PS 200D: Causal Inference
Problem Set 3

Chao-yo Cheng

June 3, 2015

Problem 1

Given that $D = 1$ in New Jersey and $T = 1$ in November. The average wage matrix in Table 1 can be rewritten as follows:

$$
\begin{array}{ccc}
 & T=0 & T=1 \\
D=1 & \alpha & \beta \\
D=0 & \gamma & \delta
\end{array}
$$

where

- $\mathbb{E}[Y_i|T = 0, D = 1] = \alpha$
- $\mathbb{E}[Y_i|T = 1, D = 1] = \beta$
- $\mathbb{E}[Y_i|T = 0, D = 0] = \gamma$
- $\mathbb{E}[Y_i|T = 1, D = 0] = \delta$

Part (1)

Part (i)

The difference-in-differences (DID) effect, denoted by τ_{DID}, can be defined as the following estimand.

$$
\begin{aligned}
\tau_{DID} &= \underbrace{\{\mathbb{E}[Y_i|T = 1, D = 1] - \mathbb{E}[Y_i|T = 0, D = 1]\}}_{\text{Effect of time and treatment}} - \underbrace{\{\mathbb{E}[Y_i|T = 1, D = 0] - \mathbb{E}[Y_i|T = 0, D = 0]\}}_{\text{Effect of time}} \\
&= (\beta - \alpha) - (\delta - \gamma)
\end{aligned} \tag{1}
$$

Part (ii)

Considering all potential outcomes for α, β, γ, and δ, there are only four potential outcomes are observed, as suggested in the average wage matrix:

- $\alpha(0) = \mathbb{E}[Y_0|T = 0, D = 1]$.
- $\beta(1) = \mathbb{E}[Y_1|T = 1, D = 1]$.
- $\gamma(0) = \mathbb{E}[Y_0|T = 0, D = 0]$.
- $\delta(0) = \mathbb{E}[Y_0|T = 1, D = 0]$.

The ATT of interest is defined as

$$\tau_{ATT} = \mathbb{E}[Y_1|T=1,D=1] - \mathbb{E}[Y_0|T=1,D=1] = \beta(1) - \beta(0) \tag{2}$$

Part (iii)

To identify the ATT with the DID estimand, we need to assume "parallel trends." That is, in the absence of treatment, the treated and control units should exhibit the same time trend. As a result, the difference in the non-treated potential outcomes for the treated group before and after T should be equal to that for the control group.

$$\mathbb{E}[Y_0|T=1,D=1] - \mathbb{E}[Y_0|T=0,D=1] = \mathbb{E}[Y_0|T=1,D=0] - \mathbb{E}[Y_0|T=0,D=0] \tag{3}$$

That is, $\beta(0) - \alpha(0) = \delta(0) - \gamma(0)$, which follows that $\beta(0) = \alpha(0) + \delta(0) - \gamma(0)$. The DID estimand then identify τ_{ATT} as follows.

$$\begin{aligned}\tau_{ATT} &= \mathbb{E}[Y_1|T=1,D=1] - \mathbb{E}[Y_0|T=1,D=1] \\ &= \beta(1) - \beta(0) \\ &= \beta(1) - (\alpha(0) + \delta(0) - \gamma(0))\end{aligned} \tag{4}$$

all of which are observed.

Part (b)

See the following R output.

```
> treat = dta[dta$state %in% "nj",] # D=1
> pre.treat = treat$emp_pre # D=1 and T=0
> post.treat = treat$emp_post # D=1 and T=1
>
> control = dta[dta$state %in% "pa",] # D=0
> pre.control = control$emp_pre # D=0 and T=0
> post.control = control$emp_post # D=0 and T=1
>
> ## Table 1
> alpha = mean(pre.treat, na.rm=T)
> beta = mean(post.treat, na.rm=T)
> gamma = mean(pre.control, na.rm=T)
> delta = mean(post.control, na.rm=T)
>
> res = matrix(c(alpha, gamma, beta, delta), 2, 2)
> rownames(res) = c("NJ", "PA")
> colnames(res) = c("Feb", "Nov")
> round(res, digits=3)
      Feb    Nov
NJ 20.439 21.027
PA 23.331 21.166
```

