# Problem Set 6

Program Evaluation and Causal Inference

Christian Birchler    Fenqi Guo    Mingrui Zhang    Wenjie Tu    Zunhan Zhang

Spring Semester 2021

Names are listed in alphabetical order

## Analysis of a Regression Discontinuity

### 1. Identification in the RDD with constant treatment effect

#### 1(a)

The cut-off might not be strictly implemented. The probability of treatment assignment changes discontinuously by less than 100% at the cut-off point $c$ and there would be both treated and untreated observations on either side of the cut-off point. Namely, $D \neq Z = \mathbf{1}[X \geq c]$. We therefore have a fuzzy design.

$$
\begin{aligned}
Y &= \beta_0 + \beta_1 D + \delta_1 X + U \\
Z &= \mathbf{1}[X \geq c]
\end{aligned}
$$

$$
\begin{cases}
D & \text{treatment status} \\
Z & \text{treatment assignment}
\end{cases}
$$

$$
\begin{cases}
D = Z = \mathbf{1}[X \geq c] & \text{sharp RDD} \\
D \neq Z = \mathbf{1}[X \geq c] & \text{fuzzy RDD}
\end{cases}
$$

Assumptions:

- Constant treatment effects assumption (i.e., $\beta_i = \beta \quad \forall i$). This assumption implies that if we instrument $D$ with $Z$, we will be able to capture the treatment effect using the given model. Namely, $\mathbb{E}(U|Z) = 0$ or $\mathrm{Cov}(U, Z) = 0$

$$
\begin{aligned}
Y &= \beta_0 + \beta_1 D + \delta_1 X + U \\
\mathrm{Cov}(Y, Z) &= \mathrm{Cov}(\beta_0 + \beta_1 D + \delta_1 X + U, Z) \\
\mathrm{Cov}(Y, Z) &= \beta_1 \mathrm{Cov}(D, Z) \\
\beta_1 &= \frac{\mathrm{Cov}(Y, Z)}{\mathrm{Cov}(D, Z)} \\
\beta_1 &= \frac{\mathbb{E}(Y|Z=1) - \mathbb{E}(Y|Z=0)}{\mathbb{E}(D|Z=1) - \mathbb{E}(D|Z=0)}
\end{aligned}
$$

- $\mathbb{E}(Y|X=x)$ is continuous in $x$. This assumption implies

$$\begin{cases} \mathbb{E}(Y|X<c) = \lim_{x\uparrow c}\mathbb{E}(Y|X=x) \\ \mathbb{E}(D|X<c) = \lim_{x\uparrow c}\mathbb{E}(D|X=x) \end{cases}$$

Put these two assumptions together, we can obtain

$$\begin{aligned} \beta_1 &= \frac{\mathbb{E}(Y|Z=1)-\mathbb{E}(Y|Z=0)}{\mathbb{E}(D|Z=1)-\mathbb{E}(D|Z=0)} \\ &= \frac{\mathbb{E}(Y|X=c)-\mathbb{E}(Y|X<c)}{\mathbb{E}(D|X=c)-\mathbb{E}(D|X<c)} \\ &= \frac{\mathbb{E}(Y|X=c)-\lim_{x\uparrow c}\mathbb{E}(Y|X=x)}{\mathbb{E}(D|X=c)-\lim_{x\uparrow c}\mathbb{E}(D|X=x)} \end{aligned}$$

**1(b)**

In a sharp design, treatment probability changes from 0 to 100% at the cut-off point. All units in the sample are compliers. Namely, $D = Z = \mathbf{1}[X \geq c]$

$$\begin{cases} \lim_{x\uparrow c}\mathbb{E}(D|X=x) = Pr(D|X<c) = 0 \\ \mathbb{E}(D|X=c) = Pr(D|X=c) = 1 \end{cases}$$

$$\begin{aligned} \mathbb{E}(Y|X=c) &= \mathbb{E}(\beta_0+\beta_1 D+\delta_1 X+U|X=c) \\ &= \beta_0+\beta_1+\delta_1\cdot c \end{aligned}$$

$$\begin{aligned} \lim_{x\uparrow c}\mathbb{E}(Y|X=x) &= \lim_{x\uparrow c}\mathbb{E}(\beta_0+\beta_1 D+\delta_1 X+U|X=x) \\ &= \beta_0+\delta_1\cdot c \end{aligned}$$

$$\begin{aligned} \Delta^{SRD} &= \frac{\mathbb{E}(Y|X=c)-\lim_{x\uparrow c}\mathbb{E}(Y|X=x)}{\mathbb{E}(D|X=c)-\lim_{x\uparrow c}\mathbb{E}(D|X=x)} \\ &= \frac{\beta_0+\beta_1+\delta_1\cdot c-(\beta_0+\delta_1\cdot c)}{1-0} \\ &= \beta_1 = \beta \end{aligned}$$

## 2. Fuzzy RDD is IV

**2(a)**

$$Pr(D_i=1|X_i) = \begin{cases} g_1(X_i) & \text{if} \quad X_i \geq c \\ g_0(X_i) & \text{if} \quad X_i < c \end{cases}$$

In a fuzzy RDD, $0 < g_0(X_i) < g_1(X_i) < 1$. This implies that there are always some units below the threshold $X_i < c$ in the observed treatment group $Pr(D_i = 1|X_i)$. Therefore, the observed treatment indicator $D_i$ is not "clean" and we need to use IV to solve this endogeneity issue.

$$\textbf{Structural equation:} \quad Y_i = \beta_0+\beta_1 D_i+\beta_2 X_i+\nu_i$$

**2(b)**

$$Z_i = \mathbf{1}[X_i \geq c]$$

$Z_i$ is a binary encouragement indicator that captures whether units are above threshold or below the threshold $c$.

In the first stage, we instrument $D_i$ with a dummy $\mathbf{1}[X_i \geq c]$,

$$\textbf{First stage:} \quad D_i = \alpha_0 + \alpha_1 \mathbf{1}[X_i \geq c] + \alpha_2 X_i + \eta_i$$

Plug first-stage equation into the structural equation,

$$\begin{aligned} Y_i &= \beta_0 + \beta_1 D_i + \delta_1 X_i + \nu_i \\ &= \beta_0 + \beta_1(\alpha_0 + \alpha_1 \mathbf{1}[X_i \geq c] + \alpha_2 X_i + \eta_i) + \beta_2 X_i + \nu_i \\ &= \beta_0 + \alpha_0\beta_1 + \alpha_1\beta_1 \mathbf{1}[X_i \geq c] + (\alpha_2\beta_1 + \beta_2)X_i + \beta_1\eta_i + \nu_i \end{aligned}$$

$$Y_i = \underbrace{\beta_0 + \alpha_0\beta_1}_{\gamma_0} + \underbrace{\alpha_1\beta_1}_{\gamma_1} \mathbf{1}[X_i \geq c] + \underbrace{(\alpha_2\beta_1 + \beta_2)}_{\gamma_2} X_i + \underbrace{\beta_1\eta_i + \nu_i}_{\varepsilon_i}$$

$$\textbf{Reduced-form equation:} \quad Y_i = \gamma_0 + \gamma_1 \mathbf{1}[X_i \geq c] + \gamma_2 X_i + \varepsilon_i$$

$$\begin{aligned} \text{Structural equation:} \quad & Y_i = \beta_0 + \beta_1 D_i + \beta_2 X_i + \nu_i \\ \text{First-stage equation:} \quad & D_i = \alpha_0 + \alpha_1 \mathbf{1}[X_i \geq c] + \alpha_2 X_i + \eta_i \\ \text{Second-stage equation:} \quad & Y_i = \beta_0 + \beta_1 \hat{D}_i + \beta_2 X_i + u_i \\ \text{Reduced-form equation:} \quad & Y_i = \gamma_0 + \gamma_1 \mathbf{1}[X_i \geq c] + \gamma_2 X_i + \varepsilon_i \end{aligned}$$

$$\Delta^{FRD} = \Delta^{IV} = \frac{\gamma_1}{\alpha_1} = \frac{\alpha_1\beta_1}{\alpha_1} = \beta_1$$

