with the `rdrobust` package in `R`). The estimated treatment effects are negative and significantly different from zero for all specifications, as expected. The standard errors of the point estimates increase with the order of the local polynomial.

Table 4: MSE-Optimal RD Treatment Effects with Different Polynomial Estimators

	Coeff	Std. Err.	CI Lower	CI Upper
		$p = 0$		
Conventional	-2.11	0.99	-4.05	-0.17
Bias-Corrected	-2.56	0.99	-4.50	-0.62
Robust	-2.56	1.23	-4.96	-0.15
		$p = 1$		
Conventional	-2.41	1.21	-4.77	-0.05
Bias-Corrected	-2.78	1.21	-5.14	-0.42
Robust	-2.78	1.37	-5.46	-0.10
		$p = 2$		
Conventional	-3.47	1.37	-6.16	-0.79
Bias-Corrected	-3.78	1.37	-6.46	-1.10
Robust	-3.78	1.45	-6.62	-0.94

2.3.2 Placebo tests

Table 5 shows estimated RD treatment effects for the pre-intervention outcome variable `mort_related_pre` and the unaffected post-intervention variable, `mort_injury_post`. Clearly, we cannot reject the null of no treatment effect on these placebo outcomes.

Table 5: Robustness Checks of RD Treatment Effects using Different Outcome Variables

	Coeff	Std. Err.	CI Lower	CI Upper
	$y =$ `mort_related_pre`			
Conventional	-2.38	2.25	-6.78	2.03
Bias-Corrected	-1.77	2.25	-6.17	2.64
Robust	-1.77	2.68	-7.01	3.48
	$y =$ `mort_injury_post`			
Conventional	1.13	3.77	-6.26	8.53
Bias-Corrected	1.52	3.77	-5.87	8.92
Robust	1.52	4.39	-7.07	10.12

2.4 Local randomization inference

I use the `rdlocrand` package in `R` to conduct local randomization inference on the Head Start data.

2.4.1 Window selection

First we need to select a window around the cutoff where randomization inference is appropriate. To do so, I use the `rdwinselect` function in `R`, with pre-intervention covariate `mort_related_pre` and the unaffected post-intervention variable, `mort_injury_post`. The basic idea is to choose the largest window around the cutoff where these covariates (which should not be related to the treatment) are "reasonably" balanced for counties above and below the cutoff. Essentially, we're trying to choose the biggest window around the cutoff where counties' poverty rates are pretty much random. Within this window, the RD design can be interpreted as a randomized experiment.

Using `rdwinselect` (and the default options), the recommended window is $[-0.976; 0.976]$ with 126 observations (68 below, 58 above).

2.4.2 Randomization inference

With the recommended window above, the observed difference in means is -2.304%, with Fisher p-value of 0.024 (associated with the sharp null of no treatment effect and 1000 replications).