For the variance-covariance matrix, we can assume that the sampling of stores in Pennsylvania and New Jersey was independent. As a result, we chave $cov(\alpha,\delta) = cov(\alpha,\gamma) = cov(\beta,\gamma) = cov(\beta,\delta) = 0$. See the following R output.

```
> ## Table 2
> alpha.var = var(pre.treat, na.rm=T)/nrow(treat)
> beta.var = var(post.treat, na.rm=T)/nrow(treat)
```

```
> gamma.var = var(pre.control, na.rm=T)/nrow(control)
> delta.var = var(post.control, na.rm=T)/nrow(control)
> ab = cov(dta$emp_pre[dta$state %in% "nj"], dta$emp_post[dta$state %in% "nj"], use="
    pairwise.complete.obs")/nrow(treat)
> gd = cov(dta$emp_pre[dta$state %in% "pa"], dta$emp_post[dta$state %in% "pa"], use="
    pairwise.complete.obs")/nrow(control)
> a.row = c(alpha.var, ab, 0, 0)
> b.row = c(ab, beta.var, 0, 0)
> g.row = c(0, 0, gamma.var, gd)
> d.row = c(0, 0, gd, delta.var)
> vcov = rbind(a.row, b.row, g.row, d.row)
> rownames(vcov) = c("alpha", "beta", "gamma", "delta")
> colnames(vcov) = c("alpha", "beta", "gamma", "delta")
> round(vcov, digits=3)
      alpha  beta gamma delta
alpha 0.251 0.153 0.000 0.000
beta  0.153 0.261 0.000 0.000
gamma 0.000 0.000 1.779 0.611
delta 0.000 0.000 0.611 0.867
```

Part (c)

See the following R output.

```
> alpha = mean(pre.treat, na.rm=T)
> beta = mean(post.treat, na.rm=T)
> gamma = mean(pre.control, na.rm=T)
> delta = mean(post.control, na.rm=T)
> did = beta - alpha - delta + gamma; did
[1] 2.753606
```

The variance of τ_{ATT} is as follows.

$$\begin{aligned}\mathbb{V}[\tau_{ATT}] &= \mathbb{V}[\beta(1) - \alpha(0) - \delta(0) + \gamma(0)] \\ &= \mathbb{V}[\widehat{\beta}] + \mathbb{V}[\widehat{\alpha}] + \mathbb{V}[\widehat{\delta}] + \mathbb{V}[\widehat{\gamma}] - 2cov(\widehat{\alpha}, \widehat{\beta}) - 2cov(\widehat{\gamma}, \widehat{\delta})\end{aligned} \tag{5}$$

Therefore,

```
> alpha.var = var(pre.treat, na.rm=T)/nrow(treat)
> beta.var = var(post.treat, na.rm=T)/nrow(treat)
> gamma.var = var(pre.control, na.rm=T)/nrow(control)
> delta.var = var(post.control, na.rm=T)/nrow(control)
> ab = cov(dta$emp_pre[dta$state %in% "nj"], dta$emp_post[dta$state %in% "nj"], use="
    pairwise.complete.obs")/nrow(treat)
> gd = cov(dta$emp_pre[dta$state %in% "pa"], dta$emp_post[dta$state %in% "pa"], use="
    pairwise.complete.obs")/nrow(control)
> var = alpha.var + beta.var + gamma.var + delta.var - 2*ab - 2*gd; var
[1] 1.628625
```

We can then calculate a two-sided p-value with a normal distribution.

```
> z = did/sqrt(var)
> 2*pnorm(-abs(z))
[1] 0.0309511
```

Based on the p-value, the impact is statistically significant and positive with $p < 0.05$.

Problem 2

Part (a)

Given relevance and monotonicity, the first stage can be formally defined as follows.

$$D = \tau + \rho Z + \eta \tag{6}$$

To identify the first-stage effect, we need the following assumption.

- Independence (or ignorability), namely the half of the ignorability assumption $\{D_{1i}, D_{0i}\} \perp\!\!\!\perp Z$.