## 3. Replicate Ludwig and Miller (2007)

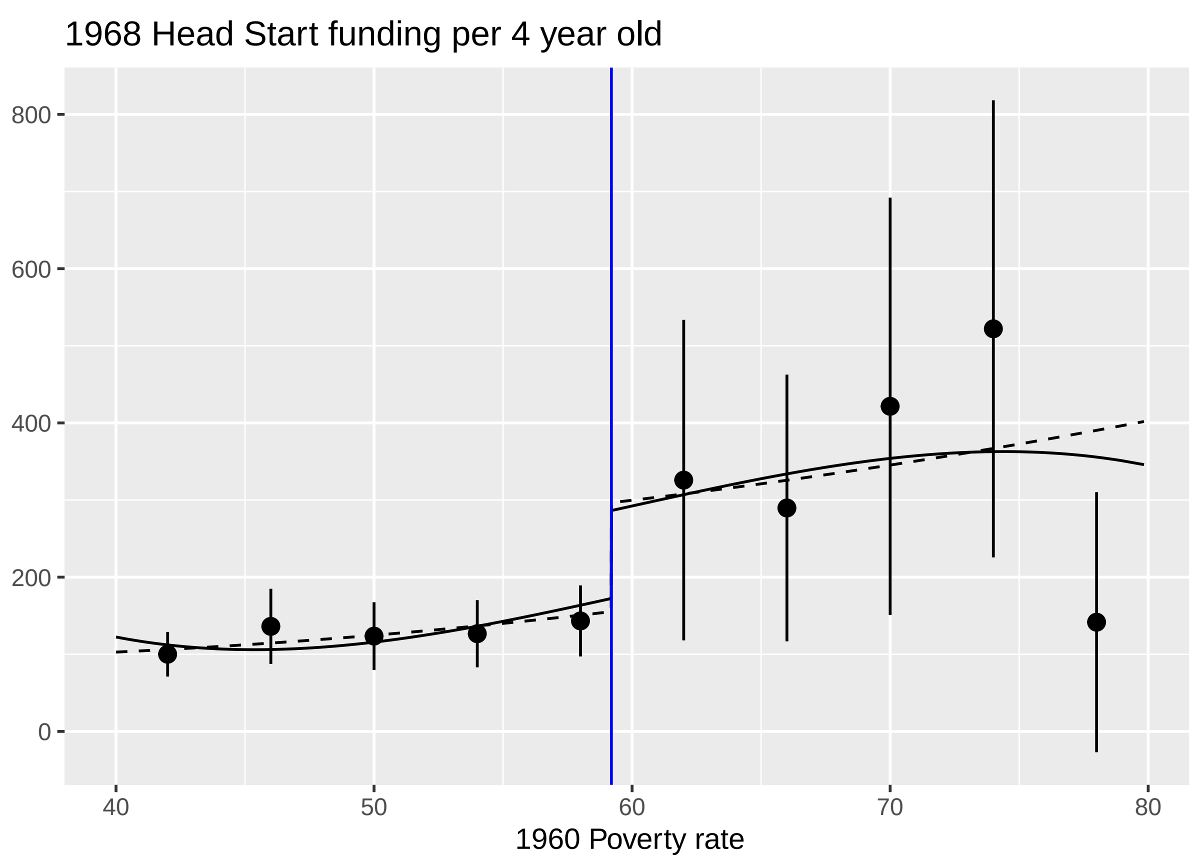
**3(b)**

```r
# import packages
library(stargazer)
library(dplyr)
library(ggplot2)
```

```r
# read data
dd <- read.csv('rdd.csv')

# remove missing values
dd <- na.omit(dd)

# figure 2

# defined in the paper
cutoff <- 59.1984
```

```r
# indicate if entry is below or above the cutoff
dd$treatment <- ifelse(dd$povrate60>=cutoff, 1, 0)

# use only entries with poverty rate >=40% and <=80%
dd.sub <- subset(dd, povrate60>=40 & povrate60<=80)

# define the bins in order to calculate the means and CIs
dd.sub$bin <- 0
dd.sub[dd.sub$povrate60>=40 & dd.sub$povrate60<44,]$bin <- 1
dd.sub[dd.sub$povrate60>=44 & dd.sub$povrate60<48,]$bin <- 2
dd.sub[dd.sub$povrate60>=48 & dd.sub$povrate60<52,]$bin <- 3
dd.sub[dd.sub$povrate60>=52 & dd.sub$povrate60<56,]$bin <- 4
dd.sub[dd.sub$povrate60>=56 & dd.sub$povrate60<60,]$bin <- 5
dd.sub[dd.sub$povrate60>=60 & dd.sub$povrate60<64,]$bin <- 6
dd.sub[dd.sub$povrate60>=64 & dd.sub$povrate60<68,]$bin <- 7
dd.sub[dd.sub$povrate60>=68 & dd.sub$povrate60<72,]$bin <- 8
dd.sub[dd.sub$povrate60>=72 & dd.sub$povrate60<76,]$bin <- 9
dd.sub[dd.sub$povrate60>=76 & dd.sub$povrate60<80,]$bin <- 10

# calculate mean, standard deviation, and number of entries inside a certain bin
get_mean_sd_n <- function(dd.sub, bin_nr){
  mean_ <- mean(dd.sub[dd.sub$bin==bin_nr,]$hsspend_per_kid_68)
  sd_ <- sd(dd.sub[dd.sub$bin==bin_nr,]$hsspend_per_kid_68)
  n_ <- nrow(dd.sub[dd.sub$bin==bin_nr,])
  return(c(mean_,sd_,n_))
}

# create special data frame for calculating the CIs
ci_data <- data.frame(bin=numeric(), mean=numeric(), sd=numeric(), n=numeric())
for (i in seq(1,10)) {
  ci_data <- rbind(ci_data, c(i, get_mean_sd_n(dd.sub, i)))
}
colnames(ci_data) <- c('bin','mean','sd','n')

# define upper and lower bounds of CIs
ci_data$ci_lower <- 0
ci_data$ci_upper <- 0

# function for calculating the CI
calc_ci <- function(mean, sd, n, z){
  lower <- mean-(z*sd/sqrt(n))
  upper <- mean+(z*sd/sqrt(n))
  return(c(lower, upper))
}

# define value for 95% CI
z_95_percent <- 1.96

# calculate the CI of each bin
for (i in seq(1,10)) {
  ci <- calc_ci(ci_data[i,]$mean, ci_data[i,]$sd, ci_data[i,]$n, z_95_percent)
  ci_data[i,]$ci_lower <- ci[1]
  ci_data[i,]$ci_upper <- ci[2]
```

