Economics 675: Applied Microeconometrics
Fall 2018 - Assignment 6

Paul R. Organ*

December 5, 2018

Contents

*prorgan@umich.edu

1 Question 1: Continuity-Based Identification in SRD Designs

1. We show the result as follows:

$$\begin{aligned}
\lim_{\varepsilon\to 0^+}\mathbb{E}[Y_i|\tilde{X}_i=\varepsilon]-\lim_{\varepsilon\to 0^+}\mathbb{E}[Y_i|\tilde{X}_i=-\varepsilon] &= \lim_{\varepsilon\to 0^+}\mathbb{E}[Y_i|X_i=C_i+\varepsilon]-\lim_{\varepsilon\to 0^+}\mathbb{E}[Y_i|X_i=C_i-\varepsilon]\\
&= \lim_{\varepsilon\to 0^+}\mathbb{E}[Y_{1i}(C_i)|X_i=C_i+\varepsilon]-\lim_{\varepsilon\to 0^+}\mathbb{E}[Y_{0i}(C_i)|X_i=C_i-\varepsilon]\\
&= \lim_{\varepsilon\to 0^+}\mathbb{E}[Y_{1i}(c)|X_i=c+\varepsilon]-\lim_{\varepsilon\to 0^+}\mathbb{E}[Y_{0i}(c)|X_i=c-\varepsilon]\\
&= \mathbb{E}[Y_{1i}(c)|X_i=c]-\mathbb{E}[Y_{0i}(c)|X_i=c]\\
&= \mathbb{E}[Y_{1i}(c)-Y_{0i}(c)|X_i=c]
\end{aligned}$$

2. We show the result as follows:

$$\begin{aligned}
\lim_{\varepsilon\to 0^+}\mathbb{E}[Y_i|\tilde{X}_i=\varepsilon]-\lim_{\varepsilon\to 0^+}\mathbb{E}[Y_i|\tilde{X}_i=-\varepsilon] &= \lim_{\varepsilon\to 0^+}\mathbb{E}[Y_i|X_i=C_i+\varepsilon]-\lim_{\varepsilon\to 0^+}\mathbb{E}[Y_i|X_i=C_i-\varepsilon]\\
&= \lim_{\varepsilon\to 0^+}\mathbb{E}[Y_{1i}(C_i)|X_i=C_i+\varepsilon]-\lim_{\varepsilon\to 0^+}\mathbb{E}[Y_{0i}(C_i)|X_i=C_i-\varepsilon]\\
&= \sum_{c\in C}\lim_{\varepsilon\to 0^+}\Big[\mathbb{E}[Y_{1i}(c)|X_i=c+\varepsilon,C_i=c]\cdot\mathbb{P}[C_i=c|X_i=c+\varepsilon]\Big]\\
&\quad-\sum_{c\in C}\lim_{\varepsilon\to 0^+}\Big[\mathbb{E}[Y_{0i}(c)|X_i=c-\varepsilon,C_i=c]\cdot\mathbb{P}[C_i=c|X_i=c-\varepsilon]\Big]\\
&= \sum_{c\in C}\lim_{\varepsilon\to 0^+}\left[\mathbb{E}[Y_{1i}(c)|X_i=c+\varepsilon,C_i=c]\cdot\frac{f_{X|C}(c+\varepsilon|c)\mathbb{P}[C_i=c]}{\sum_{c\in C}f_{X|C}(c+\varepsilon|c)\mathbb{P}[C_i=c]}\right]\\
&\quad-\sum_{c\in C}\lim_{\varepsilon\to 0^+}\left[\mathbb{E}[Y_{0i}(c)|X_i=c-\varepsilon,C_i=c]\cdot\frac{f_{X|C}(c-\varepsilon|c)\mathbb{P}[C_i=c]}{\sum_{c\in C}f_{X|C}(c-\varepsilon|c)\mathbb{P}[C_i=c]}\right]\\
&= \sum_{c\in C}\left[\mathbb{E}[Y_{1i}(c)|X_i=c,C_i=c]\cdot\frac{f_{X|C}(c|c)\mathbb{P}[C_i=c]}{\sum_{c\in C}f_{X|C}(c|c)\mathbb{P}[C_i=c]}\right]\\
&\quad-\sum_{c\in C}\left[\mathbb{E}[Y_{0i}(c)|X_i=c,C_i=c]\cdot\frac{f_{X|C}(c|c)\mathbb{P}[C_i=c]}{\sum_{c\in C}f_{X|C}(c|c)\mathbb{P}[C_i=c]}\right]\\
&= \sum_{c\in C}\mathbb{E}[Y_{1i}(c)-Y_{0i}(c)|X_i=c,C_i=c]\cdot\frac{f_{X|C}(c|c)\mathbb{P}[C_i=c]}{\sum_{c\in C}f_{X|C}(c|c)\mathbb{P}[C_i=c]}
\end{aligned}$$

Now we allow for multiple cutoffs, and the estimand τ_{SRD} is the weighted average of the individual cutoffs calculated as in part (1) above.

The multi-cutoff and single-cutoff methods estimate the same parameter if $\mathbb{E}[Y_{1i}(c)-Y_{0i}(c)|X_i=c,C_i=c]$ is constant (i.e., if the average treatment effect is the same regardless of the cutoff).

3. We show the result as follows:

$$
\begin{aligned}
\lim_{\varepsilon\to 0^+}\mathbb{E}[Y_i|\tilde{X}_i=\varepsilon]-\lim_{\varepsilon\to 0^+}\mathbb{E}[Y_i|\tilde{X}_i=-\varepsilon] &= \sum_{c\in C}\mathbb{E}[Y_{1i}(c)-Y_{0i}(c)|X_i=c,C_i=c]\cdot\frac{f_{X|C}(c|c)\mathbb{P}[C_i=c]}{\sum_{c\in C}f_{X|C}(c|c)\mathbb{P}[C_i=c]}\\
&= \sum_{c\in C}\int\mathbb{E}[Y_{1i}(c)-Y_{0i}(c)|X_i=c,C_i=c,W_i=w]f_{W|X,C}(w|c,c)\cdot\\
&\qquad\frac{f_{X|C}(c|c)\mathbb{P}[C_i=c]}{\sum_{c\in C}f_{X|C}\mathbb{P}[C_i=c]}dw\\
&= \sum_{c\in C}\int\big(y_1(c,w)-y_0(c,w)\big)\cdot f_{W|X,C}(w|c,c)\cdot\frac{f_{X|C}(c|c)\mathbb{P}[C_i=c]}{\sum_{c\in C}f_{X|C}\mathbb{P}[C_i=c]}dw\\
&= \sum_{c\in C}\int\big(y_1(c,w)-y_0(c,w)\big)\cdot\frac{f_{X|C,W}(c|c,w)}{f_{X|C}(c|c)}\cdot f_{W|C}(w|c)\cdot\\
&\qquad\frac{f_{X|C}(c|c)\mathbb{P}[C_i=c]}{\sum_{c\in C}f_{X|C}\mathbb{P}[C_i=c]}dw\\
&= \sum_{c\in C}\int\big(y_1(c,w)-y_0(c,w)\big)\cdot\frac{f_{X|C,W}(c|c,w)}{f_{X|C}(c|c)}\cdot\frac{\mathbb{P}[C_i=c|W_i=w]}{\mathbb{P}[C_i=c]}\cdot\\
&\qquad\frac{f_{X|C}(c|c)\mathbb{P}[C_i=c]}{\sum_{c\in C}f_{X|C}\mathbb{P}[C_i=c]}dw\\
&= \sum_{c\in C}\Big[\int\big(y_1(c,w)-y_0(c,w)\big)\cdot\frac{f_{X|C,W}(c|c,w)}{f_{X|C}(c|c)}F_W(dw)\Big]\cdot\frac{f_{X|C}(c|c)\mathbb{P}[C_i=c]}{\sum_{c\in C}f_{X|C}\mathbb{P}[C_i=c]}\\
&= \int\big(y_1(w)-y_0(w)\big)\cdot\frac{f_{X|W}(c|w)}{f_X(c)}F_W(dw)
\end{aligned}
$$

2 Question 2 using Stata: The Effect of Head Start on Child Mortality

All results in this section rely on the underlying STATA code shown in Appendix A.

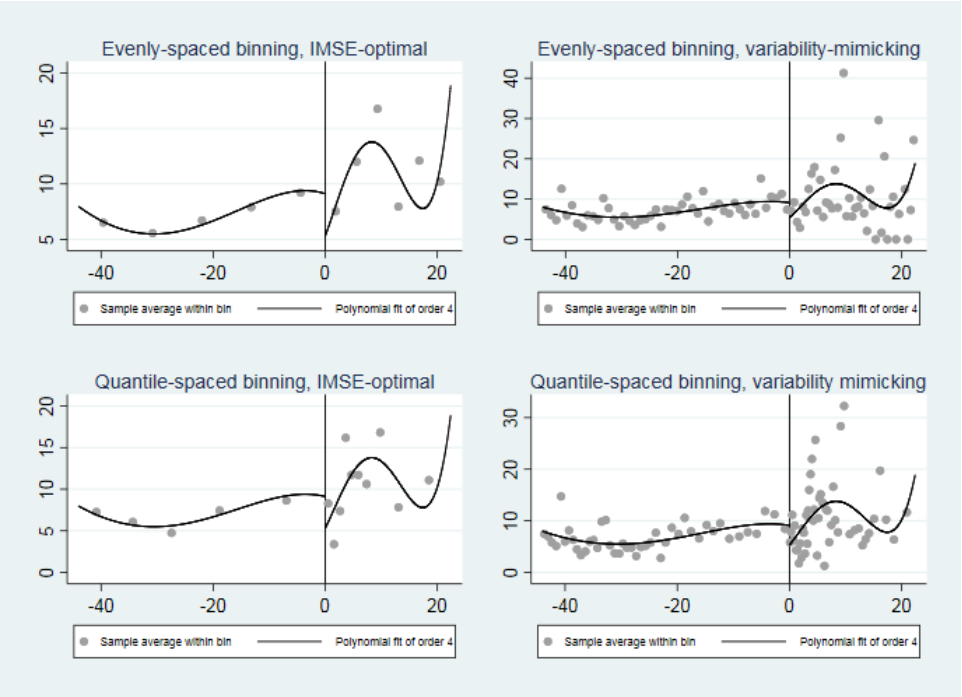
2.1 RD Plots and Falsification Tests

2.1.1 RD Plots

I construct four RD plots of the pre-intervention variable *mort_related_pre*, shown in Figure 1, using both (i) evenly-spaced and (ii) quantile-spaced binning, and choosing the total number of bins either (a) to be IMSE-optimal or (b) to mimic the overall data variability.

Although the polynomials might suggest a slight discontinuity, overall I do not think the plots as a whole show any clear evidence of a discontinuity in the pre-intervention variable.

Figure 1: RD Plots: STATA



2.1.2 Histogram Plots

A histogram plot of the running variable *povrate60* is shown in Figure 2.

The density of the running variable at the cutoff doesn't appear to jump sharply, although this is not conclusive.

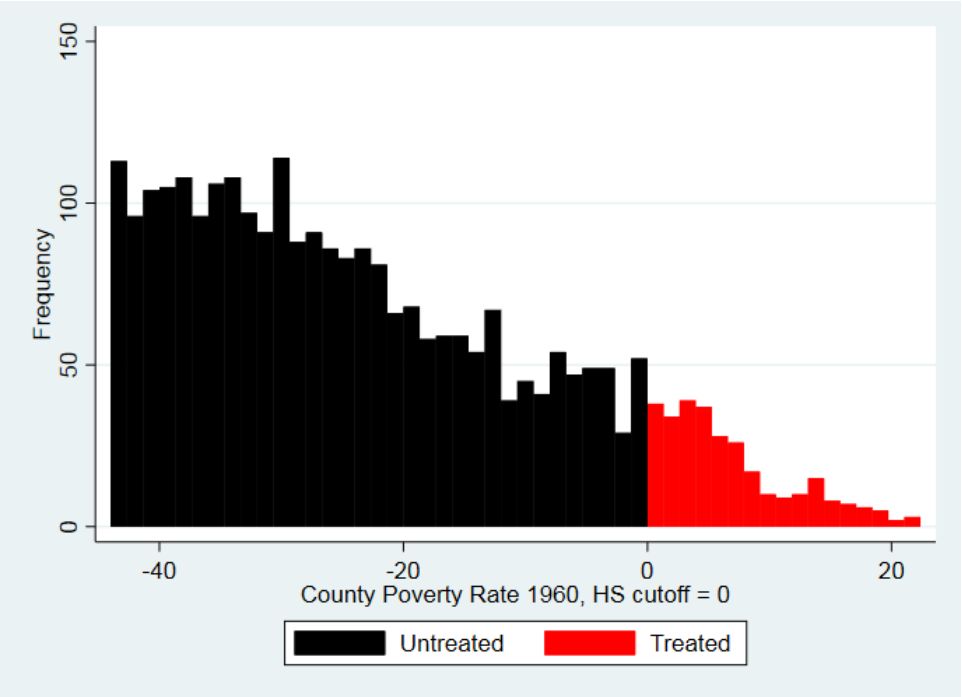
2.1.3 Local Binomial Tests

The results of a local binomial test using the RDWINSELECT function show that the test is not reject for a wide range of window widths, suggesting there is no clear evidence of running variable manipulation.