Part (b)

Together with the first stage defined in Part (a), we can formally define the second stage as follows.

$$Y = \gamma + \alpha D + \epsilon \tag{7}$$

The intent-to-treat (ITT) estimate can then be defined as

$$\begin{aligned} Y &= \gamma + \alpha D + \epsilon \\ &= \gamma + \alpha(\tau + \rho Z + \eta) + \epsilon \\ &= \gamma + \alpha\tau + \alpha\rho Z + \alpha\eta + \epsilon \\ &= (\gamma + \alpha\tau) + \alpha\rho Z + (\alpha\eta + \epsilon) \end{aligned} \tag{8}$$

In addition to the assumptions for the first-stage effect (i.e. ρ), we need the following assumption to identify the ITT effect.

- Exclusion, as implied by independence, such that $\{Y_{1i}, Y_{0i}\} \perp\!\!\!\perp Z$. In other words, Z can only affect Y through D. This can also be formally written as $Cov(Z, \epsilon) = 0$. If this is not the case, then we know ϵ contains something that can be explained by Z, which suggests that Z can affect Y through channels other than D.

Part (c)

As explained above, the exclusion assumption requires that Z, the instrument variable, can only affect Y through D. This can be implied by the ignorability assumption because we need it to identify the causal effect of Z on D (i.e. the first-stage effect), from which we identify the effect of D on Y.

Part (d)

No. If the exclusion assumption is violated, then we know $cov(Z, \epsilon) \neq 0$, which indicates that Z can affect Y without affecting D beforehand. The estimation for the ITT effect will then become invalid as

$$\begin{aligned} Y &= \gamma + \alpha D + \epsilon \\ &= \gamma + \alpha D + \beta Z + \xi \\ &= \gamma + \alpha(\tau + \rho Z + \eta) + (\beta Z + \xi) \\ &= \gamma + \alpha\tau + \alpha\rho\beta Z + \alpha\eta + \xi \\ &= (\gamma + \alpha\tau) + \alpha\rho\beta Z + (\alpha\eta + \xi) \end{aligned} \tag{9}$$

Z's coefficient, namely the ITT estimate, will be biased as it contains β.

Part (e)

To define the ATE of Z on the outcome Y, the instrument (or the encouragement), we first need to know that there can be four possible scenarios after any unit takes Z: "Never-takers" (those who will never take treatment regardless of the instrument), "always-takers" (those who will always take treatment regardless of the instrument), "defiers" (those who will not take the treatment after being exposed to the instrument), and "compliers" (those who will take the treatment after being exposed to the instrument). As a result, the ATE of Z on Y, denoted as θ, is the following according to the law of iterated expectations.

$$\begin{aligned}\mathbb{E}[\theta] = &\ \mathbb{E}[\theta|\text{never-taker}] \times Pr(\text{never-taker}) \\ &+ \mathbb{E}[\theta|\text{always-taker}] \times Pr(\text{always-taker}) \\ &+ \mathbb{E}[\theta|\text{complier}] \times Pr(\text{complier}) \\ &+ \mathbb{E}[\theta|\text{defier}] \times Pr(\text{defier}) \\ = &\ \mathbb{E}[\theta|\text{complier}] \times Pr(\text{complier})\end{aligned} \tag{10}$$

because (1) we assume there are no defiers (hence $Pr(\text{complier}) = 0$) and (2) encouragement has no impact on always=takers and never-takers (hence $\mathbb{E}[\theta|\text{never-taker}] = \mathbb{E}[\theta|\text{always-taker}] = 0$). The ATE among the compliers, namely $\mathbb{E}[\theta|\text{complier}]$, can be identified by

$$\mathbb{E}[\theta|\text{complier}] = \frac{\mathbb{E}[\theta]}{Pr(\text{complier})} \tag{11}$$

Intuitively, the ATE among the compliers can be divided by the reduced form effect (i.e. the effect of Z on Y) and the first-stage effect (i.e. the effect of Z on D).

Part (f)

Based on the independence assumption, it follows that $Y_i(D = 1, Z = 1) = Y_i(D = 0, Z = 1)$. Similarly, $Y_i(D = 1, Z = 0) = Y_i(D = 0, Z = 0)$. Therefore, we do not need to write the potential outcomes with two arguments.