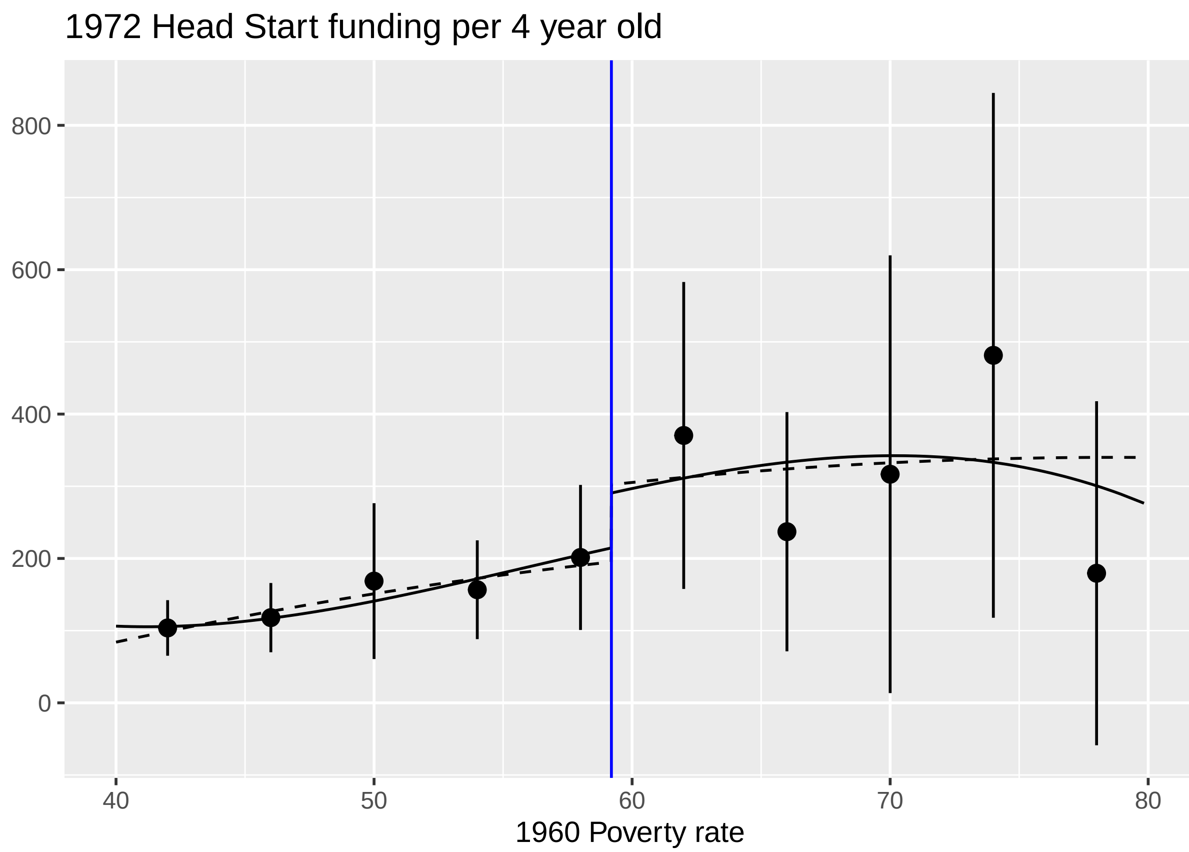
```
}

# add central poverty of each bin
ci_data$poverty <- 0
for (i in seq(1,10)) {
  ci_data[i,]$poverty <- 38+i*4
}

# use cubic as "non-parametric" and quadratic as parametric
cubic <- lm(hsspend_per_kid_68 ~ poly(povrate60, 3) + treatment, data=dd.sub)
quad <- lm(hsspend_per_kid_68 ~ poly(povrate60, 2) + treatment, data=dd.sub)

# put everything in a plot
ggplot() + geom_line(aes(x=dd.sub$povrate60, y=cubic$fitted.values)) +
  geom_line(aes(x=dd.sub$povrate60, y=quad$fitted.values), linetype="dashed") +
  geom_vline(xintercept=cutoff, col="blue") +
  geom_pointrange(aes(x=ci_data$poverty, y=ci_data$mean,
                      ymin=ci_data$ci_lower, ymax=ci_data$ci_upper)) +
  ggtitle("1968 Head Start funding per 4 year old") +
  xlab("1960 Poverty rate") +
  ylab("")
```




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

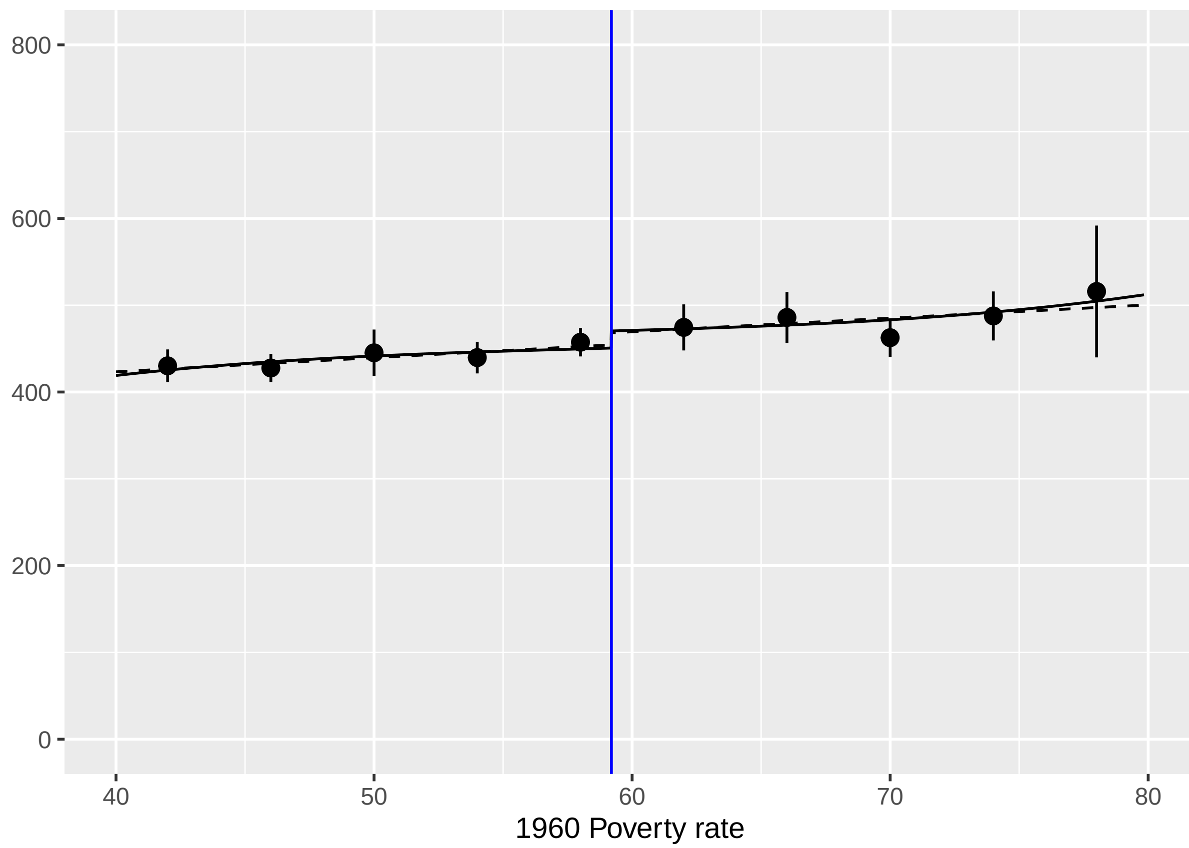
```r
# setup second plot for figure II in the paper

# calculate mean, standard deviation, and number of entries inside a certain bin
get_mean_sd_n <- function(dd.sub, bin_nr){
  mean_ <- mean(dd.sub[dd.sub$bin==bin_nr,]$hsspend_per_kid_72)
  sd_ <- sd(dd.sub[dd.sub$bin==bin_nr,]$hsspend_per_kid_72)
  n_ <- nrow(dd.sub[dd.sub$bin==bin_nr,])
  return(c(mean_,sd_,n_))
}

# create special data frame for calculating the CIs
ci_data <- data.frame(bin=numeric(), mean=numeric(), sd=numeric(), n=numeric())
for (i in seq(1,10)) {
  ci_data <- rbind(ci_data, c(i, get_mean_sd_n(dd.sub, i)))
}
colnames(ci_data) <- c('bin','mean','sd','n')

# define upper and lower bounds of CIs
ci_data$ci_lower <- 0
ci_data$ci_upper <- 0

# function for calculating the CI
calc_ci <- function(mean, sd, n, z){
  lower <- mean-(z*sd/sqrt(n))
  upper <- mean+(z*sd/sqrt(n))
  return(c(lower, upper))
}

# define value for 95% CI
z_95_percent <- 1.96

# calculate the CI of each bin
for (i in seq(1,10)) {
  ci <- calc_ci(ci_data[i,]$mean, ci_data[i,]$sd, ci_data[i,]$n, z_95_percent)
  ci_data[i,]$ci_lower <- ci[1]
  ci_data[i,]$ci_upper <- ci[2]
}

# add central poverty of each bin
ci_data$poverty <- 0
for (i in seq(1,10)) {
  ci_data[i,]$poverty <- 38+i*4
}

# use cubic as "non-parametric" and quadratic as parametric
cubic <- lm(hsspend_per_kid_72 ~ poly(povrate60, 3) + treatment, data=dd.sub)
quad <- lm(hsspend_per_kid_72 ~ poly(povrate60, 2) + treatment, data=dd.sub)

# put everything in a plot
ggplot() + geom_line(aes(x=dd.sub$povrate60, y=cubic$fitted.values)) +
  geom_line(aes(x=dd.sub$povrate60, y=quad$fitted.values), linetype="dashed") +
  geom_vline(xintercept=cutoff, col="blue") +
  geom_pointrange(aes(x=ci_data$poverty, y=ci_data$mean,
```

```
                     ymin=ci_data$ci_lower, ymax=ci_data$ci_upper)) +
  ggtitle("1972 Head Start funding per 4 year old") +
  xlab("1960 Poverty rate") +
  ylab("")
```