2.1.4 Continuity-in-Density Tests

The results of a series of continuity-in-density tests using the RDWINSELECT function under several different specifications show that the tests cannot reject the null hypothesis that there is no density discontinuity.

Figure 2: Histogram of Poverty Rates in 1960: STATA



2.2 Global and Flexible Parametric Methods

I estimate the treatment effect under three assumptions: constant treatment, heterogeneous treatment, and local parametric. The results are qualitatively consistent: each method produces a negative point estimate, suggesting Head Start reduces the related infant mortality rate. However the estimates vary in size, and are very sensitive to the model specifications.

2.2.1 Constant Treatment Effect Model

Assuming a "constant treatment effect model" I estimate the RD treatment effect using a p-th order global polynomial, with $p = 3, 4, 5, 6$. The point estimates and standard errors are shown in Table 1, and the fitted values are plotted in Figure 3.

The results are generally consistent (although they very somewhat with the polynomial order). However, the constant treatment effect assumption is unlikely to hold - hence we can proceed with the following sections.

Table 1: Global polynomial fit under constant treatment effect assumption: STATA

	p=3	p=4	p=5	p=6
Point Estimate	-1.1200	-1.0182	-1.6592	-1.7468
Standard Error	0.5947	0.7535	0.8112	0.8604

2.2.2 Heterogeneous Treatment Effect Model

Assuming a "heterogeneous treatment effect model" I estimate the RD treatment effect using a p-th order global polynomial, with $p = 3, 4, 5, 6$. The point estimates and standard errors are shown in Table 2, and the fitted values are plotted in Figure 4.

Here we are estimating the polynomial separately on the two sides of the cutoff. However, we see that the point estimates jump around quite a bit with the polynomial order. So we proceed in the next section to more refined estimates.

Figure 3: Fitted values for global fit under constant treatment: STATA

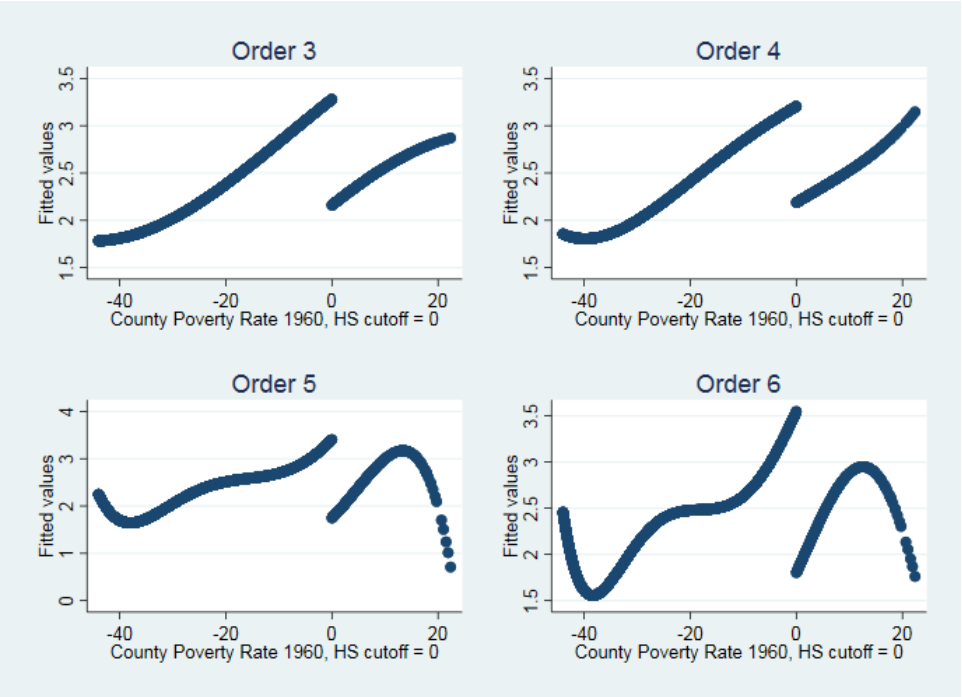


Table 2: Global polynomial fit under heterogeneous treatment effect assumption: STATA

	p=3	p=4	p=5	p=6
Point Estimate	-2.0196	-3.0647	-2.6763	-4.1240
Standard Error	0.8703	1.0881	1.2862	1.4620

2.2.3 Local Parametric Model

Assuming a "local parametric model" I estimate the RD treatment effect using a p-th order global polynomial, with $p = 0, 1, 2$ and with ad-hoc bandwidths of $h = 1, 5, 9, 18$. The point estimates and standard errors are shown in Table 3, and the fitted values are plotted in Figure 5.

Like before, we see that the point estimates can vary wildly depending on the choices of h and p, suggesting this is not a particularly stable approach.

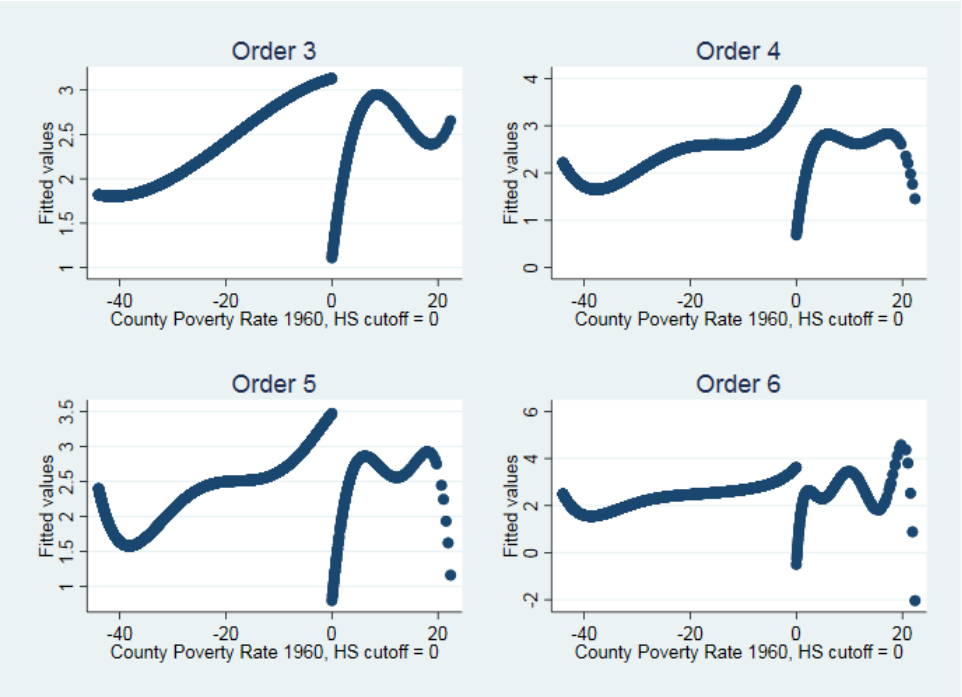
2.3 Robust Local Polynomial Methods

I employ robust local polynomial methods to estimate the effect of Head Start on child mortality, and then conduct several robustness checks. I find that a reasonable estimate of the effect of Head Start at the cutoff is that it reduced related infant mortality by 2 to 3, slightly less than the control mean presented in Ludwig and Miller (2007) Table III. This suggests Head Start had a statistically and economically (or socially) significant impact on related mortality.

2.3.1 MSE-optimal RD Estimation

I construct MSE-optimal RD point estimators and robust confidence intervals using local constant, local linear, and local quadratic estimators ($p = 0, 1, 2$). The results are shown in Table 4, and are generally more stable than we saw with the global polynomial methods above.

Figure 4: Fitted values for global fit under heterogeneous treatment: STATA



2.3.2 Robustness Check - Placebo Outcomes

Using the standard function RDROBUST (which I used above for the actual RD estimation) I test two alternative outcome variables: *mort_related_pre* (the pre-intervention test) and *mort_injury_post* (the post-intervention unrelated test). In both cases I find no evidence of a signficant treatment effect at the cutoff, which supports the claim of Head Start having an effect on related mortality at the cutoff.

2.3.3 Robustness Check - Bandwidth and Kernel Sensitivity

I test different bandwidth and kernel choices using a local linear regression ($p = 1$) and report the main inference results in Table 5.

The estimates are generally stable, especially so as the bandwidth gets above 2 or 3. The kernel choice does not seem to have a large effect on the estimates.

2.3.4 Robustness Check - "Donut Hole" Approach

I employ the "donut hole" approach and recompute the main inference results when excluding the closest $l \in \{1, ..., 10\}$ observations to the cutoff, using a local linear regression ($p = 1$). The results are shown in Table 6.

The estimates are not sensitive to the removal of observations close to the cutoff.

2.3.5 Robustness Check - Placebo Cutoffs

I employ a "placebo cutoff" approach and recompute the main inference results using cutoffs $c \in \{-10, -8, ..., 10\} \backslash \{0\}$. The results are shown in Table 7.

The treatment effects are not significant when using alternative cutoffs, except for one: $c = 2$. This could be because 2 is quite close to 0, so perhaps we are capturing some of the real effect at $c = 0$.

Table 3: Local parametric model results: STATA

	p=0	p=1	p=2
h=1			
Point Estimate	-2.3111	-2.5597	-5.7913
Standard Error	0.9044	2.0729	3.2231
h=5			
Point Estimate	-0.9780	-2.3962	-4.2097
Standard Error	0.6372	1.2524	1.4171
h=9			
Point Estimate	-0.6909	-1.8952	-2.6229
Standard Error	0.4544	0.9848	1.3165
h=18			
Point Estimate	-0.4140	-1.1983	-2.1321
Standard Error	0.3756	0.6626	1.0252

Table 4: Local Polynomial Estimation:STATA

(a) $p = 0$, local constant

	Point estimate	Standard error	Confidence interval
Conventional	-2.1137	(0.98967)	[-4.05344,-0.173988]
Bias-corrected	-2.5561	(0.98967)	[-4.49581,-0.616362]
Robust	-2.5561	(1.22820)	[-4.96328,-0.148888]

(b) $p = 1$, local linear

	Point estimate	Standard error	Confidence interval
Conventional	-2.4090	(1.2056)	[-4.77201,-0.046024]
Bias-corrected	-2.7806	(1.2056)	[-5.14364,-0.417655]
Robust	-2.7806	(1.3683)	[-5.46238,-0.098916]

(c) $p = 2$, local quadratic

	Point estimate	Standard error	Confidence interval
Conventional	-3.4744	(1.3684)	[-6.15640,-0.792448]
Bias-corrected	-3.7789	(1.3684)	[-6.46087,-1.09692]
Robust	-3.7789	(1.4480)	[-6.61696,-0.940831]

2.4 Local Randomization Methods

Using the local randomization approach, I come to similar conclusion as I did with the robust local polynomial approach. Namely, I find that Head Start did reduce related child mortality in the treated counties.

2.4.1 Window Selection

I use the function RDWINSELECT with the four included statistics. The tests recommend a window of within either 1.7 or 1.9 of the cutoff point, zero. I use 1.7 in the following analysis.

An RD plot of the outcome variable within the neighborhood of [-1.7,1.7] is shown in Figure 6, constructed using evenly-spaced binning and the IMSE-optimal number of bins.

2.4.2 Basic Analysis

The estimated treatment effect within the neighborhood of [-1.7,1.7] is 2.656.