Part (g)

Even if Z_i is randomly assigned, the ignorability assumption can be violated if Z_i can affect Y_i without affecting D_i first because the instrument variable might affect the outcome of interest through variables other than the treatment. In other words, random assignment does not guarantee ignorability because it does not necessarily guarantee exclusion.

Problem 3

Part (a)

See the following table.

		(1) No Covariates	(2) Including Latitude
First stage (dep: avexpr)	logem4	-0.607^{***}	-0.510^{***}
		(0.153)	(0.169)
	lat_abst		2.002
			(1.419)
Reduced form (dep: logpgp95)	logem4	-0.573^{***}	-0.508^{***}
		(0.074)	(0.092)
	lat_abst		1.346^{*}
			(0.864)
2SLS (dep: logpgp95)	avexpr	0.944^{***}	0.996^{***}
		(0.179)	(0.245)
	lat_abst		-0.647
			(1.287)

Note: *p<0.1; **p<0.05; ***p<0.01

Part (b)

Instead of controlling for `lat_abst`, we can conduct the 2SLS estimation by "partially" out the effect of latitude. To do so, we fist regress `lat_abst` on `avexpr`, `logem4`, and `logpgp95`, and then use the residuals to re-do the estimation. The results virtually remain the same.

```
> library(lmtest)
> pa1 = lm(logpgp95 ~ lat_abst, data=dta)#; coeftest(pa1)
> pa2 = lm(avexpr ~ lat_abst, data=dta)#; coeftest(pa2)
> pa3 = lm(logem4 ~ lat_abst, data=dta)#; coeftest(pa3)
>
> dta$logpgp95.re = pa1$residuals
> dta$avexpr.re = pa2$residuals
> dta$logem4.re = pa3$residuals
>
> mod.ire = ivreg(logpgp95.re ~ avexpr.re | logem4.re, data=dta)
> mod.irer = coeftest(mod.ire, vcov = vcovHC(mod.ire, type = "HC2"))
> round(mod.irer, digits=3) # Results from partialling out the effect of latitude

t test of coefficients:

            Estimate Std. Error t value Pr(>|t|)
(Intercept)    0.000      0.124   0.000        1
avexpr.re      0.996      0.244   4.082   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

> round(mod.i2r, digits=3) # Results from controlling for the effect of latitude

t test of coefficients:

            Estimate Std. Error t value Pr(>|t|)
(Intercept)    1.692      1.472   1.150    0.255
avexpr         0.996      0.245   4.062   <2e-16 ***
lat_abst      -0.647      1.287  -0.503    0.617
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Part (c)

For settler's morality to be a "good" instrument variable, we need to check whether it meets the identification assumptions. To begin with, AJR indeed propose a rather compelling story to explain why high settler mortality led to lower protection against expropriation risks by establishing "extractive" colonies that later became independent countries, so the assumption of first-stage effect (i.e. relevance) seems plausible. Meanwhile, since the assumption of monotonicity is not directly testable, qualitative information becomes essential. Based on existing case studies, it indeed appears that colonies with high settler casualties tend to witness "extractive" infrastructure that led to lower property rights. Unlike relevance and monotonicity, the assumption of ignorability seems rather fragile. On the one hand, it requires that the instrument, namely settler mortality, is "randomly" assigned, but we can never be sure whether or not this is the case. More importantly, settler mortality does not necessarily affect economic development through the protection for property rights. Instead, it might affect economic development through the accumulation of human capital, thus violating the assumption of exclusion.

Problem 4

Part (a)

We cannot simply run a regression of reading scores on class sizes. The results will not be a valid causal estimate since class size is a randomly assigned treatment. There might be other confounding/omitted variables.

Part (b)

See the following figure. The results imply the presence of several potential thresholds of enrollment

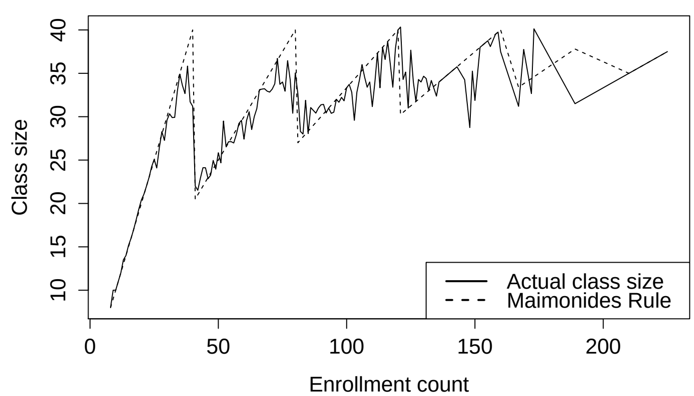


Figure 1: Class Size in 1991 by Initial Enrollment Count, Actual Average Size and as Predicted by Maimonides Rule (4th Grade)

where discontinuity on class size takes place.

Part (c)

The following `R` function creates the discontinuity sample with specified `threshold` (c) and `window` (i.e. bandwidth) and show the sample in a scatterplot.