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

# setup for figure III in the paper

# calculate mean, standard deviation, and number of entries inside a certain bin
get_mean_sd_n <- function(dd.sub, bin_nr){
  mean_ <- mean(dd.sub[dd.sub$bin==bin_nr,]$socspend_per_cap72)
  sd_ <- sd(dd.sub[dd.sub$bin==bin_nr,]$socspend_per_cap72)
  n_ <- nrow(dd.sub[dd.sub$bin==bin_nr,])
  return(c(mean_,sd_,n_))
}

# create special data frame for calculating the CIs
ci_data <- data.frame(bin=numeric(), mean=numeric(), sd=numeric(), n=numeric())
for (i in seq(1,10)) {
  ci_data <- rbind(ci_data, c(i, get_mean_sd_n(dd.sub, i)))
}
colnames(ci_data) <- c('bin','mean','sd','n')
```

```r
# define upper and lower bounds of CIs
ci_data$ci_lower <- 0
ci_data$ci_upper <- 0

# function for calculating the CI
calc_ci <- function(mean, sd, n, z){
  lower <- mean-(z*sd/sqrt(n))
  upper <- mean+(z*sd/sqrt(n))
  return(c(lower, upper))
}

# define value for 95% CI
z_95_percent <- 1.96

# calculate the CI of each bin
for (i in seq(1,10)) {
  ci <- calc_ci(ci_data[i,]$mean, ci_data[i,]$sd, ci_data[i,]$n, z_95_percent)
  ci_data[i,]$ci_lower <- ci[1]
  ci_data[i,]$ci_upper <- ci[2]
}

# add central poverty of each bin
ci_data$poverty <- 0
for (i in seq(1,10)) {
  ci_data[i,]$poverty <- 38+i*4
}

# use cubic as "non-parametric" and quadratic as parametric
cubic <- lm(socspend_per_cap72 ~ poly(povrate60, 3) + treatment, data=dd.sub)
quad <- lm(socspend_per_cap72 ~ poly(povrate60, 2) + treatment, data=dd.sub)

# put everything in a plot
ggplot() + geom_line(aes(x=dd.sub$povrate60, y=cubic$fitted.values)) +
  geom_line(aes(x=dd.sub$povrate60, y=quad$fitted.values), linetype="dashed") +
  geom_vline(xintercept=cutoff, col="blue") +
  geom_pointrange(aes(x=ci_data$poverty, y=ci_data$mean,
                      ymin=ci_data$ci_lower, ymax=ci_data$ci_upper)) +
  xlab("1960 Poverty rate") +
  ylab("") +
  ylim(0,800)
```




**3(c)**

```
# create dummy variable
dd$dummy <- ifelse(dd$povrate60 < 59.1984, 0,1)
# create new rates
dd$rate <- dd$povrate60 - 59.1984
dd$ratesq <- dd$rate^2
dd$ratecub <- dd$rate^3
dd$ratedum <- dd$rate*dd$dummy
dd$ratesqdum <- dd$ratesq*dd$dummy
dd$ratecubdum <- dd$ratecub*dd$dummy

# use bandwidth 18 as written in the paper
dd$bandwidth <- ifelse(dd$povrate60>=41.1984 & dd$povrate60<=77.1984,1,0)

# linear fit
lin <- lm(rate~dummy+ratedum, data=subset(dd, bandwidth==1))
# quadratic fit
quad <- lm(rate~ratesq+ratesqdum+dummy+ratedum, data=subset(dd,bandwidth==1))
# qubic fit
cub <- lm(rate~ratecub+ratecubdum+ratesqdum+ratesq+dummy+ratedum,
          data=subset(dd,bandwidth==1))

# create bins from 40% to 80%
```

```r
dd$bins <- floor(dd$rate/2)*2 + 1 + 59.1984
sub <- subset(dd, bins>=40 & bins<= 80)
sub <- subset(sub, povrate60>=40 & povrate60 <= 80)

sub <-sub%>%
    group_by(bins)%>%
    mutate(mean=mean(bins), std=sd(bins))

# bandwidth 16 and 8
sub$bandwidth16 <- ifelse(sub$povrate60>=43.1984 & sub$povrate60<=75.1984,1,0)
sub$bandwidth8 <- ifelse(sub$povrate60>=51.1984 & sub$povrate60<=67.1984,1,0)

# bandwidth 12 and 19
sub$bandwidth12 <- ifelse(sub$povrate60>=47.1984 & sub$povrate60<=71.1984,1,0)
sub$bandwidth19 <- ifelse(sub$povrate60>=40.1984 & sub$povrate60<=79.1984,1,0)

# create table for bandwidth 8
stargazer(lm(hsspend_per_kid_68~dummy+rate+ratedum,
             data=subset(sub,bandwidth8==1)),
          lm(hsspend_per_kid_72~dummy+rate+ratedum,
             data=subset(sub,bandwidth8==1)),
          lm(socspend_per_cap72~dummy+rate+ratedum,
             data=subset(sub,bandwidth8==1)),
          keep = "dummy",report="c*sp", p.auto = T, header=F,
          omit.stat = c("ser","ll","rsq","adj.rsq","f"),
          covariate.labels = "Assistance",
          title= "Bandwidth 8")
```

Table 1: Bandwidth 8

| | *Dependent variable:* | | |
|---|---|---|---|
| | hsspend_per_kid_68 | hsspend_per_kid_72 | socspend_per_cap72 |
| | (1) | (2) | (3) |
| | 130.472 | 179.897 | 5.842 |
| | (120.893) | (143.319) | (22.307) |
| | p = 0.282 | p = 0.211 | p = 0.794 |
| Observations | 482 | 482 | 482 |
| *Note:* | | | *p<0.1; **p<0.05; ***p<0.01 |

```r
# create table for bandwidth 16
stargazer(lm(hsspend_per_kid_68~dummy+rate+ratedum+ratesqdum+ratesq,
             data=subset(sub,bandwidth16==1)),
          lm(hsspend_per_kid_72~dummy+rate+ratedum+ratesqdum+ratesq,
             data=subset(sub,bandwidth16==1)),
          lm(socspend_per_cap72~dummy+rate+ratedum+ratesqdum+ratesq,
             data=subset(sub,bandwidth16==1)),
          keep = "dummy",report="c*sp", p.auto = T, header=F,
          omit.stat = c("ser","ll","rsq","adj.rsq","f"),
          covariate.labels = "Assistance",
```