Figure 5: Fitted values for local parametric models: STATA

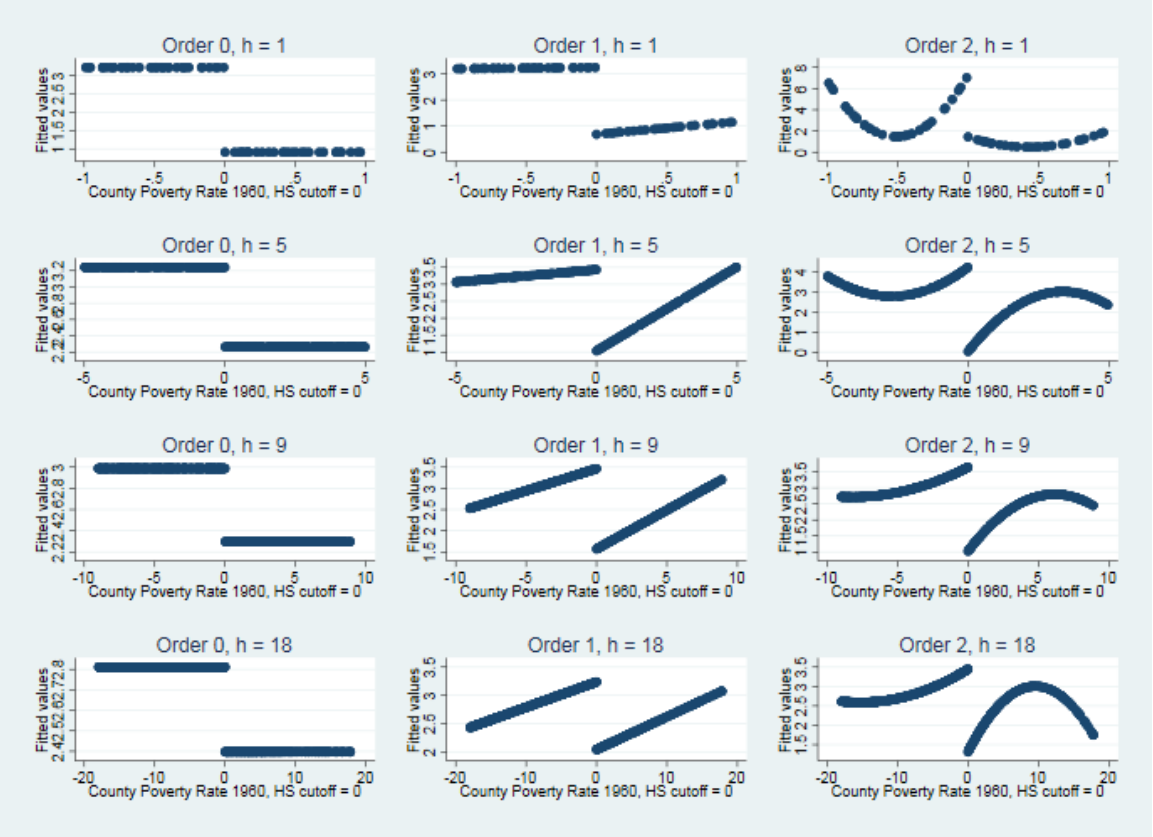


Table 5: Bias-corrected coefficient estimates for various bandwidths and kernels: STATA

bandwidth $h =$	1	2	3	4	5	6	7	8	9	10
Triangular	-4.72	-3.07	-2.04	-2.85	-3.59	-3.86	-3.67	-3.29	-3.04	-2.92
Uniform	-5.79	-1.82	-2.28	-3.83	-4.21	-3.94	-3.10	-2.33	-2.62	-2.75
Epanechnikov	-4.97	-2.61	-1.86	-3.03	-3.85	-4.04	-3.69	-3.17	-2.87	-2.78

2.4.3 Sensitivity Analysis

I test the sensitivity of the window size using a simple regression of *mort_rel_post* on *treatment* within a variety of windows. The results are shown in Table 8, and show relatively similar results to the analyses above.

Table 6: Bias-corrected coefficient estimates when dropping observations close to the cutoff: STATA

#(obs) dropped $l =$	1	2	3	4	5	6	7	8	9	10
Point estimate	-2.73	-2.80	-2.58	-2.83	-2.97	-2.93	-2.67	-2.62	-2.55	-2.48

Table 7: Bias-corrected coefficient estimates and p-values when using alternative cutoffs: STATA

cutoff $c =$	-10	-8	-6	-4	-2	2	4	6	8	10
Point estimate	0.55	-0.26	0.40	-0.09	2.24	3.13	-1.53	1.64	-5.80	4.18
p-value	0.56	0.82	0.70	0.94	0.24	0.04	0.38	0.19	0.10	0.37

Figure 6: Outcome variables within the selected neighborhood: STATA

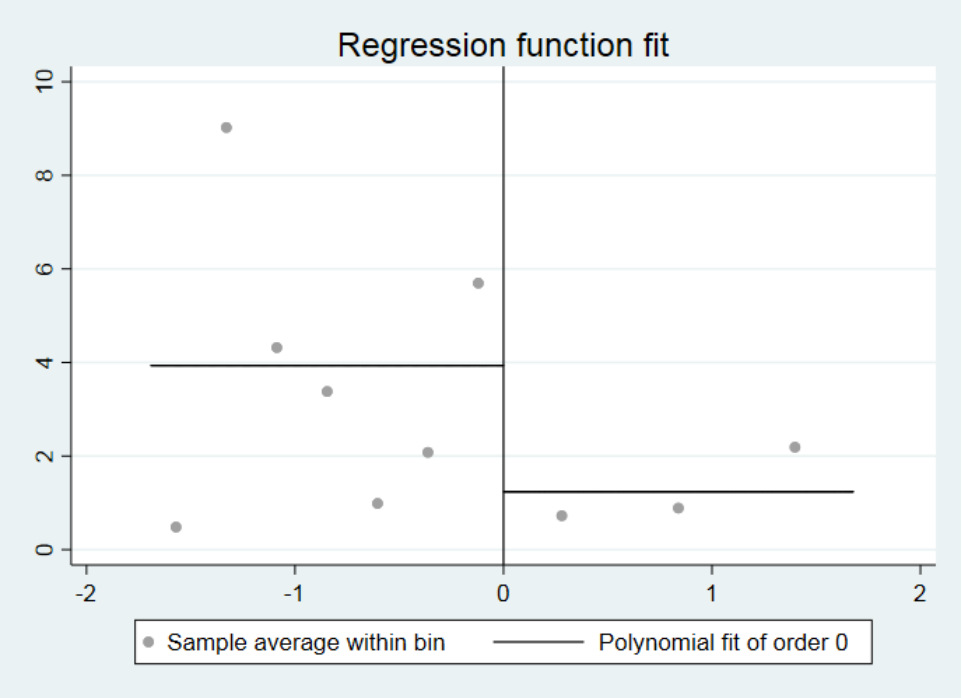


Table 8: Randomization sensitivity to window size: STATA

Window size $w =$	0.8	1	1.2	1.4	1.6	1.8	2	2.2	2.4	2.6
Point estimate	-1.664	-2.311	-2.194	-3.089	-2.937	-2.555	-2.521	-2.465	-1.996	-1.757
Standard error	0.993	0.904	0.821	1.381	1.359	1.203	1.096	1.050	1.044	1.000
p-value	0.099	0.013	0.009	0.028	0.033	0.036	0.023	0.020	0.058	0.081

3 Question 2 using R: The Effect of Head Start on Child Mortality

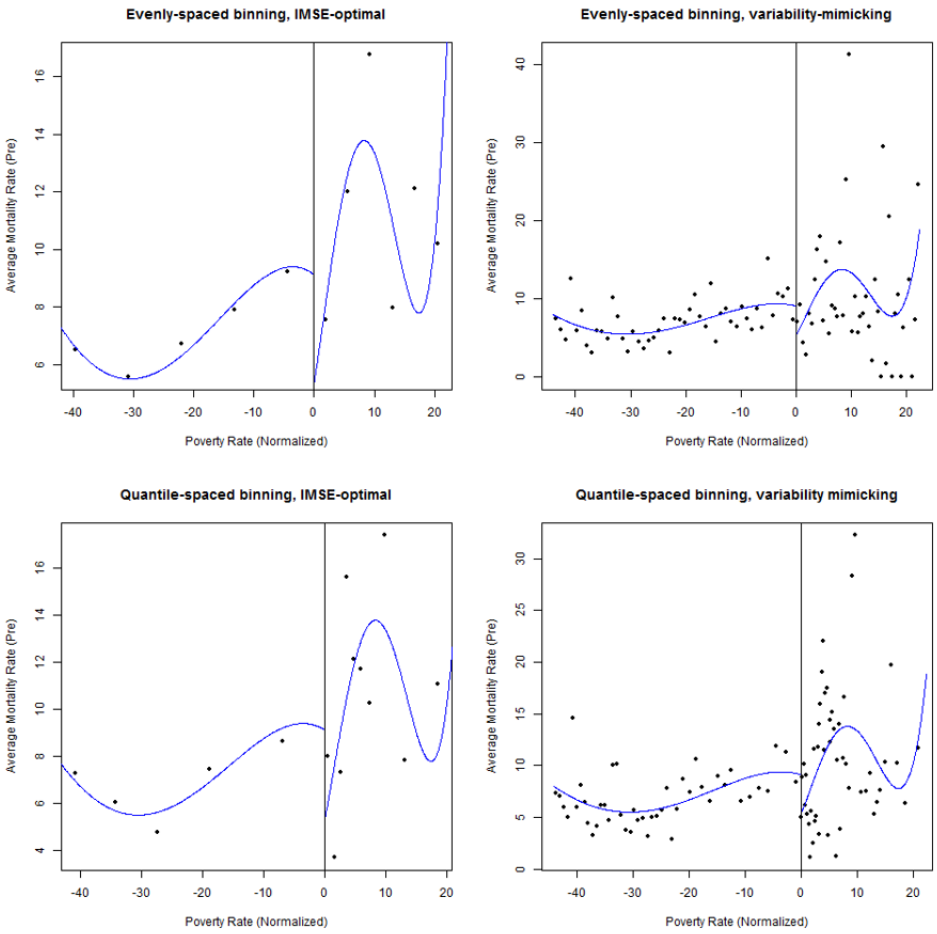
All results in this section rely on the underlying R code shown in Appendix B.

3.1 RD Plots and Falsification Tests

I construct four RD plots of the pre-intervention variable *mort_related_pre*, shown in Figure 7, using both (i) evenly-spaced and (ii) quantile-spaced binning, and choosing the total number of bins either (a) to be IMSE-optimal or (b) to mimic the overall data variability.

Although the polynomials might suggest a slight discontinuity, overall I do not think the plots as a whole show any clear evidence of a discontinuity in the pre-intervention variable.

Figure 7: RD Plots: R

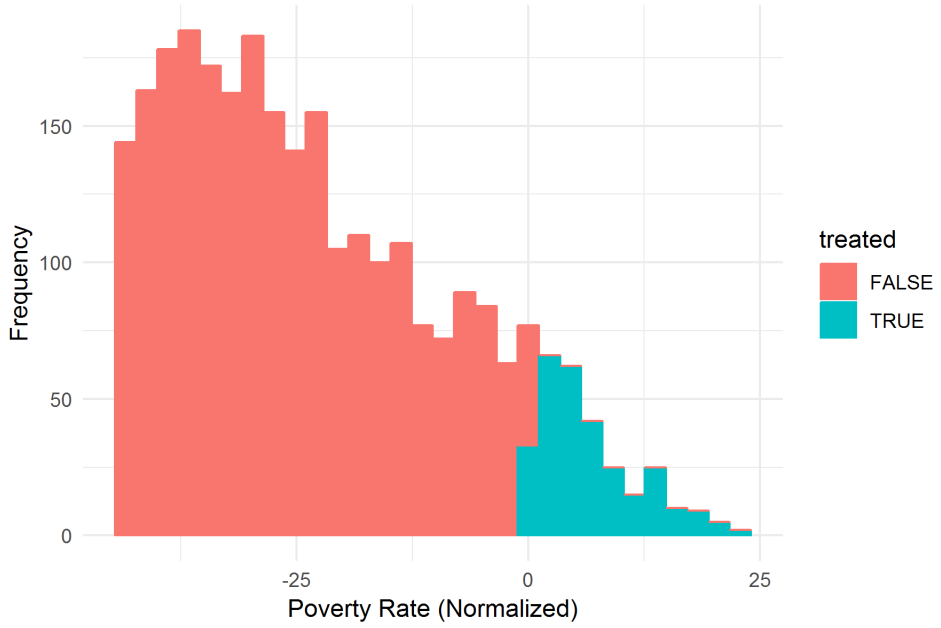


3.1.1 Histogram Plots

A histogram plot of the running variable *povrate60* is shown in Figure 8.

The density of the running variable at the cutoff doesn't appear to jump sharply, although this is not conclusive.

Figure 8: Histogram of Poverty Rates in 1960: R



3.1.2 Local Binomial Tests

The results of a local binomial test using the RDWINSELECT function show that the test is not reject for a wide range of window widths, suggesting there is no clear evidence of running variable manipulation.

3.1.3 Continuity-in-Density Tests

The results of a series of continuity-in-density tests using the RDWINSELECT function under several different specifications show that the tests cannot reject the null hypothesis that there is no density discontinuity.