```
rdd = function(threshold, window=5){
  dta.rdd = dta[dta$enrollment >= threshold - window,]
  dta.rdd = dta.rdd[dta.rdd$enrollment <= threshold + window,]
  dta.rdd$forcing = dta.rdd$enrollment - threshold
  dta.rdd$treat = ifelse(dta.rdd$forcing >0, 1, 0)
  dta.rdd$threshold = threshold
  plot(dta.rdd$forcing, dta.rdd$classize, type="p", xlab="Forcing Variable", ylab="Class
       Size", main=paste("Threshold=", threshold))
  abline(v=0, lty=2, col="red")
  return(dta.rdd = dta.rdd)
}
```

Given that we only include schools where enrollment is more than 5 students away from the threshold, we can use 41, 81, and 121 as the enrollment thresholds to create discontinuity sample. The "forcing variable" lies between -5 and 5 because it indicate the distance from the discontinuity point. The following figures show the discontinuity sample around the thresholds. I then conduct the RDD analysis while allowing

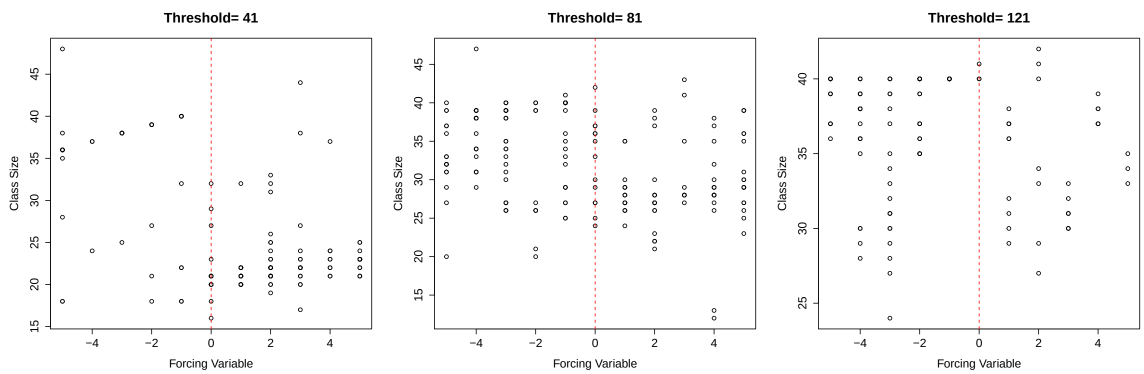
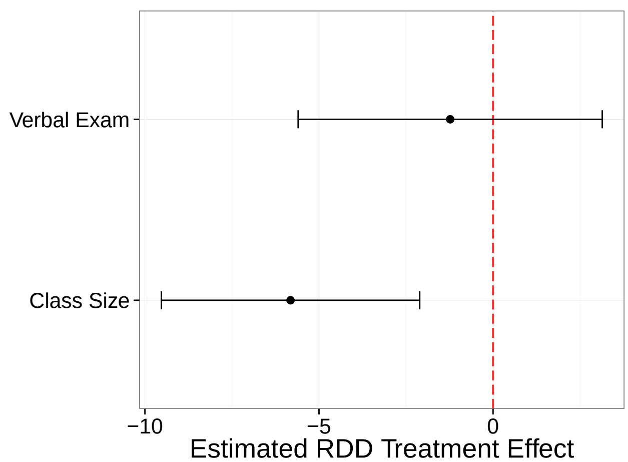


Figure 2: Discontinuity Sample at Different Thresholds

$\mathbb{E}[Y_0|X]$ and $\mathbb{E}[Y_1|X]$ to be different linear functions of X (in this case, X refers to enrollment). As a result, I estimate

$$\mathbb{E}[Y_i|X_i, D_i] = \gamma + \beta_0 X_i + \alpha D_i + \beta_1 X_i \times D_i \tag{12}$$

where D_i refers to the treatment (i.e. "forcing variable" greater than 1) and $\tilde{X}_i = X_i - c$ (i.e. the "forcing variable"). Y_i is the class size and the average reading comprehension score in a class. I use cluster standard errors because standard errors might be correlated within individual schools. The following figure plots the 95% confidence interval of the estimated effect. As illustrated, discontinuities significantly reduce class size while impose a relatively ambiguous effect on the average reading comprehension score in a class.


Verbal Exam
Class Size
−10
−5
0
Estimated RDD Treatment Effect

Part (d)

Until now, we have assumed that the discontinuity thresholds perfectly determine treatment exposure. However, this may not be the case. Instead, thresholds only create a discontinuity in the "probability" of treatment exposure. This is the idea of "fuzzy" RDD. In this study, the idea of fuzzy RDD suggests that enrollment only determines the propensity of having a certain class size but may not perfectly determine it. As suggested by the figure in Part (b), this seems highly plausible as the actual average class size indeed deviates from the class size dictated by the Maimonides Rule.