```
        title= "Bandwidth 16")
```

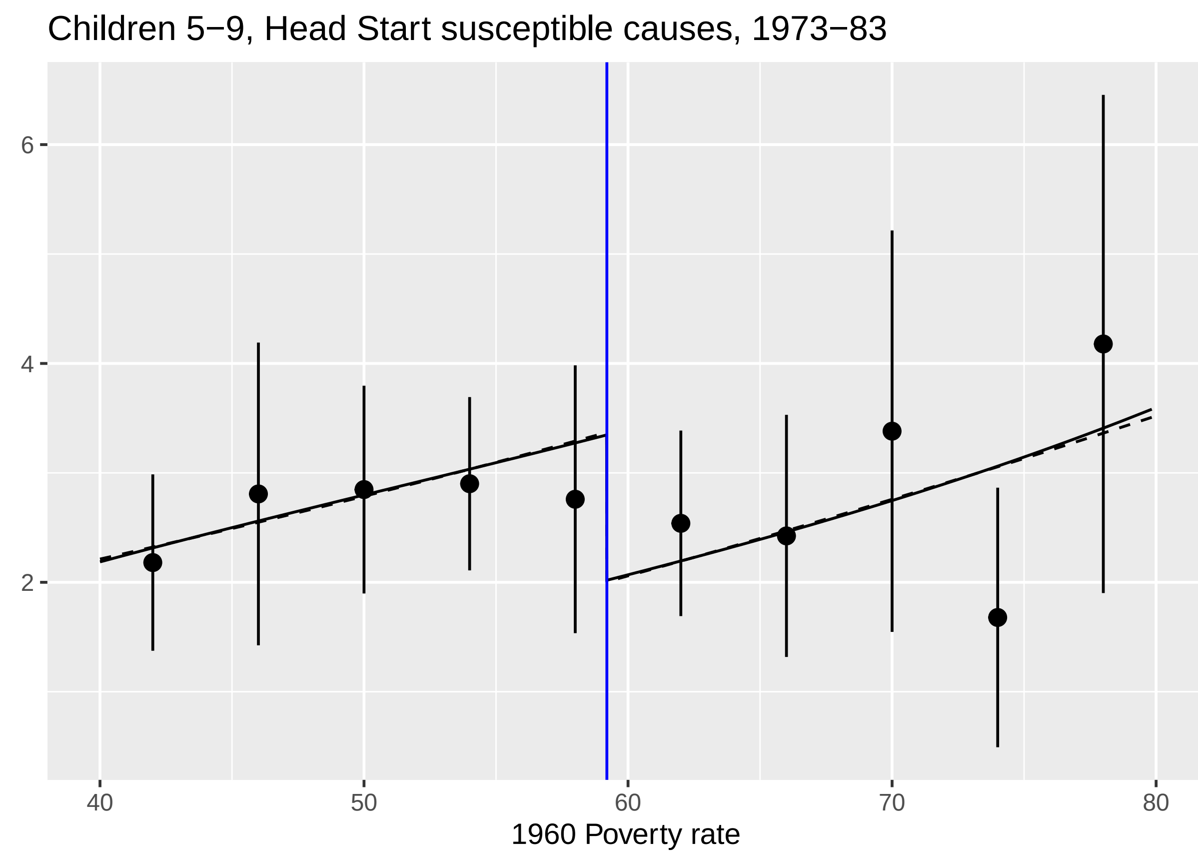
Table 2: Bandwidth 16

| | *Dependent variable:* | | |
|---|---|---|---|
| | hsspend_per_kid_68 | hsspend_per_kid_72 | socspend_per_cap72 |
| | (1) | (2) | (3) |
| | 117.881<br>(113.625)<br>p = 0.300 | 162.388<br>(133.501)<br>p = 0.225 | 11.244<br>(24.425)<br>p = 0.646 |
| Observations | 858 | 858 | 858 |
| *Note:* | | | *p<0.1; **p<0.05; ***p<0.01 |

**3(d)**

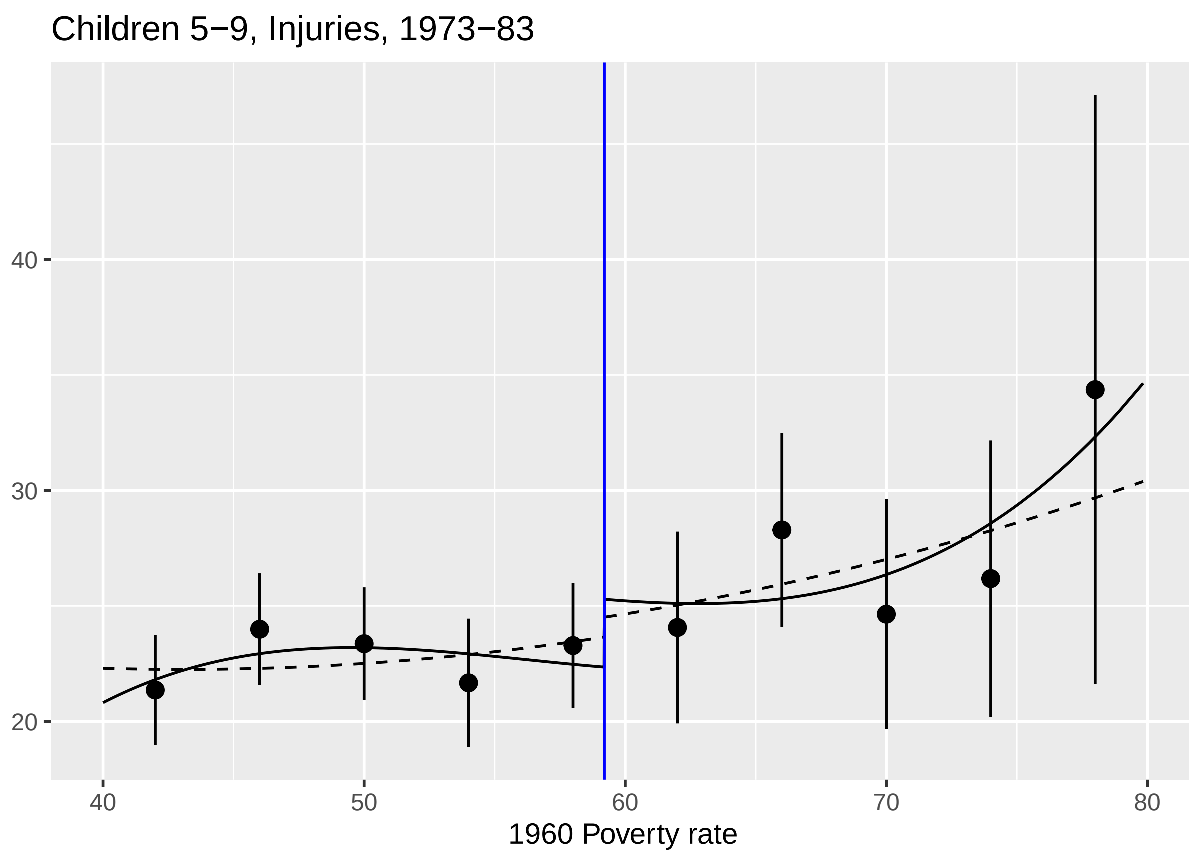
```
# calculate mean, standard deviation, and number of entries inside a certain bin
get_mean_sd_n <- function(dd.sub, bin_nr){
  mean_ <- mean(dd.sub[dd.sub$bin==bin_nr,]$age5_9_sum2)
  sd_ <- sd(dd.sub[dd.sub$bin==bin_nr,]$age5_9_sum2)
  n_ <- nrow(dd.sub[dd.sub$bin==bin_nr,])
  return(c(mean_,sd_,n_))
}