3 Appendix: R code

```
## ECON675: ASSIGNMENT 5
## Q2: WEAK INSTRUMENTS SIMULATIONS
## Anirudh Yadav
## 11/19/2018

#########################################################################
# Load packages, clear workspace
#########################################################################
rm(list = ls())             #clear workspace
library(foreach)            #for looping
library(data.table)         #for data manipulation
library(Matrix)             #fast matrix calcs
library(ggplot2)            #for pretty plots
library(sandwich)           #for variance-covariance estimation
library(xtable)             #for latex tables
library(rdrobust)           #for RD plots and other stuff
library(rddensity)          #for RD density continuity tests
library(rdlocrand)          #for RD randomization inference
options(scipen = 999)       #forces R to use normal numbers instead of scientific notation

#########################################################################
# Input data
#########################################################################
setwd("/Users/Anirudh/Desktop/GitHub/PhD_Coursework/ECON675/HW6")
data <- as.data.table(read.csv('HeadStart.csv'))

#########################################################################
# [2.1]A RD Plots of pre-intervention mortality
#########################################################################

# Evenly-spaced bins, IMSE optimal
rdplot(data[,mort_related_pre],data[,povrate60],p=1,binselect = "es",x.label="povrate60",y.label="mort_related_pre",title="")
dev.copy(pdf,'q2-1-es.pdf')
dev.off()

# Evenly-spaced bins, mimicking variance
rdplot(data[,mort_related_pre],data[,povrate60],p=1,binselect = "esmv",x.label="povrate60",y.label="mort_related_pre",title="")
dev.copy(pdf,'q2-1-esmv.pdf')
dev.off()

# Quantile-spaced bins, IMSE optimal
rdplot(data[,mort_related_pre],data[,povrate60],p=1,binselect = "qs",x.label="povrate60",y.label="mort_related_pre",title="")
dev.copy(pdf,'q2-1-qs.pdf')
dev.off()

# Quantile-spaced bins, mimicking variance
rdplot(data[,mort_related_pre],data[,povrate60],p=1,binselect = "qsmv",x.label="povrate60",y.label="mort_related_pre",title="")
dev.copy(pdf,'q2-1-qsmv.pdf')
dev.off()

#########################################################################
# [2.1]B Formal falsification tests
#########################################################################

## Exact binomial tests for different windows around the cutoff

# Vector of windows
h.vec = seq(0.3,1.3,0.2)

# Get running variable
x     = data[,povrate60]

# Number of observations just above and below the cutoff
N.l   = sapply(1:length(h.vec),function(i) sum(x >= -h.vec[i] & x <=0))
N.u   = sapply(1:length(h.vec),function(i) sum(x >= 0 & x <= h.vec[i]))
```

```
# Total number of observations in the window
N.t   = N.l + N.u

# Conduct exact binomial tests (p=0.5), where success is treatment and store p-vals
binom.pvals = sapply(1:length(h.vec),function(i) binom.test(N.u[i],N.t[i])$p.value)

# Put results together for latex
binom.results = cbind(h.vec,N.l,N.u,binom.pvals)
xtable(binom.results,digits = c(0,1,0,0,3))

## Continuity in density tests (defaults are triangular kernel, jackknife SEs)
rdtest = rddensity(x)

#########################################################################
# [2.2]A Global polynomial regression - constant treatment effect
#########################################################################

# Create treatment dummy for regressions
treat=ifelse(data[,povrate60]>=0,1,0)

# Get outcome variable
Y = data[,mort_related_post]

# Generate covariates for polynomial regressions
X.pol = cbind(x,x^2,x^3,x^4,x^5,x^6)

# Run polynomial regressions
global.regs = lapply(0:3,function(i) lm(Y ~ treat + X.pol[,c(1:(3+i))]))

# Get point estimates
global.betas = sapply(1:4,function(i) global.regs[[i]]$coefficients[2])

# Get robust SEs
global.SEs   = sapply(1:4,function(i) sqrt(diag(vcovHC(global.regs[[i]],"HC2")))[2])

# Put results together
global.results = rbind(global.betas, global.SEs)
colnames(global.results) = c(3,4,5,6)
xtable(global.results,digits=c(0,2,2,2,2))

# Plot fitted values and data
temp.rd = rdplot(Y,x,hide=TRUE)
plot(temp.rd$vars_bins$rdplot_mean_x,temp.rd$vars_bins$rdplot_mean_y,pch=20,xlab="povrate60",ylab="mort_related_post")
points(x,global.regs[[2]]$fitted.values,pch=6,col="blue")
abline(v=0)
dev.copy(pdf,'q2-2-const.pdf')
dev.off()

#########################################################################
# [2.2]B Global polynomial regression - fully interacted model
#########################################################################

# Run fully-interacted polynomial regressions
global.regs.full = lapply(0:3,function(i) lm(Y ~ treat + X.pol[,c(1:(3+i))] + treat*X.pol[,c(1:(3+i))]))

# Get point estimates of treatment effect
global.betas.full = sapply(1:4,function(i) global.regs.full[[i]]$coefficients[2])

# Get robust SEs
global.SEs.full   = sapply(1:4,function(i) sqrt(diag(vcovHC(global.regs.full[[i]],"HC2")))[2])

# Put results together
global.results.full = rbind(global.betas.full, global.SEs.full)
colnames(global.results.full) = c(3,4,5,6)
xtable(global.results.full,digits=c(0,2,2,2,2))

# Plot fitted values + data
temp.rd = rdplot(Y,x,hide=TRUE)
```

```
plot(temp.rd$vars_bins$rdplot_mean_x,temp.rd$vars_bins$rdplot_mean_y,pch=20,xlab="povrate60",ylab="mort_related_post")
points(x,global.regs.full[[2]]$fitted.values,pch=6,col="blue")
abline(v=0)
dev.copy(pdf,'q2-2-full.pdf')
dev.off()

########################################################################
# [2.3] Robust local polynomial methods
########################################################################

# MSE-optimal RD treatment effect estimates
rd.regs = lapply(0:2, function(i) rdrobust(Y,x,p=i,all=TRUE))

# Combine results for different polynomial orders
rd.p0   = cbind(rd.regs[[1]]$coef,rd.regs[[1]]$se,rd.regs[[1]]$ci)
rd.p1   = cbind(rd.regs[[2]]$coef,rd.regs[[2]]$se,rd.regs[[2]]$ci)
rd.p2   = cbind(rd.regs[[3]]$coef,rd.regs[[3]]$se,rd.regs[[3]]$ci)

xtable(rd.p0,digits=c(0,2,2,2,2))
xtable(rd.p1,digits=c(0,2,2,2,2))
xtable(rd.p2,digits=c(0,2,2,2,2))

# Robustness checks
rd.rob1 = rdrobust(data[,mort_related_pre],x,p=1,all=TRUE)
rd.rob2 = rdrobust(data[,mort_injury_post],x,p=1,all=TRUE)

rd.rob1.res   = cbind(rd.rob1$coef,rd.rob1$se,rd.rob1$ci)
rd.rob2.res   = cbind(rd.rob2$coef,rd.rob2$se,rd.rob2$ci)

xtable(rd.rob1.res,digits=c(0,2,2,2,2))
xtable(rd.rob2.res,digits=c(0,2,2,2,2))

########################################################################
# [2.3] Local randomization inference
########################################################################

# Use defaults to compute recommended window for local randomization
rdwindow = rdwinselect(x,c(data[,mort_related_pre],data[,mort_injury_post]))

# Conduct randomization inference using recommended window
rd.rand.res = rdrandinf(Y,x,wl=rdwindow$window[1],wr=rdwindow$window[2])
```