3.2 Global and Flexible Parametric Methods

I estimate the treatment effect under three assumptions: constant treatment, heterogeneous treatment, and local parametric. The results are qualitatively consistent: each method produces a negative point estimate, suggesting Head Start reduces the related infant mortality rate. However the estimates vary in size, and are very sensitive to the model specifications.

3.2.1 Constant Treatment Effect Model

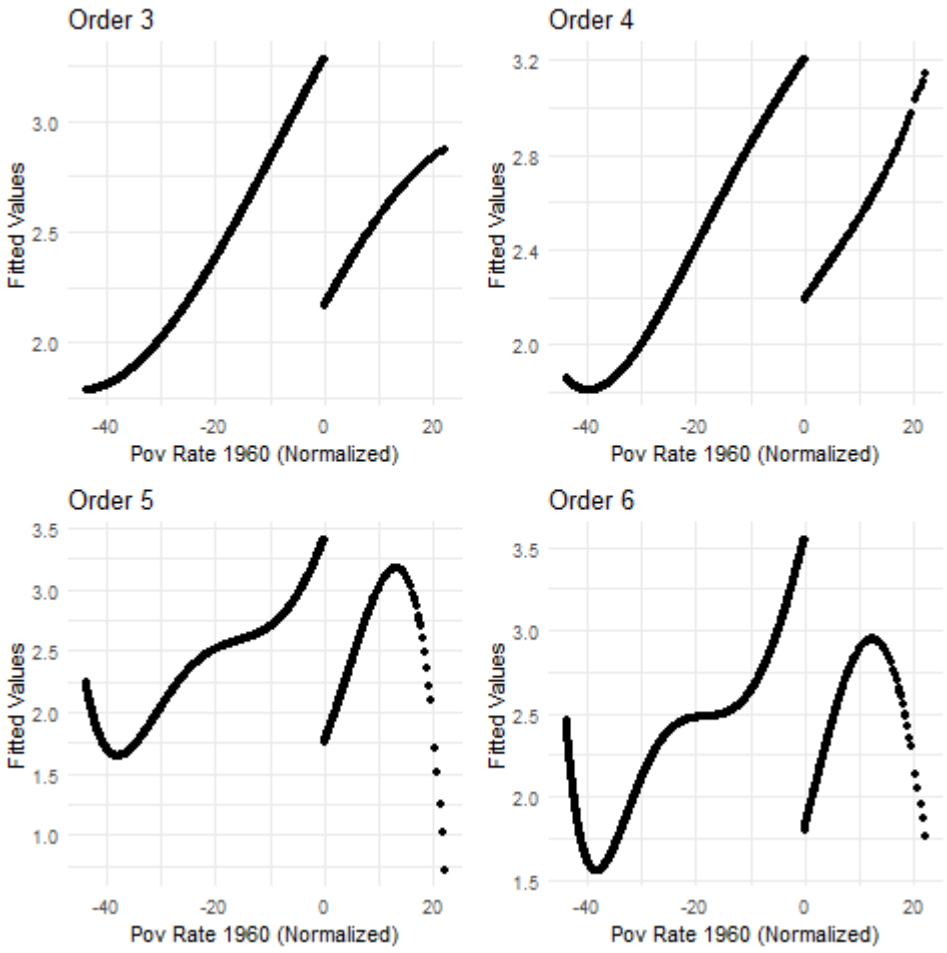
Assuming a "constant treatment effect model" I estimate the RD treatment effect using a p-th order global polynomial, with $p = 3, 4, 5, 6$. The point estimates and standard errors are shown in Table 9, and the fitted values are plotted in Figure 9.

The results are generally consistent (although they very somewhat with the polynomial order). However, the constant treatment effect assumption is unlikely to hold - hence we can proceed with the following sections.

Table 9: Global polynomial fit under constant treatment effect assumption: R

	p=3	p=4	p=5	p=6
Point Estimate	-1.12	-1.02	-1.66	-1.75
Standard Error	0.59	0.75	0.81	0.86

Figure 9: Fitted values for global fit under constant treatment: R



3.2.2 Heterogeneous Treatment Effect Model

Assuming a "heterogeneous treatment effect model" I estimate the RD treatment effect using a p-th order global polynomial, with $p = 3, 4, 5, 6$. The point estimates and standard errors are shown in Table 10, and the fitted values are plotted in Figure 10.

Here we are estimating the polynomial separately on the two sides of the cutoff. However, we see that the point estimates jump around quite a bit with the polynomial order. So we proceed in the next section to more refined estimates.

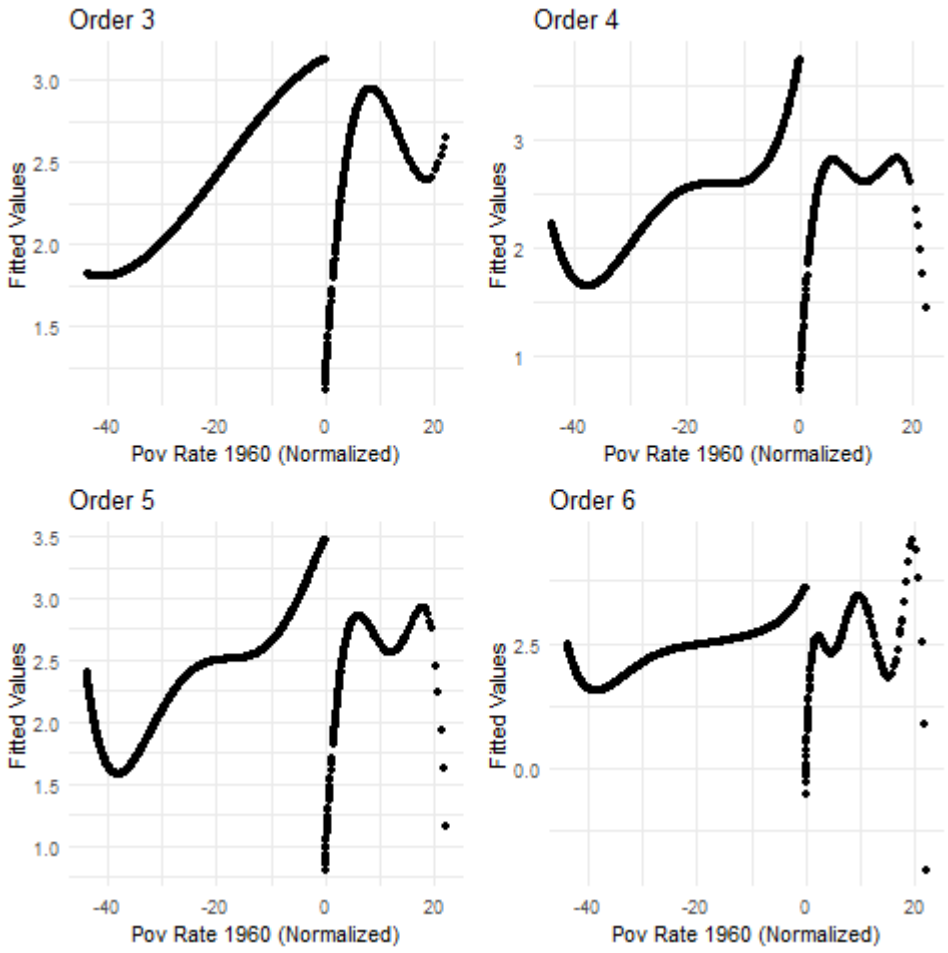
Table 10: Global polynomial fit under heterogeneous treatment effect assumption: R

	p=3	p=4	p=5	p=6
Point Estimate	-2.02	-3.06	-2.68	-4.12
Standard Error	0.87	1.09	1.29	1.46

3.2.3 Local Parametric Model

Assuming a "local parametric model" I estimate the RD treatment effect using a p-th order global polynomial, with $p = 0, 1, 2$ and with ad-hoc bandwidths of $h = 1, 5, 9, 18$. The point estimates and standard errors are shown in Table 11, and the fitted values are plotted in Figure 11.

Figure 10: Fitted values for global fit under heterogeneous treatment: R



Like before, we see that the point estimates can vary wildly depending on the choices of h and p, suggesting this is not a particularly stable approach.

3.3 Robust Local Polynomial Methods

I employ robust local polynomial methods to estimate the effect of Head Start on child mortality, and then conduct several robustness checks. I find that a reasonable estimate of the effect of Head Start at the cutoff is that it reduced related infant mortality by 2 to 3, slightly less than the control mean presented in Ludwig and Miller (2007) Table III. This suggests Head Start had a statistically and economically (or socially) significant impact on related mortality.

3.3.1 MSE-optimal RD Estimation

I construct MSE-optimal RD point estimators and robust confidence intervals using local constant, local linear, and local quadratic estimators ($p = 0, 1, 2$). The results are shown in Table 12, and are generally more stable than we saw with the global polynomial methods above.

3.3.2 Robustness Check - Placebo Outcomes

Using the standard function RDROBUST (which I used above for the actual RD estimation) I test two alternative outcome variables: *mort_related_pre* (the pre-intervention test) and *mort_injury_post* (the post-

Table 11: Local parametric model results: R

	p=0	p=1	p=2
h=1			
Point Estimate	-2.31	-2.56	-5.79
Standard Error	0.90	2.07	3.22
h=5			
Point Estimate	-0.98	-2.40	-4.21
Standard Error	0.64	1.25	1.42
h=9			
Point Estimate	-0.69	-1.90	-2.62
Standard Error	0.45	0.98	1.32
h=18			
Point Estimate	-0.41	-1.20	-2.13
Standard Error	0.38	0.66	1.03

Table 12: Local Polynomial Estimation:R

(a) $p = 0$, local constant

	Point estimate	Standard error	Confidence interval
Conventional	-2.114	(0.990)	[-4.053,-0.174]
Bias-corrected	-2.556	(0.990)	[-4.496,-0.617]
Robust	-2.556	(1.228)	[-4.963,-0.149]

(b) $p = 1$, local linear

	Point estimate	Standard error	Confidence interval
Conventional	-2.409	(1.206)	[-4.772,-0.046]
Bias-corrected	-2.781	(1.206)	[-5.144,-0.418]
Robust	-2.781	(1.368)	[-5.462,-0.099]

(c) $p = 2$, local quadratic

	Point estimate	Standard error	Confidence interval
Conventional	-3.474	(1.368)	[-6.156,-0.792]
Bias-corrected	-3.779	(1.368)	[-6.461,-1.097]
Robust	-3.779	(1.448)	[-6.617,-0.941]

intervention unrelated test). In both cases I find no evidence of a signficant treatment effect at the cutoff, which supports the claim of Head Start having an effect on related mortality at the cutoff.

3.3.3 Robustness Check - Bandwidth and Kernel Sensitivity

I test different bandwidth and kernel choices using a local linear regression ($p = 1$) and report the main inference results in Table 13.

The estimates are generally stable, especially so as the bandwidth gets above 2 or 3. The kernel choice does not seem to have a large effect on the estimates.

3.3.4 Robustness Check - "Donut Hole" Approach

I employ the "donut hole" approach and recompute the main inference results when excluding the closest $l \in \{1, ..., 10\}$ observations to the cutoff, using a local linear regression ($p = 1$). The results are shown in Table 14.

The estimates are not sensitive to the removal of observations close to the cutoff.

Table 13: Bias-corrected coefficient estimates for various bandwidths and kernels: R

bandwidth $h =$	1	2	3	4	5	6	7	8	9	10
Triangular	-4.72	-3.07	-2.04	-2.85	-3.59	-3.86	-3.67	-3.29	-3.04	-2.92
Uniform	-5.79	-1.82	-2.28	-3.83	-4.21	-3.94	-3.10	-2.33	-2.62	-2.75
Epanechnikov	-4.97	-2.61	-1.86	-3.03	-3.85	-4.04	-3.69	-3.17	-2.87	-2.78

Table 14: Bias-corrected coefficient estimates when dropping observations close to the cutoff: R

#(obs) dropped $l =$	1	2	3	4	5	6	7	8	9	10
Point estimate	-2.73	-2.80	-2.58	-2.83	-2.97	-2.93	-2.67	-2.62	-2.55	-2.48

3.3.5 Robustness Check - Placebo Cutoffs

I employ a "placebo cutoff" approach and recompute the main inference results using cutoffs $c \in \{-10, -8, ..., 10\} \backslash \{0\}$. The results are shown in Table 15.

The treatment effects are not significant when using alternative cutoffs, except for one: $c = 2$. This could be because 2 is quite close to 0, so perhaps we are capturing some of the real effect at $c = 0$.