# create special data frame for calculating the CIs
ci_data <- data.frame(bin=numeric(), mean=numeric(), sd=numeric(), n=numeric())
for (i in seq(1,10)) {
  ci_data <- rbind(ci_data, c(i, get_mean_sd_n(dd.sub, i)))
}
colnames(ci_data) <- c('bin','mean','sd','n')

# define upper and lower bounds of CIs
ci_data$ci_lower <- 0
ci_data$ci_upper <- 0

# function for calculating the CI
calc_ci <- function(mean, sd, n, z){
  lower <- mean-(z*sd/sqrt(n))
  upper <- mean+(z*sd/sqrt(n))
  return(c(lower, upper))
}

# define value for 95% CI
z_95_percent <- 1.96

# calculate the CI of each bin
for (i in seq(1,10)) {
  ci <- calc_ci(ci_data[i,]$mean, ci_data[i,]$sd, ci_data[i,]$n, z_95_percent)
```

```r
  ci_data[i,]$ci_lower <- ci[1]
  ci_data[i,]$ci_upper <- ci[2]
}

# add central poverty of each bin
ci_data$poverty <- 0
for (i in seq(1,10)) {
  ci_data[i,]$poverty <- 38+i*4
}

# use cubic as "non-parametric" and quadratic as parametric
cubic <- lm(age5_9_sum2 ~ poly(povrate60, 3) + treatment, data=dd.sub)
quad <- lm(age5_9_sum2 ~ poly(povrate60, 2) + treatment, data=dd.sub)

################################################################################
# Panel A
################################################################################

# put everything in a plot
ggplot() + geom_line(aes(x=dd.sub$povrate60, y=cubic$fitted.values)) +
  geom_line(aes(x=dd.sub$povrate60, y=quad$fitted.values), linetype="dashed") +
  geom_vline(xintercept=cutoff, col="blue") +
  geom_pointrange(aes(x=ci_data$poverty, y=ci_data$mean,
                      ymin=ci_data$ci_lower, ymax=ci_data$ci_upper)) +
  ggtitle("Children 5-9, Head Start susceptible causes, 1973-83") +
  xlab("1960 Poverty rate") +
  ylab("")
```




```
##############################################################################
# Panel B
##############################################################################

# calculate mean, standard deviation, and number of entries inside a certain bin
get_mean_sd_n <- function(dd.sub, bin_nr){
  mean_ <- mean(dd.sub[dd.sub$bin==bin_nr,]$age5_9_injury_rate)
  sd_ <- sd(dd.sub[dd.sub$bin==bin_nr,]$age5_9_injury_rate)
  n_ <- nrow(dd.sub[dd.sub$bin==bin_nr,])
  return(c(mean_,sd_,n_))
}

# create special data frame for calculating the CIs
ci_data <- data.frame(bin=numeric(), mean=numeric(), sd=numeric(), n=numeric())
for (i in seq(1,10)) {
  ci_data <- rbind(ci_data, c(i, get_mean_sd_n(dd.sub, i)))
}
colnames(ci_data) <- c('bin','mean','sd','n')

# define upper and lower bounds of CIs
ci_data$ci_lower <- 0
ci_data$ci_upper <- 0

# function for calculating the CI
calc_ci <- function(mean, sd, n, z){
  lower <- mean-(z*sd/sqrt(n))
```

```
  upper <- mean+(z*sd/sqrt(n))
  return(c(lower, upper))
}

# define value for 95% CI
z_95_percent <- 1.96

# calculate the CI of each bin
for (i in seq(1,10)) {
  ci <- calc_ci(ci_data[i,]$mean, ci_data[i,]$sd, ci_data[i,]$n, z_95_percent)
  ci_data[i,]$ci_lower <- ci[1]
  ci_data[i,]$ci_upper <- ci[2]
}

# add central poverty of each bin
ci_data$poverty <- 0
for (i in seq(1,10)) {
  ci_data[i,]$poverty <- 38+i*4
}

# use cubic as "non-parametric" and quadratic as parametric
cubic <- lm(age5_9_injury_rate ~ poly(povrate60, 3) + treatment, data=dd.sub)
quad <- lm(age5_9_injury_rate ~ poly(povrate60, 2) + treatment, data=dd.sub)

# put everything in a plot
ggplot() + geom_line(aes(x=dd.sub$povrate60, y=cubic$fitted.values)) +
  geom_line(aes(x=dd.sub$povrate60, y=quad$fitted.values), linetype="dashed") +
  geom_vline(xintercept=cutoff, col="blue") +
  geom_pointrange(aes(x=ci_data$poverty, y=ci_data$mean,
                      ymin=ci_data$ci_lower, ymax=ci_data$ci_upper)) +
  ggtitle("Children 5-9, Injuries, 1973-83") +
  xlab("1960 Poverty rate") +
  ylab("")
```

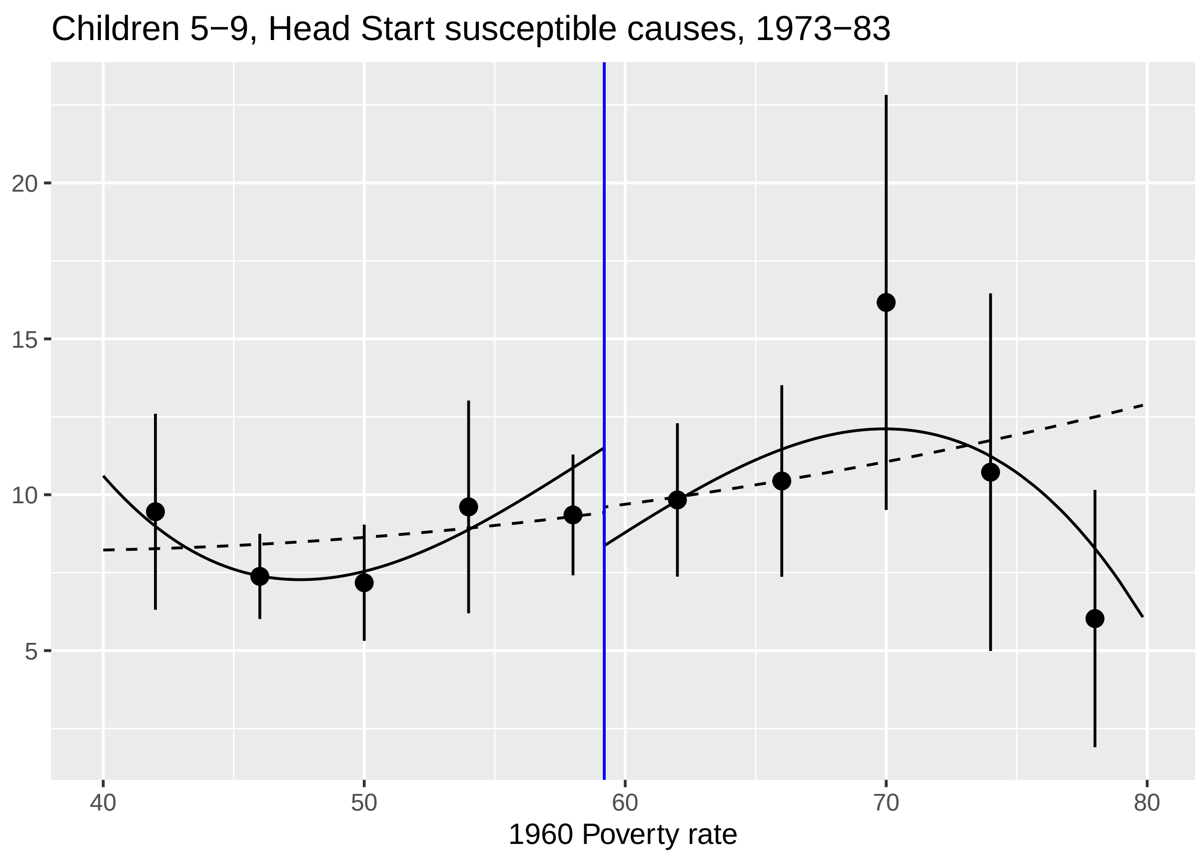



```
################################################################################
# Panel C
################################################################################

# calculate mean, standard deviation, and number of entries inside a certain bin
get_mean_sd_n <- function(dd.sub, bin_nr){
  mean_ <- mean(dd.sub[dd.sub$bin==bin_nr,]$age25plus_sum2)
  sd_ <- sd(dd.sub[dd.sub$bin==bin_nr,]$age25plus_sum2)
  n_ <- nrow(dd.sub[dd.sub$bin==bin_nr,])
  return(c(mean_,sd_,n_))
}

# create special data frame for calculating the CIs
ci_data <- data.frame(bin=numeric(), mean=numeric(), sd=numeric(), n=numeric())
for (i in seq(1,10)) {
  ci_data <- rbind(ci_data, c(i, get_mean_sd_n(dd.sub, i)))
}
colnames(ci_data) <- c('bin','mean','sd','n')

# define upper and lower bounds of CIs
ci_data$ci_lower <- 0
ci_data$ci_upper <- 0

# function for calculating the CI
calc_ci <- function(mean, sd, n, z){
  lower <- mean-(z*sd/sqrt(n))
```

```r
  upper <- mean+(z*sd/sqrt(n))
  return(c(lower, upper))
}

# define value for 95% CI
z_95_percent <- 1.96

# calculate the CI of each bin
for (i in seq(1,10)) {
  ci <- calc_ci(ci_data[i,]$mean, ci_data[i,]$sd, ci_data[i,]$n, z_95_percent)
  ci_data[i,]$ci_lower <- ci[1]
  ci_data[i,]$ci_upper <- ci[2]
}

# add central poverty of each bin
ci_data$poverty <- 0
for (i in seq(1,10)) {
  ci_data[i,]$poverty <- 38+i*4
}

# use cubic as "non-parametric" and quadratic as parametric
cubic <- lm(age25plus_sum2 ~ poly(povrate60, 3) + treatment, data=dd.sub)
quad <- lm(age25plus_sum2 ~ poly(povrate60, 2) + treatment, data=dd.sub)

# put everything in a plot
ggplot() + geom_line(aes(x=dd.sub$povrate60, y=cubic$fitted.values)) +
  geom_line(aes(x=dd.sub$povrate60, y=quad$fitted.values), linetype="dashed") +
  geom_vline(xintercept=cutoff, col="blue") +
  geom_pointrange(aes(x=ci_data$poverty, y=ci_data$mean,
                      ymin=ci_data$ci_lower, ymax=ci_data$ci_upper)) +
  ggtitle("Adults 25+, Head Start susceptible causes, 1973-83") +
  xlab("1960 Poverty rate") +
  ylab("")
```


```
################################################################################
# Panel D
################################################################################

# calculate mean, standard deviation, and number of entries inside a certain bin
get_mean_sd_n <- function(dd.sub, bin_nr){
  mean_ <- mean(dd.sub[dd.sub$bin==bin_nr,]$rate_5964)
  sd_ <- sd(dd.sub[dd.sub$bin==bin_nr,]$rate_5964)
  n_ <- nrow(dd.sub[dd.sub$bin==bin_nr,])
  return(c(mean_,sd_,n_))
}

# create special data frame for calculating the CIs
ci_data <- data.frame(bin=numeric(), mean=numeric(), sd=numeric(), n=numeric())
for (i in seq(1,10)) {
  ci_data <- rbind(ci_data, c(i, get_mean_sd_n(dd.sub, i)))
}
colnames(ci_data) <- c('bin','mean','sd','n')

# define upper and lower bounds of CIs
ci_data$ci_lower <- 0
ci_data$ci_upper <- 0

# function for calculating the CI
calc_ci <- function(mean, sd, n, z){
  lower <- mean-(z*sd/sqrt(n))
```

```r
  upper <- mean+(z*sd/sqrt(n))
  return(c(lower, upper))
}

# define value for 95% CI
z_95_percent <- 1.96

# calculate the CI of each bin
for (i in seq(1,10)) {
  ci <- calc_ci(ci_data[i,]$mean, ci_data[i,]$sd, ci_data[i,]$n, z_95_percent)
  ci_data[i,]$ci_lower <- ci[1]
  ci_data[i,]$ci_upper <- ci[2]
}

# add central poverty of each bin
ci_data$poverty <- 0
for (i in seq(1,10)) {
  ci_data[i,]$poverty <- 38+i*4
}

# use cubic as "non-parametric" and quadratic as parametric
cubic <- lm(rate_5964 ~ poly(povrate60, 3) + treatment, data=dd.sub)
quad <- lm(rate_5964 ~ poly(povrate60, 2) + treatment, data=dd.sub)

# put everything in a plot
ggplot() + geom_line(aes(x=dd.sub$povrate60, y=cubic$fitted.values)) +
  geom_line(aes(x=dd.sub$povrate60, y=quad$fitted.values), linetype="dashed") +
  geom_vline(xintercept=cutoff, col="blue") +
  geom_pointrange(aes(x=ci_data$poverty, y=ci_data$mean,
                      ymin=ci_data$ci_lower, ymax=ci_data$ci_upper)) +
  ggtitle("Children 5-9, Head Start susceptible causes, 1973-83") +
  xlab("1960 Poverty rate") +
  ylab("")
```




**3(e)**

```
# bandwidth 16 and 8
sub$bandwidth16 <- ifelse(sub$povrate60>=43.1984 & sub$povrate60<=75.1984,1,0)
sub$bandwidth8 <- ifelse(sub$povrate60>=51.1984 & sub$povrate60<=67.1984,1,0)

# bandwidth 12 and 19
sub$bandwidth12 <- ifelse(sub$povrate60>=47.1984 & sub$povrate60<=71.1984,1,0)
sub$bandwidth19 <- ifelse(sub$povrate60>=40.1984 & sub$povrate60<=79.1984,1,0)

# create table for bandwidth 8
stargazer(lm(age5_9_sum2~dummy+rate+ratedum,
             data=subset(sub,bandwidth8==1)),
          lm(age5_9_injury_rate~dummy+rate+ratedum,
             data=subset(sub,bandwidth8==1)),
          lm(age25plus_sum2~dummy+rate+ratedum,
             data=subset(sub,bandwidth8==1)),
            lm(rate_5964~dummy+rate+ratedum,
             data=subset(sub,bandwidth8==1)),
          keep = "dummy",report="c*sp", p.auto = T, header=F,
          omit.stat = c("ser","ll","rsq","adj.rsq","f"),
          covariate.labels = "Assistance",
          title= "Bandwidth 8")
```

Table 3: Bandwidth 8

| | *Dependent variable:* | | | |
|---|---|---|---|---|
| | age5_9_sum2 | age5_9_injury_rate | age25plus_sum2 | rate_5964 |
| | (1) | (2) | (3) | (4) |
| | $-2.201^{**}$ | $-0.164$ | 2.091 | $-3.682$ |
| | (1.004) | (3.380) | (5.581) | (2.886) |
| | p = 0.029 | p = 0.962 | p = 0.709 | p = 0.203 |
| Observations | 482 | 482 | 482 | 482 |
| *Note:* | | | | $^{*}$p<0.1; $^{**}$p<0.05; $^{***}$p<0.01 |