Table 15: Bias-corrected coefficient estimates and p-values when using alternative cutoffs: R

cutoff $c =$	-10	-8	-6	-4	-2	2	4	6	8	10
Point estimate	0.55	-0.26	0.40	-0.09	2.24	3.13	-1.53	1.64	-5.80	4.18
p-value	0.56	0.82	0.70	0.94	0.24	0.04	0.38	0.19	0.10	0.37

3.4 Local Randomization Methods

Using the local randomization approach, I come to similar conclusion as I did with the robust local polynomial approach. Namely, I find that Head Start did reduce related child mortality in the treated counties.

3.4.1 Window Selection

I use the function RDWINSELECT with the four included statistics. The tests recommend a window of within either 1.7 or 1.9 of the cutoff point, zero. I use 1.7 in the following analysis.

An RD plot of the outcome variable within the neighborhood of [-1.7,1.7] is shown in Figure 12, constructed using evenly-spaced binning and the IMSE-optimal number of bins.

3.4.2 Basic Analysis

The estimated treatment effect within the neighborhood of [-1.7,1.7] is 2.656.

3.4.3 Sensitivity Analysis

I test the sensitivity of the window size using a simple regression of *mort_rel_post* on *treatment* within a variety of windows. The results are shown in Table 16, and show relatively similar results to the analyses above.

Table 16: Randomization sensitivity to window size: R

Window size $w =$	0.8	1	1.2	1.4	1.6	1.8	2	2.2	2.4	2.6
Point estimate	-1.66	-2.31	-2.19	-3.09	-2.94	-2.56	-2.52	-2.47	-2.00	-1.76
Standard error	0.99	0.90	0.82	1.38	1.36	1.20	1.10	1.05	1.04	1.00
p-value	0.11	0.02	0.01	0.05	0.05	0.06	0.04	0.03	0.07	0.09

Figure 11: Fitted values for local parametric models: R

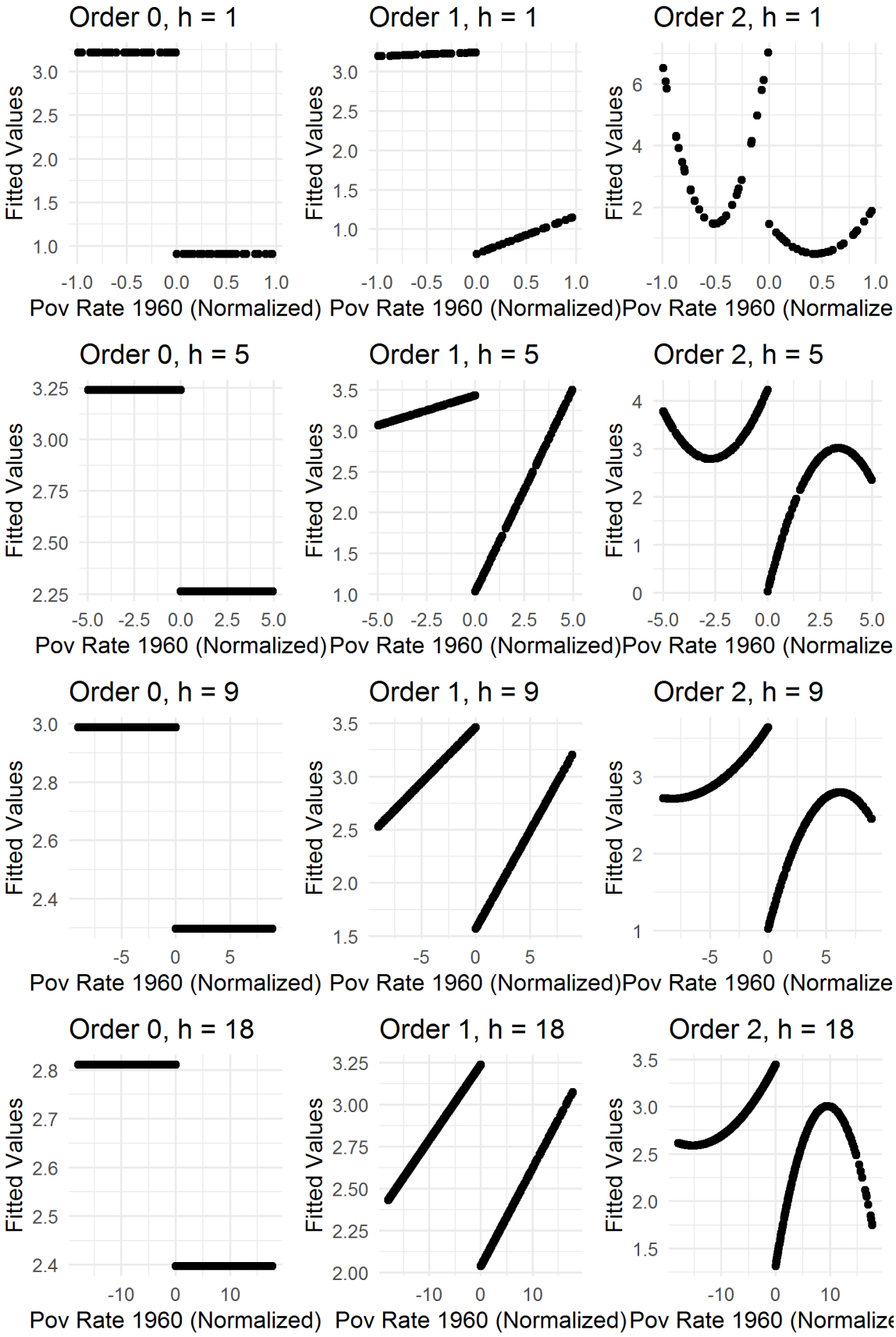
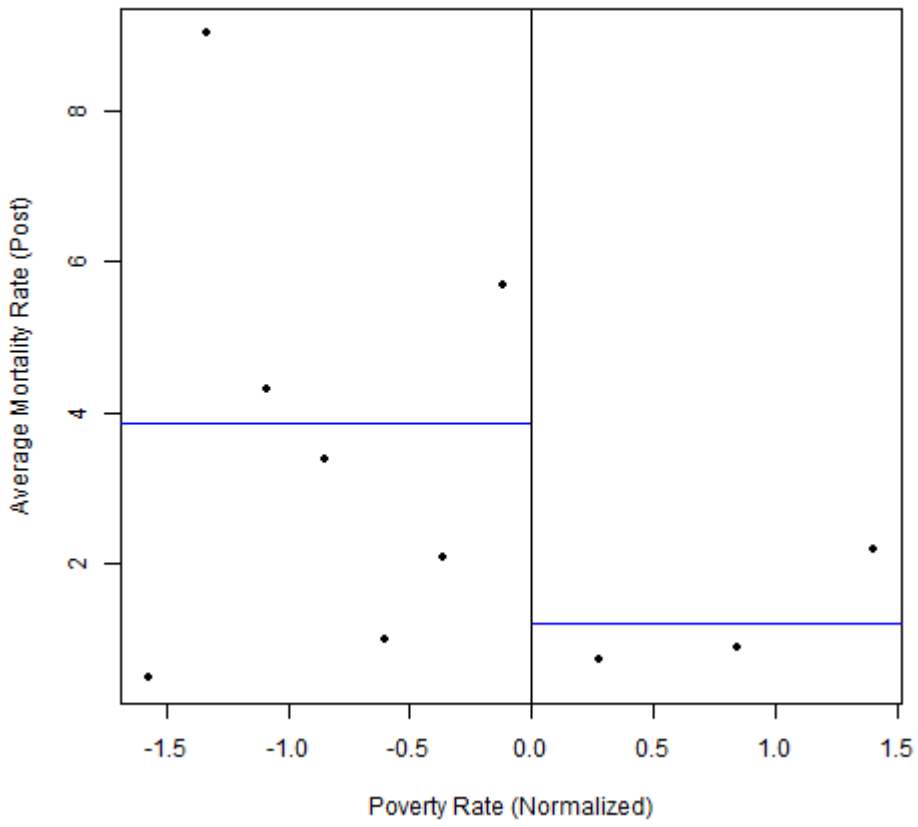