```
# create table for bandwidth 16
stargazer(lm(age5_9_sum2~dummy+rate+ratedum+ratesqdum+ratesq,
             data=subset(sub,bandwidth16==1)),
          lm(age5_9_injury_rate~dummy+rate+ratedum+ratesqdum+ratesq,
             data=subset(sub,bandwidth16==1)),
          lm(age25plus_sum2~dummy+rate+ratedum+ratesqdum+ratesq,
             data=subset(sub,bandwidth16==1)),
            lm(rate_5964~dummy+rate+ratedum+ratesqdum+ratesq,
             data=subset(sub,bandwidth16==1)),
          keep = "dummy",report="c*sp", p.auto = T, header=F,
          omit.stat = c("ser","ll","rsq","adj.rsq","f"),
          covariate.labels = "Assistance",
          title= "Bandwidth 16")
```

Table 4: Bandwidth 16

| | *Dependent variable:* | | | |
|---|---|---|---|---|
| | age5_9_sum2 | age5_9_injury_rate | age25plus_sum2 | rate_5964 |
| | (1) | (2) | (3) | (4) |
| | $-2.558^{**}$ | 0.775 | 2.574 | $-4.990^{*}$ |
| | (1.261) | (3.401) | (6.415) | (3.030) |
| | p = 0.043 | p = 0.820 | p = 0.689 | p = 0.100 |
| Observations | 858 | 858 | 858 | 858 |
| *Note:* | | | | $^{*}$p<0.1; $^{**}$p<0.05; $^{***}$p<0.01 |

**3(f)**

```
# bandwidth 16 and 8
sub$bandwidth16 <- ifelse(sub$povrate60>=43.1984 & sub$povrate60<=75.1984,1,0)
sub$bandwidth8 <- ifelse(sub$povrate60>=51.1984 & sub$povrate60<=67.1984,1,0)

# bandwidth 12 and 19
```

```r
sub$bandwidth12 <- ifelse(sub$povrate60>=47.1984 & sub$povrate60<=71.1984,1,0)
sub$bandwidth19 <- ifelse(sub$povrate60>=40.1984 & sub$povrate60<=79.1984,1,0)

# TODO: adapt for mortality outcomes

# create table for bandwidth 12
stargazer(lm(age5_9_injury_rate~dummy+rate+ratedum,
             data=subset(sub,bandwidth12==1)),
          lm(age5_9_sum2~dummy+rate+ratedum,
             data=subset(sub,bandwidth12==1)),
          lm(age25plus_sum2~dummy+rate+ratedum,
             data=subset(sub,bandwidth12==1)),
          lm(rate_5964~dummy+rate+ratedum,
             data=subset(sub,bandwidth12==1)),
          keep = "dummy",report="c*sp", p.auto = T, header=F,
          omit.stat = c("ser","ll","rsq","adj.rsq","f"),
          covariate.labels = "Assistance",
          title= "Linear with Bandwidth 12")
```

Table 5: Linear with Bandwidth 12

| | *Dependent variable:* | | | |
|---|---|---|---|---|
| | age5_9_injury_rate | age5_9_sum2 | age25plus_sum2 | rate_5964 |
| | (1) | (2) | (3) | (4) |
| | 1.194 | $-1.830^{**}$ | 4.237 | $-3.516$ |
| | (2.707) | (0.839) | (4.950) | (2.347) |
| | p = 0.660 | p = 0.030 | p = 0.393 | p = 0.135 |
| Observations | 645 | 645 | 645 | 645 |
| *Note:* | | | | $^{*}$p<0.1; $^{**}$p<0.05; $^{***}$p<0.01 |

```r
# create table for bandwidth 19
stargazer(lm(age5_9_injury_rate~dummy+rate+ratedum,
             data=subset(sub,bandwidth19==1)),
          lm(age5_9_sum2~dummy+rate+ratedum,
             data=subset(sub,bandwidth19==1)),
          lm(age25plus_sum2~dummy+rate+ratedum,
             data=subset(sub,bandwidth19==1)),
          lm(rate_5964~dummy+rate+ratedum,
             data=subset(sub,bandwidth19==1)),
          keep = "dummy",report="c*sp", p.auto = T, header=F,
          omit.stat = c("ser","ll","rsq","adj.rsq","f"),
          covariate.labels = "Assistance",
          title= "Linear with Bandwidth 19")
```

```r
# create table for bandwidth 12
stargazer(lm(age5_9_injury_rate~dummy+poly(rate,2)+poly(ratedum,2),
             data=subset(sub,bandwidth12==1)),
          lm(age5_9_sum2~dummy+poly(rate,2)+poly(ratedum,2),
```

Table 6: Linear with Bandwidth 19

| | *Dependent variable:* | | | |
|---|---|---|---|---|
| | age5_9_injury_rate | age5_9_sum2 | age25plus_sum2 | rate_5964 |
| | (1) | (2) | (3) | (4) |
| | 1.326 | −1.306* | 5.797 | −0.060 |
| | (2.132) | (0.771) | (4.082) | (1.949) |
| | p = 0.535 | p = 0.091 | p = 0.156 | p = 0.976 |
| Observations | 1,013 | 1,013 | 1,013 | 1,013 |
| *Note:* | | | | *p<0.1; **p<0.05; ***p<0.01 |

```
          data=subset(sub,bandwidth12==1)),
        lm(age25plus_sum2~dummy+poly(rate,2)+poly(ratedum,2),
          data=subset(sub,bandwidth12==1)),
        lm(rate_5964~dummy+poly(rate,2)+poly(ratedum,2),
          data=subset(sub,bandwidth12==1)),
        keep = "dummy",report="c*sp", p.auto = T, header=F,
        omit.stat = c("ser","ll","rsq","adj.rsq","f"),
        covariate.labels = "Assistance",
        title= "Quadratic with Bandwidth 12")
```

Table 7: Quadratic with Bandwidth 12

| | *Dependent variable:* | | | |
|---|---|---|---|---|
| | age5_9_injury_rate | age5_9_sum2 | age25plus_sum2 | rate_5964 |
| | (1) | (2) | (3) | (4) |
| | −0.360 | −2.161* | 1.531 | −4.892 |
| | (3.950) | (1.225) | (7.225) | (3.420) |
| | p = 0.928 | p = 0.079 | p = 0.833 | p = 0.154 |
| Observations | 645 | 645 | 645 | 645 |
| *Note:* | | | | *p<0.1; **p<0.05; ***p<0.01 |

```
# create table for bandwidth 19
stargazer(lm(age5_9_injury_rate~dummy+poly(rate,2)+poly(ratedum,2),
          data=subset(sub,bandwidth19==1)),
        lm(age5_9_sum2~dummy+poly(rate,2)+poly(ratedum,2),
          data=subset(sub,bandwidth19==1)),
        lm(age25plus_sum2~dummy+poly(rate,2)+poly(ratedum,2),
          data=subset(sub,bandwidth19==1)),
        lm(rate_5964~dummy+poly(rate,2)+poly(ratedum,2),
          data=subset(sub,bandwidth19==1)),
        keep = "dummy",report="c*sp", p.auto = T, header=F,
        omit.stat = c("ser","ll","rsq","adj.rsq","f"),
        covariate.labels = "Assistance",
        title= "Quadratic with Bandwidth 19")
```

Table 8: Quadratic with Bandwidth 19

| | *Dependent variable:* | | | |
|---|---|---|---|---|
| | age5_9_injury_rate | age5_9_sum2 | age25plus_sum2 | rate_5964 |
| | (1) | (2) | (3) | (4) |
| | 2.106 | $-1.823$ | 2.194 | $-4.955^{*}$ |
| | (3.159) | (1.142) | (6.047) | (2.879) |
| | p = 0.506 | p = 0.111 | p = 0.717 | p = 0.086 |
| Observations | 1,013 | 1,013 | 1,013 | 1,013 |
| *Note:* | | | | $^{*}$p<0.1; $^{**}$p<0.05; $^{***}$p<0.01 |