Figure 12: Outcome variables within the selected neighborhood: R



A Stata Code

```
*******************************************************************************
* Author: Paul R. Organ
* Purpose: ECON 675, PS6
* Last Update: Dec 4, 2018
*******************************************************************************
clear all
set more off
capture log close

cd "C:\Users\prorgan\Box\Classes\Econ 675\Problem Sets\PS6"
log using ps6.log, replace

use HeadStart.dta

* from the appendix
* counties are treated if povrate60 >0, not treated o/w
* outcome var is mort_related_post
* preintervention var is more_related_pre
* exog post var is mort_injury_post

* packages: rdrobust, rdlocrand, rddensity, rdmulti, rdpower
* https://sites.google.com/site/rdpackages/home

* rename variable for ease of use
rename povrate60 pov
rename mort_related_pre rel_pre
rename mort_related_post rel_post
rename mort_injury_post inj_post

*******************************************************************************
*** Question 2: The Effect of Head Start on Child Mortality
*******************************************************************************
* Q2.1.1: RD Plots

* using the rdrobust package from Matias
rdplot rel_pre pov, c(0) binselect(es) ///
    graph_options(title("Evenly-spaced binning, IMSE-optimal", size(medium)) ///
                                        legend(size(vsmall)))
graph save "s\1_1a.gph", replace

rdplot rel_pre pov, c(0) binselect(esmv) ///
    graph_options(title("Evenly-spaced binning, variability-mimicking", ///
                  size(medium)) legend(size(vsmall)))
graph save "s\1_1b.gph", replace

rdplot rel_pre pov, c(0) binselect(qs) ///
    graph_options(title("Quantile-spaced binning, IMSE-optimal", ///
                  size(medium)) legend(size(vsmall)))
graph save "s\1_1c.gph", replace

rdplot rel_pre pov, c(0) binselect(qsmv) ///
    graph_options(title("Quantile-spaced binning, variability mimicking", ///
```

```
                    size(medium)) legend(size(vsmall)))
graph save "s\1_1d.gph", replace

graph combine "s\1_1a.gph" "s\1_1b.gph" "s\1_1c.gph" "s\1_1d.gph"
graph export "s\1_1s.png", replace

*******************************************************************************
* Q2.1.2: Falsification Tests
* falsification of RD design using three methods

gen t = pov > 0

* (1) histogram plots
twoway (hist pov if t, freq bcolor(red)) ///
           (hist pov if !t, freq bcolor(black)), ///
           legend(label(1 "Treated") label(2 "Untreated") order(2 1))
graph export "s\1_2s.png", replace

* (2) binomial tests using rdlocrand package - how to interpret?
rdwinselect pov, wmin(0.05) wstep(0.05) nwindows(100)

* (3) continuity-in-density tests using rddensity package - how to interpret?
rddensity pov, all
rddensity pov, all h(5 5)
rddensity pov, all h(1 1)

*******************************************************************************
* Q2.2.1: constant treatment effect model

* generate variables for polynomials of order 3-6
gen p2 = pov^2
gen p3 = pov^3
gen p4 = pov^4
gen p5 = pov^5
gen p6 = pov^6

* empty matrix to fill with point estimates and standard errors
matrix tbl = J(2,4,.)

* order 3 (run reg, grab ests, predict vals, plot, save for combine later)
reg rel_post t pov p2 p3, vce(hc2)
matrix tbl[1,1] = _b["t"]
matrix tbl[2,1] = _se["t"]
capture drop pred
predict pred
twoway scatter pred pov, title("Order 3")
graph save "s\2_1a.gph", replace

* order 4
reg rel_post t pov p2 p3 p4, vce(hc2)
matrix tbl[1,2] = _b["t"]
matrix tbl[2,2] = _se["t"]
capture drop pred
predict pred
```

```
twoway scatter pred pov, title("Order 4")
graph save "s\2_1b.gph", replace

* order 5
reg rel_post t pov p2 p3 p4 p5, vce(hc2)
matrix tbl[1,3] = _b["t"]
matrix tbl[2,3] = _se["t"]
capture drop pred
predict pred
twoway scatter pred pov, title("Order 5")
graph save "s\2_1c.gph", replace

* order 6
reg rel_post t pov p2 p3 p4 p5 p6, vce(hc2)
matrix tbl[1,4] = _b["t"]
matrix tbl[2,4] = _se["t"]
capture drop pred
predict pred
twoway scatter pred pov, title("Order 6")
graph save "s\2_1d.gph", replace

* combine graphs
graph combine "s\2_1a.gph" "s\2_1b.gph" "s\2_1c.gph" "s\2_1d.gph"
graph export "s\2_1s.png", replace

* write table for LaTeX
mat2txt, matrix(tbl) saving("s\2_1.txt") format(%9.4f) replace

*******************************************************************************
* Q2.2.2: heterogeneous treatment effect model

* define new, interacted terms for polynomials
gen p1t = pov*t
gen p1u = pov*(1-t)
gen p2t = p2*t
gen p2u = p2*(1-t)
gen p3t = p3*t
gen p3u = p3*(1-t)
gen p4t = p4*t
gen p4u = p4*(1-t)
gen p5t = p5*t
gen p5u = p5*(1-t)
gen p6t = p6*t
gen p6u = p6*(1-t)

* empty matrix to fill with point estimates and standard errors
matrix tbl = J(2,4,.)

local base = "p1t p1u p2t p2u p3t p3u"

* order 3 (run reg, grab ests, predict vals, plot, save for combine later)
reg rel_post t `base', vce(hc2)
matrix tbl[1,1] = _b["t"]
matrix tbl[2,1] = _se["t"]
```

```
capture drop pred
predict pred
twoway scatter pred pov, title("Order 3")
graph save "s\2_2a.gph", replace

* order 4
reg rel_post t `base' p4t p4u, vce(hc2)
matrix tbl[1,2] = _b["t"]
matrix tbl[2,2] = _se["t"]
capture drop pred
predict pred
twoway scatter pred pov, title("Order 4")
graph save "s\2_2b.gph", replace

* order 5
reg rel_post t `base' p4t p4u p5t p5u, vce(hc2)
matrix tbl[1,3] = _b["t"]
matrix tbl[2,3] = _se["t"]
capture drop pred
predict pred
twoway scatter pred pov, title("Order 5")
graph save "s\2_2c.gph", replace

* order 6
reg rel_post t `base' p4t p4u p5t p5u p6t p6u, vce(hc2)
matrix tbl[1,4] = _b["t"]
matrix tbl[2,4] = _se["t"]
capture drop pred
predict pred
twoway scatter pred pov, title("Order 6")
graph save "s\2_2d.gph", replace

* combine graphs
graph combine "s\2_2a.gph" "s\2_2b.gph" "s\2_2c.gph" "s\2_2d.gph"
graph export "s\2_2s.png", replace

* write table for LaTeX
mat2txt, matrix(tbl) saving("s\2_2.txt") format(%9.4f) replace

*******************************************************************************
* Q2.2.3: local parametric model (p = 0,1,2 and h = 1,5,9,18)

* matrix to fill with estimates
matrix tbl = J(12,3,.)
matrix tbl[1,1] = 1
matrix tbl[4,1] = 5
matrix tbl[7,1] = 9
matrix tbl[10,1] = 18

* for h = 1
reg rel_post t if abs(pov)<=1, vce(hc2)
matrix tbl[2,1] = _b["t"]
matrix tbl[3,1] = _se["t"]
capture drop pred
```

```
predict pred if abs(pov)<=1
twoway scatter pred pov if abs(pov)<=1, title("Order 0, h = 1")
graph save "s\2_3h1a.gph", replace

reg rel_post t p1t p1u if abs(pov)<=1, vce(hc2)
matrix tbl[2,2] = _b["t"]
matrix tbl[3,2] = _se["t"]
capture drop pred
predict pred if abs(pov)<=1
twoway scatter pred pov if abs(pov)<=1, title("Order 1, h = 1")
graph save "s\2_3h1b.gph", replace

reg rel_post t p1t p1u p2t p2u if abs(pov)<=1, vce(hc2)
matrix tbl[2,3] = _b["t"]
matrix tbl[3,3] = _se["t"]
capture drop pred
predict pred if abs(pov)<=1
twoway scatter pred pov if abs(pov)<=1, title("Order 2, h = 1")
graph save "s\2_3h1c.gph", replace

* for h = 5
reg rel_post t if abs(pov)<=5, vce(hc2)
matrix tbl[5,1] = _b["t"]
matrix tbl[6,1] = _se["t"]
capture drop pred
predict pred if abs(pov)<=5
twoway scatter pred pov if abs(pov)<=5, title("Order 0, h = 5")
graph save "s\2_3h5a.gph", replace

reg rel_post t p1t p1u if abs(pov)<=5, vce(hc2)
matrix tbl[5,2] = _b["t"]
matrix tbl[6,2] = _se["t"]
capture drop pred
predict pred if abs(pov)<=5
twoway scatter pred pov if abs(pov)<=5, title("Order 1, h = 5")
graph save "s\2_3h5b.gph", replace

reg rel_post t p1t p1u p2t p2u if abs(pov)<=5, vce(hc2)
matrix tbl[5,3] = _b["t"]
matrix tbl[6,3] = _se["t"]
capture drop pred
predict pred if abs(pov)<=5
twoway scatter pred pov if abs(pov)<=5, title("Order 2, h = 5")
graph save "s\2_3h5c.gph", replace

* for h = 9
reg rel_post t if abs(pov)<=9, vce(hc2)
matrix tbl[8,1] = _b["t"]
matrix tbl[9,1] = _se["t"]
capture drop pred
predict pred if abs(pov)<=9
twoway scatter pred pov if abs(pov)<=9, title("Order 0, h = 9")
graph save "s\2_3h9a.gph", replace
```

```
reg rel_post t p1t p1u if abs(pov) <=9, vce(hc2)
matrix tbl[8,2] = _b["t"]
matrix tbl[9,2] = _se["t"]
capture drop pred
predict pred if abs(pov)<=9
twoway scatter pred pov if abs(pov)<=9, title("Order 1, h = 9")
graph save "s\2_3h9b.gph", replace

reg rel_post t p1t p1u p2t p2u if abs(pov)<=9, vce(hc2)
matrix tbl[8,3] = _b["t"]
matrix tbl[9,3] = _se["t"]
capture drop pred
predict pred if abs(pov)<=9
twoway scatter pred pov if abs(pov)<=9, title("Order 2, h = 9")
graph save "s\2_3h9c.gph", replace

* for h = 18
reg rel_post t if abs(pov)<=18, vce(hc2)
matrix tbl[11,1] = _b["t"]
matrix tbl[12,1] = _se["t"]
capture drop pred
predict pred if abs(pov)<=18
twoway scatter pred pov if abs(pov)<=18, title("Order 0, h = 18")
graph save "s\2_3h18a.gph", replace

reg rel_post t p1t p1u if abs(pov)<=18, vce(hc2)
matrix tbl[11,2] = _b["t"]
matrix tbl[12,2] = _se["t"]
capture drop pred
predict pred if abs(pov)<=18
twoway scatter pred pov if abs(pov)<=18, title("Order 1, h = 18")
graph save "s\2_3h18b.gph", replace

reg rel_post t p1t p1u p2t p2u if abs(pov)<=18, vce(hc2)
matrix tbl[11,3] = _b["t"]
matrix tbl[12,3] = _se["t"]
capture drop pred
predict pred if abs(pov)<=18
twoway scatter pred pov if abs(pov)<=18, title("Order 2, h = 18")
graph save "s\2_3h18c.gph", replace

* combine all graphs
graph combine "s\2_3h1a.gph" "s\2_3h1b.gph" "s\2_3h1c.gph" ///
        "s\2_3h5a.gph" "s\2_3h5b.gph" "s\2_3h5c.gph" ///
        "s\2_3h9a.gph" "s\2_3h9b.gph" "s\2_3h9c.gph" ///
        "s\2_3h18a.gph" "s\2_3h18b.gph" "s\2_3h18c.gph", c(3)
graph export "s\2_3s.png", replace

* write table to LaTeX
mat2txt, matrix(tbl) saving("s\2_3.txt") format(%9.4f) replace

*******************************************************************************
* Q2.3.1: MSE-optimal RD estimators
```

```
* just manually typing these into LaTeX
* 'all' gives us the three methods, p and q tell the polynomial orders
rdrobust rel_post pov, p(0) q(1) c(0) all
rdrobust rel_post pov, p(1) q(2) c(0) all
rdrobust rel_post pov, p(2) q(3) c(0) all

*******************************************************************************
* Q2.3.2: Robustness checks

** a) Placebo outcome tests
* check preintervention related mortality
rdrobust rel_pre pov, p(1) q(2) c(0) all
* check postintervention unrelated mortality
rdrobust inj_post pov, p(1) q(2) c(0) all

** b) Bandwidth and Kernel sensitivity
* empty matrix to fill with results
matrix tbl = J(3, 10, .)

* loop over bandwidths, calc for three different kernels
forvalues h = 1(1)10 {
    quiet rdrobust rel_post pov, h(`h') kernel(tri) p(1) q(2) c(0) all
        matrix tbl[1, `h'] = round(e(tau_bc), .01)
        quiet rdrobust rel_post pov, h(`h') kernel(uni) p(1) q(2) c(0) all
        matrix tbl[2, `h'] = round(e(tau_bc), .01)
        quiet rdrobust rel_post pov, h(`h') kernel(epa) p(1) q(2) c(0) all
        matrix tbl[3, `h'] = round(e(tau_bc), .01)
}

* write to LaTeX
mat2txt, matrix(tbl) saving("s\3_2b.txt") format(%9.4f) replace

** c) "donut hole" approach
* first sort by closeness for povrate to 0
gen pov_abs = abs(pov)
sort pov_abs
drop pov_abs

* empty matrix to fill with results
matrix tbl = J(1, 10, .)

* loop over 10 l's
forvalues l = 1(1)10 {
    quiet rdrobust rel_post pov if _n>`l', p(1) q(2) c(0) all
        matrix tbl[1, `l'] = round(e(tau_bc), .01)
}

* write to LaTeX
mat2txt, matrix(tbl) saving("s\3_2c.txt") format(%9.4f) replace

** d) Placebo cutoff approach
* empty matrix to fill with results
matrix tbl = J(2, 10, .)
```

```
* loop over various cutoffs, save point estimates and p-values
forvalues c = -10(2)10 {
    if `c' != 0 {
            quiet rdrobust rel_post pov, c(`c') p(1) q(2) all
                * to know which matrix position to store in
                if `c' < 0 {
                    local i = `c'/2 + 6
                }
                if `c' > 0 {
                    local i = `c'/2 + 5
                }
                matrix tbl[1, `i'] = round(e(tau_bc), .01)
                matrix tbl[2, `i'] = round(e(pv_rb), .01)
        }
}

* write to LaTeX
mat2txt, matrix(tbl) saving("s\3_2d.txt") format(%9.4f) replace

*******************************************************************************
* Q2.4.1: Local Randomization Methods - Window Selection

* select windows using the four included methods
rdwinselect pov inj_post rel_pre, ///
    cutoff(0) wmin(0.1) wstep(0.2) nwindows(20) statistic(diffmeans)
* recommended window is 1.7

rdwinselect pov inj_post rel_pre, ///
    cutoff(0) wmin(0.1) wstep(0.2) nwindows(20) statistic(ksmirnov)
* recommended window is 1.9

rdwinselect pov inj_post rel_pre, ///
    cutoff(0) wmin(0.1) wstep(0.2) nwindows(20) statistic(ranksum)
* recommended window is 1.7

rdwinselect pov inj_post rel_pre, ///
    cutoff(0) wmin(0.1) wstep(0.2) nwindows(20) statistic(hotelling)
* recommended window is 1.9

* plot based on the selected binwidth of 1.7 (using diffmeans)
rdplot rel_post pov if abs(pov)<=1.7, c(0) p(0) binselect(es)
graph export "s\4_1s.png", replace

*******************************************************************************
* Q2.4.2: Local Randomization Methods - Basic Analysis

* do the randomization analysis using window from above
rdrandinf rel_post pov, wl(-1.7) wr(1.7) seed(22)

*******************************************************************************
* Q2.4.2: Local Randomization Methods - Sensitivity Analysis

* empty matrix to fill with results
matrix tbl = J(4, 10, .)
```

```
* loop over window values, store results
forvalues i = 1(1)10 {
        local win = 0.8+(`i'-1)*0.2
        matrix tbl[1,`i'] = `win'

    reg rel_post t if abs(pov)<=`win', vce(hc2)
        matrix temp = r(table)

    matrix tbl[2, `i'] = temp["b","t"]
    matrix tbl[3, `i'] = temp["se","t"]
    matrix tbl[4, `i'] = temp["pvalue","t"]
}

* write to LaTeX
mat2txt, matrix(tbl) saving("s\4_3.txt") format(%9.4f) replace

*******************************************************************************
log close
*******************************************************************************
```

B R Code

```
#############################################################################################
# Author: Paul R. Organ
# Purpose: ECON 675, PS6
# Last Update: Dec 4, 2018
#############################################################################################
# Preliminaries
options(stringsAsFactors = F)

# packages
require(tidyverse) # data cleaning and manipulation
require(magrittr)  # syntax
require(ggplot2)   # plots
require(sandwich)  # robust standard errors
require(xtable)    # tables for LaTeX
require(stargazer) # tables for LaTeX
require(rdrobust)  # RD stuff
require(rdlocrand) # RD stuff
require(rddensity) # RD stuff
require(gridExtra) # plotting multiple plots

source('multiplot.R') # multi-panel plots

options(scipen = 999)
setwd('C:/Users/prorgan/Box/Classes/Econ 675/Problem Sets/PS6')

# read in dataset
df <- read_csv('HeadStart.csv')

# rename for ease of use
df %<>% rename(pov = povrate60,
               rel_post = mort_related_post,
               rel_pre  = mort_related_pre,
               inj_post = mort_injury_post)

#############################################################################################
# Question 2) The Effect of Head Start on Child Mortality
#############################################################################################
# Q2.1.1: RD Plots

# four plots: {evenly-spaced, quantile-spaced} X (IMSE-optimal,data var}
png('r/1_1a.png')
rdplot(df$rel_pre, df$pov, binselect = 'es',
       title = 'Evenly-spaced binning, IMSE-optimal',
       x.label = 'Poverty Rate (Normalized)',
       y.label = 'Average Mortality Rate (Pre)')
dev.off()

png('r/1_1b.png')
rdplot(df$rel_pre, df$pov, binselect = 'esmv',
       title = 'Evenly-spaced binning, variability-mimicking',
       x.label = 'Poverty Rate (Normalized)',
       y.label = 'Average Mortality Rate (Pre)')
```

```
dev.off()

png('r/1_1c.png')
rdplot(df$rel_pre, df$pov, binselect = 'qs',
       title = 'Quantile-spaced binning, IMSE-optimal',
       x.label = 'Poverty Rate (Normalized)',
       y.label = 'Average Mortality Rate (Pre)')
dev.off()

png('r/1_1d.png')
rdplot(df$rel_pre, df$pov, binselect = 'qsmv',
       title = 'Quantile-spaced binning, variability mimicking',
       x.label = 'Poverty Rate (Normalized)',
       y.label = 'Average Mortality Rate (Pre)')
dev.off()

# combine in PPT for use in LaTeX

###############################################################################
# Q2.1.2: Falsification Tests
# falsification of RD design using three methods

# (i) histogram plots
df %<>% mutate(treated = (pov > 0), t = treated * 1)

p1_2 <- ggplot(df, aes(x = pov, fill = treated, color = treated)) +
  geom_histogram() + theme_minimal() +
  labs(y = 'Frequency', x = 'Poverty Rate (Normalized)')
ggsave('r/1_2r.png', width = 6, height = 4)

# (ii) binomial tests
rdwinselect(df$pov, wmin = .05, wstep = .05, nwindows = 100)

# (iii) continuity-in-density tests
rddensity(df$pov, all = T) %>% summary
rddensity(df$pov, all = T, h = c(5,5)) %>% summary
rddensity(df$pov, all = T, h = c(1,1)) %>% summary

rm(p1_2); gc()

###############################################################################
# Q2.2.1: constant treatment effect model (additive separable effect)

# generate variables for polynomials of order 3-6
df %<>% mutate(p2=pov^2, p3=pov^3, p4=pov^4, p5=pov^5, p6=pov^6)

# run regressions
r2_1a <- lm(rel_post ~ t + pov + p2 + p3, df)
r2_1b <- lm(rel_post ~ t + pov + p2 + p3 + p4, df)
r2_1c <- lm(rel_post ~ t + pov + p2 + p3 +p4 + p5, df)
r2_1d <- lm(rel_post ~ t + pov + p2 + p3 + p4 + p5 + p6, df)

# grab estimates for table
tab <- matrix(NA,2,4) %>% as.data.frame()
```

```
rownames(tab) <- c('Point Estimate', 'Standard Error')
colnames(tab) <- c('p=3', 'p=4', 'p=5', 'p=6')

# point estimates
tab[1,1] <- r2_1a$coefficients['t']
tab[1,2] <- r2_1b$coefficients['t']
tab[1,3] <- r2_1c$coefficients['t']
tab[1,4] <- r2_1d$coefficients['t']

# robust standard errors
tab[2,1] <- diag(vcovHC(r2_1a, type = "HC2")) %>% sqrt() %>% .['t']
tab[2,2] <- diag(vcovHC(r2_1b, type = "HC2")) %>% sqrt() %>% .['t']
tab[2,3] <- diag(vcovHC(r2_1c, type = "HC2")) %>% sqrt() %>% .['t']
tab[2,4] <- diag(vcovHC(r2_1d, type = "HC2")) %>% sqrt() %>% .['t']

# write to LaTeX
xtable(tab)

# generate dataframes for plotting
d2_1a <- data.frame(pov = df$pov, pred = r2_1a$fitted.values)
d2_1b <- data.frame(pov = df$pov, pred = r2_1b$fitted.values)
d2_1c <- data.frame(pov = df$pov, pred = r2_1c$fitted.values)
d2_1d <- data.frame(pov = df$pov, pred = r2_1d$fitted.values)

# generate plot for each order
p2_1a <- ggplot(d2_1a, aes(x = pov, y = pred)) + geom_point() +
  theme_minimal() + labs(x = 'Pov Rate 1960 (Normalized)', y = 'Fitted Values',
                         title = 'Order 3')
p2_1b <- ggplot(d2_1b, aes(x = pov, y = pred)) + geom_point() +
  theme_minimal() + labs(x = 'Pov Rate 1960 (Normalized)', y = 'Fitted Values',
                         title = 'Order 4')
p2_1c <- ggplot(d2_1c, aes(x = pov, y = pred)) + geom_point() +
  theme_minimal() + labs(x = 'Pov Rate 1960 (Normalized)', y = 'Fitted Values',
                         title = 'Order 5')
p2_1d <- ggplot(d2_1d, aes(x = pov, y = pred)) + geom_point() +
  theme_minimal() + labs(x = 'Pov Rate 1960 (Normalized)', y = 'Fitted Values',
                         title = 'Order 6')

# combine plots
png('r/2_1r.png')
multiplot(p2_1a, p2_1c, p2_1b, p2_1d, cols=2)
dev.off()

# clean up
rm(list=ls(pattern="r2_|d2_|p2_|tab")); gc()

##############################################################################
# Q2.2.2: heterogeneous treatment effect model (fully interacted effect)

# define new interacted terms for polynomials
df %<>% mutate(p1t = pov*t, p1u = pov*(1-t), p2t = p2*t, p2u = p2*(1-t),
               p3t = p3*t, p3u = p3*(1-t), p4t = p4*t, p4u = p4*(1-t),
               p5t = p5*t, p5u = p5*(1-t), p6t = p6*t, p6u = p6*(1-t))
```

```
# define reg equations
v3 <- c('t', 'p1t', 'p1u', 'p2t', 'p2u', 'p3t', 'p3u')
v4 <- c(v3, 'p4t', 'p4u')
v5 <- c(v4, 'p5t', 'p5u')
v6 <- c(v5, 'p6t', 'p6u')

f3 <- paste('rel_post', paste(v3, collapse = ' + '), sep = ' ~ ') %>% as.formula()
f4 <- paste('rel_post', paste(v4, collapse = ' + '), sep = ' ~ ') %>% as.formula()
f5 <- paste('rel_post', paste(v5, collapse = ' + '), sep = ' ~ ') %>% as.formula()
f6 <- paste('rel_post', paste(v6, collapse = ' + '), sep = ' ~ ') %>% as.formula()

# run regressions
r2_2a <- lm(f3, df); r2_2b <- lm(f4, df); r2_2c <- lm(f5, df); r2_2d <- lm(f6, df)

# grab estimates for table
tab <- matrix(NA,2,4) %>% as.data.frame()
rownames(tab) <- c('Point Estimate', 'Standard Error')
colnames(tab) <- c('p=3', 'p=4', 'p=5', 'p=6')

# point estimates
tab[1,1] <- r2_2a$coefficients['t']
tab[1,2] <- r2_2b$coefficients['t']
tab[1,3] <- r2_2c$coefficients['t']
tab[1,4] <- r2_2d$coefficients['t']

# robust standard errors
tab[2,1] <- diag(vcovHC(r2_2a, type = "HC2")) %>% sqrt() %>% .['t']
tab[2,2] <- diag(vcovHC(r2_2b, type = "HC2")) %>% sqrt() %>% .['t']
tab[2,3] <- diag(vcovHC(r2_2c, type = "HC2")) %>% sqrt() %>% .['t']
tab[2,4] <- diag(vcovHC(r2_2d, type = "HC2")) %>% sqrt() %>% .['t']

# write to LaTeX
xtable(tab)

# generate dataframes for plotting
d2_2a <- data.frame(pov = df$pov, pred = r2_2a$fitted.values)
d2_2b <- data.frame(pov = df$pov, pred = r2_2b$fitted.values)
d2_2c <- data.frame(pov = df$pov, pred = r2_2c$fitted.values)
d2_2d <- data.frame(pov = df$pov, pred = r2_2d$fitted.values)

# generate plot for each order
p2_2a <- ggplot(d2_2a, aes(x = pov, y = pred)) + geom_point() +
  theme_minimal() + labs(x = 'Pov Rate 1960 (Normalized)', y = 'Fitted Values',
                         title = 'Order 3')
p2_2b <- ggplot(d2_2b, aes(x = pov, y = pred)) + geom_point() +
  theme_minimal() + labs(x = 'Pov Rate 1960 (Normalized)', y = 'Fitted Values',
                         title = 'Order 4')
p2_2c <- ggplot(d2_2c, aes(x = pov, y = pred)) + geom_point() +
  theme_minimal() + labs(x = 'Pov Rate 1960 (Normalized)', y = 'Fitted Values',
                         title = 'Order 5')
p2_2d <- ggplot(d2_2d, aes(x = pov, y = pred)) + geom_point() +
  theme_minimal() + labs(x = 'Pov Rate 1960 (Normalized)', y = 'Fitted Values',
                         title = 'Order 6')
```

```
# combine plots
png('r/2_2r.png')
multiplot(p2_2a, p2_2c, p2_2b, p2_2d, cols=2)
dev.off()

# clean up
rm(list=ls(pattern="r2_|d2_|p2_|v[0-9]|f[0-9]|tab")); gc()

##############################################################################################
# Q2.2.3: local parametric model (p = 0,1,2 and h = 1,5,9,18)

# for h = 1
r2_3h1a <- lm(rel_post ~ t, df %>% filter(abs(pov) <= 1))
r2_3h1b <- lm(rel_post ~ t + p1t + p1u, df %>% filter(abs(pov) <= 1))
r2_3h1c <- lm(rel_post ~ t + p1t + p1u + p2t + p2u, df %>% filter(abs(pov) <= 1))

# grab estimates for table
tab <- matrix(NA,12,4) %>% as.data.frame()
tab[,1] <- c('h=1','b','se','h=5','b','se',
             'h=9','b', 'se','h=18', 'b', 'se')
colnames(tab) <- c('','p=0', 'p=1', 'p=2')

# point estimates
tab[2,2] <- r2_3h1a$coefficients['t']
tab[2,3] <- r2_3h1b$coefficients['t']
tab[2,4] <- r2_3h1c$coefficients['t']

# robust standard errors
tab[3,2] <- diag(vcovHC(r2_3h1a, type = "HC2")) %>% sqrt() %>% .['t']
tab[3,3] <- diag(vcovHC(r2_3h1b, type = "HC2")) %>% sqrt() %>% .['t']
tab[3,4] <- diag(vcovHC(r2_3h1c, type = "HC2")) %>% sqrt() %>% .['t']

# generate dataframes for plotting
d2_3h1a <- data.frame(pov = df$pov[abs(df$pov) <= 1], pred = r2_3h1a$fitted.values)
d2_3h1b <- data.frame(pov = df$pov[abs(df$pov) <= 1], pred = r2_3h1b$fitted.values)
d2_3h1c <- data.frame(pov = df$pov[abs(df$pov) <= 1], pred = r2_3h1c$fitted.values)

# generate plot for each order
p2_3h1a <- ggplot(d2_3h1a, aes(x = pov, y = pred)) + geom_point() +
  theme_minimal() + labs(x = 'Pov Rate 1960 (Normalized)', y = 'Fitted Values',
                         title = 'Order 0, h = 1')
p2_3h1b <- ggplot(d2_3h1b, aes(x = pov, y = pred)) + geom_point() +
  theme_minimal() + labs(x = 'Pov Rate 1960 (Normalized)', y = 'Fitted Values',
                         title = 'Order 1, h = 1')
p2_3h1c <- ggplot(d2_3h1c, aes(x = pov, y = pred)) + geom_point() +
  theme_minimal() + labs(x = 'Pov Rate 1960 (Normalized)', y = 'Fitted Values',
                         title = 'Order 2, h = 1')

# for h = 5
r2_3h5a <- lm(rel_post ~ t, df %>% filter(abs(pov) <= 5))
r2_3h5b <- lm(rel_post ~ t + p1t + p1u, df %>% filter(abs(pov) <= 5))
r2_3h5c <- lm(rel_post ~ t + p1t + p1u + p2t + p2u, df %>% filter(abs(pov) <= 5))

# point estimates
```

```
tab[5,2] <- r2_3h5a$coefficients['t']
tab[5,3] <- r2_3h5b$coefficients['t']
tab[5,4] <- r2_3h5c$coefficients['t']

# robust standard errors
tab[6,2] <- diag(vcovHC(r2_3h5a, type = "HC2")) %>% sqrt() %>% .['t']
tab[6,3] <- diag(vcovHC(r2_3h5b, type = "HC2")) %>% sqrt() %>% .['t']
tab[6,4] <- diag(vcovHC(r2_3h5c, type = "HC2")) %>% sqrt() %>% .['t']

# generate dataframes for plotting
d2_3h5a <- data.frame(pov = df$pov[abs(df$pov) <= 5], pred = r2_3h5a$fitted.values)
d2_3h5b <- data.frame(pov = df$pov[abs(df$pov) <= 5], pred = r2_3h5b$fitted.values)
d2_3h5c <- data.frame(pov = df$pov[abs(df$pov) <= 5], pred = r2_3h5c$fitted.values)

# generate plot for each order
p2_3h5a <- ggplot(d2_3h5a, aes(x = pov, y = pred)) + geom_point() +
  theme_minimal() + labs(x = 'Pov Rate 1960 (Normalized)', y = 'Fitted Values',
                         title = 'Order 0, h = 5')
p2_3h5b <- ggplot(d2_3h5b, aes(x = pov, y = pred)) + geom_point() +
  theme_minimal() + labs(x = 'Pov Rate 1960 (Normalized)', y = 'Fitted Values',
                         title = 'Order 1, h = 5')
p2_3h5c <- ggplot(d2_3h5c, aes(x = pov, y = pred)) + geom_point() +
  theme_minimal() + labs(x = 'Pov Rate 1960 (Normalized)', y = 'Fitted Values',
                         title = 'Order 2, h = 5')

# for h = 9
r2_3h9a <- lm(rel_post ~ t, df %>% filter(abs(pov) <= 9))
r2_3h9b <- lm(rel_post ~ t + p1t + p1u, df %>% filter(abs(pov) <= 9))
r2_3h9c <- lm(rel_post ~ t + p1t + p1u + p2t + p2u, df %>% filter(abs(pov) <= 9))

# point estimates
tab[8,2] <- r2_3h9a$coefficients['t']
tab[8,3] <- r2_3h9b$coefficients['t']
tab[8,4] <- r2_3h9c$coefficients['t']

# robust standard errors
tab[9,2] <- diag(vcovHC(r2_3h9a, type = "HC2")) %>% sqrt() %>% .['t']
tab[9,3] <- diag(vcovHC(r2_3h9b, type = "HC2")) %>% sqrt() %>% .['t']
tab[9,4] <- diag(vcovHC(r2_3h9c, type = "HC2")) %>% sqrt() %>% .['t']

# generate dataframes for plotting
d2_3h9a <- data.frame(pov = df$pov[abs(df$pov) <= 9], pred = r2_3h9a$fitted.values)
d2_3h9b <- data.frame(pov = df$pov[abs(df$pov) <= 9], pred = r2_3h9b$fitted.values)
d2_3h9c <- data.frame(pov = df$pov[abs(df$pov) <= 9], pred = r2_3h9c$fitted.values)

# generate plot for each order
p2_3h9a <- ggplot(d2_3h9a, aes(x = pov, y = pred)) + geom_point() +
  theme_minimal() + labs(x = 'Pov Rate 1960 (Normalized)', y = 'Fitted Values',
                         title = 'Order 0, h = 9')
p2_3h9b <- ggplot(d2_3h9b, aes(x = pov, y = pred)) + geom_point() +
  theme_minimal() + labs(x = 'Pov Rate 1960 (Normalized)', y = 'Fitted Values',
                         title = 'Order 1, h = 9')
p2_3h9c <- ggplot(d2_3h9c, aes(x = pov, y = pred)) + geom_point() +
  theme_minimal() + labs(x = 'Pov Rate 1960 (Normalized)', y = 'Fitted Values',
```

```
                        title = 'Order 2, h = 9')

# for h = 18
r2_3h18a <- lm(rel_post ~ t, df %>% filter(abs(pov) <= 18))
r2_3h18b <- lm(rel_post ~ t + p1t + p1u, df %>% filter(abs(pov) <= 18))
r2_3h18c <- lm(rel_post ~ t + p1t + p1u + p2t + p2u, df %>% filter(abs(pov) <= 18))

# point estimates
tab[11,2] <- r2_3h18a$coefficients['t']
tab[11,3] <- r2_3h18b$coefficients['t']
tab[11,4] <- r2_3h18c$coefficients['t']

# robust standard errors
tab[12,2] <- diag(vcovHC(r2_3h18a, type = "HC2")) %>% sqrt() %>% .['t']
tab[12,3] <- diag(vcovHC(r2_3h18b, type = "HC2")) %>% sqrt() %>% .['t']
tab[12,4] <- diag(vcovHC(r2_3h18c, type = "HC2")) %>% sqrt() %>% .['t']

# generate dataframes for plotting
d2_3h18a <- data.frame(pov = df$pov[abs(df$pov) <= 18], pred = r2_3h18a$fitted.values)
d2_3h18b <- data.frame(pov = df$pov[abs(df$pov) <= 18], pred = r2_3h18b$fitted.values)
d2_3h18c <- data.frame(pov = df$pov[abs(df$pov) <= 18], pred = r2_3h18c$fitted.values)

# generate plot for each order
p2_3h18a <- ggplot(d2_3h18a, aes(x = pov, y = pred)) + geom_point() +
  theme_minimal() + labs(x = 'Pov Rate 1960 (Normalized)', y = 'Fitted Values',
                         title = 'Order 0, h = 18')
p2_3h18b <- ggplot(d2_3h18b, aes(x = pov, y = pred)) + geom_point() +
  theme_minimal() + labs(x = 'Pov Rate 1960 (Normalized)', y = 'Fitted Values',
                         title = 'Order 1, h = 18')
p2_3h18c <- ggplot(d2_3h18c, aes(x = pov, y = pred)) + geom_point() +
  theme_minimal() + labs(x = 'Pov Rate 1960 (Normalized)', y = 'Fitted Values',
                         title = 'Order 2, h = 18')

# combine all and check
comb <- grid.arrange(p2_3h1a, p2_3h1b, p2_3h1c,
                     p2_3h5a, p2_3h5b, p2_3h5c,
                     p2_3h9a, p2_3h9b, p2_3h9c,
                     p2_3h18a, p2_3h18b, p2_3h18c, ncol=3)
ggsave('r/2_3r.png', plot = comb, width = 6, height = 9)

# write table to LaTeX
print(xtable(tab), include.rownames=F)

# clean up
rm(list=ls(pattern="d2_|p2_|r2_|comb|tab")); gc()

################################################################################
# Q2.3.1: MSE-optimal RD estimators

# just manually typings these into LaTeX
# 'all=T' gives the three methods
# p and q tell the polynomial orders to use
rdrobust(df$rel_post, df$pov, p = 0, q = 1, c = 0, all = T) %>% summary()
rdrobust(df$rel_post, df$pov, p = 1, q = 2, c = 0, all = T) %>% summary()
```

```
rdrobust(df$rel_post, df$pov, p = 2, q = 3, c = 0, all = T) %>% summary()

################################################################################
# Q2.3.2: Robustness checks

# a) Placebo outcome tests

# check preintervention related mortality
rdrobust(df$rel_pre, df$pov, p =1, q = 2, c = 0, all = T) %>% summary()
# check postintervention unrelated mortality
rdrobust(df$inj_post, df$pov, p =1, q = 2, c = 0, all = T) %>% summary()

# b) Bandwidth and Kernel sensitivity
# empty matrix to fill with results
tbl <- matrix(NA,3,10)

# loop over 10 bandwidths, for three kernels, grab point estimates
for(h in 1:10){
  tbl[1,h] <- rdrobust(df$rel_post, df$pov, p = 1, q = 2, c = 0, h=h,
                       all = T, kernel = 'tri')$Estimate[2]
  tbl[2,h] <- rdrobust(df$rel_post, df$pov, p =1, q = 2, c = 0, h=h,
                       all = T, kernel = 'uni')$Estimate[2]
  tbl[3,h] <- rdrobust(df$rel_post, df$pov, p =1, q = 2, c = 0, h=h,
                       all = T, kernel = 'epa')$Estimate[2]
}

# clean up and write to LaTeX
tbl %<>% as.data.frame()
rownames(tbl) <- c('Triangular', 'Uniform', 'Epanechnikov')
colnames(tbl) <- 1:10
xtable(tbl)

# c) "donut hole" approach
# sort by proximity to 0, and number them
df %<>% mutate(pov_abs = abs(pov)) %>% arrange(pov_abs)
df$abs_pov_rank <- 1:nrow(df)

# empty matrix to fill with results
tbl <- matrix(NA, 1, 10)

# loop over 10 l's
for(l in 1:10){
  tbl[1,l] <- rdrobust(df$rel_post[df$abs_pov_rank > l], df$pov[df$abs_pov_rank > l],
                       p = 1, q = 2, c = 0, all = T)$Estimate[2]
}

# clean up and write to LaTeX
tbl %<>% as.data.frame()
colnames(tbl) <- 1:10
xtable(tbl)

# d) Placebo cutoff approach
cutoffs <- c(-10, -8, -6, -4, -2, 2, 4, 6, 8, 10)
```

```
# empty matrix to fill with results
tbl <- matrix(NA, 2, 10)

# loop over 10 cutoffs, save point estimates and p-values
for(i in 1:10){
  temp <- rdrobust(df$rel_post, df$pov, p = 1, q = 2, c = cutoffs[i], all = T)
  tbl[1,i] <- temp$Estimate[2] # bias-corrected
  tbl[2,i] <- temp$pv[3] # robust
}

# clean up and write to LaTeX
tbl %<>% as.data.frame()
rownames(tbl) <- c('Point estimate', 'p-value')
colnames(tbl) <- 1:10
xtable(tbl)

# clean up Q2.3
rm(tbl, temp, cutoffs, h, i, l); gc()

################################################################################
# Q2.4.1: Local Randomization Methods - Window Selection

pre <- df[,c('inj_post', 'rel_pre')] %>% as.matrix()

# select windows using the four included methods
rdwinselect(df$pov, X = pre, cutoff = 0, wmin = 0.1, wstep = 0.2,
            nwindows = 20, statistic = 'diffmeans')
# recommended window is 1.7

rdwinselect(df$pov, X = pre, cutoff = 0, wmin = 0.1, wstep = 0.2,
            nwindows = 20, statistic = 'ksmirnov')
# recommended window is 1.9

rdwinselect(df$pov, X = pre, cutoff = 0, wmin = 0.1, wstep = 0.2,
            nwindows = 20, statistic = 'ranksum')
# recommended window is 1.7

rdwinselect(df$pov, X = pre, cutoff = 0, wmin = 0.1, wstep = 0.2,
            nwindows = 20, statistic = 'hotelling')
# recommended window is 1.9

# plot based on the selected binwidth of 1.7 (using diffmeans)
png('r/4_1r.png')
rdplot(df$rel_post[df$pov_abs<=1.7], df$pov[df$pov_abs<=1.7],
       binselect = 'es', p = 0,
       title = '',
       x.label = 'Poverty Rate (Normalized)',
       y.label = 'Average Mortality Rate (Post)')
dev.off()

################################################################################
# Q2.4.2: Local Randomization Methods - Basic Analysis

# do the randomization analysis using window from above
```

```
set.seed(22)
rdrandinf(df$rel_post, df$pov, wl = -1.7, wr = 1.7)

##############################################################################
# Q2.4.3: Local Randomization Methods - Sensitivity Analysis

# empty matrix to fill with results
tbl <- matrix(NA,4,10)

# loop over window values, store results
for(i in 1:10){
  win <- 0.8 + (i-1)*0.2
  tbl[1,i] <- win

  reg <- lm(rel_post ~ t, df, subset = (pov_abs <= win))

  tbl[2,i] <- reg$coefficients['t']
  tbl[3,i] <- diag(vcovHC(reg, type = "HC2")) %>% sqrt() %>% .['t']
  tbl[4,i] <- summary(reg)$coefficients['t',4]
}

# write to LaTeX
tbl %<>% as.data.frame()
colnames(tbl) <- tbl[1,]
tbl <- tbl[2:4,]
rownames(tbl) <- c('Point estimate', 'Standard error', 'p-value')
xtable(tbl)

##############################################################################
```