To conduct fuzzy RDD analysis, we can then treat enrollment as the instrument variable of class size (i.e. enrollment "encourages," rather than mandating, the school to adopt a particular class size based on the Maimonides Rule). In particular, around the discontinuity thresholds, we can presume that

- Around the enrollment threshold, the adoption of the actual class size is random.
- A tiny change in enrollment around the threshold, being nearly random, only affects the average reading comprehension score in a class through class size. Notice that in this case, the treatment is not binary.

Formally speaking, we need the following assumptions:

- Ignorability and exclusion: $\{Y_{0i}, Y_{1i}, D_{1i}, D_{0i}\} \perp\!\!\!\perp Z_i$. See above.
- Relevance: $\mathbb{E}[D_{1i} \neq \mathbb{E}[D_{0i}$; in other words, on average the instrument variable must affect the adoption of class class.
- Monotonicity: $D_{1i} \geq D_{0i}$ for all i.

Within these assumptions, the causal estimate is

$$\begin{aligned}
\alpha_{FRDD} &= \mathbb{E}[Y_{1i} - Y_{0i} | X_i = c, D_{1i} \geq D_{0i}] \\
&= \frac{\lim_{X \downarrow c} \mathbb{E}[Y_i | X_i = c] - \lim_{X \uparrow c} \mathbb{E}[Y_i | X_i = c]}{\lim_{X \downarrow c} \mathbb{E}[D_i | X_i = c] - \lim_{X \uparrow c} \mathbb{E}[D_i | X_i = c]} \\
&= \frac{\text{outcome discontinuity}}{\text{treatment discontinuity}} \\
&\approx \frac{\mathbb{E}[Y_i | Z_i = 1] - \mathbb{E}[Y_i | Z_i = 0]}{\mathbb{E}[D_i | Z_i = 1] - \mathbb{E}[D_i | Z_i = 0]}
\end{aligned} \tag{13}$$

In the context of regression, we can use the discontinuity sample to fit the following IV regression

$$Y_i = \gamma + \beta_0 \tilde{X}_i + \alpha D_i + \beta_1 D_i \times \tilde{X}_i \tag{14}$$

where $Z_i = 1$ if $X_i > c$. In practice, we simply use enrollment as the instrument variable of class size to estimate the effect of latter on the performance on reading comprehension exams.

While the assumptions of relevance and ignorability are plausible, I argue that the exclusion assumption is a bit hard to maintain. In particular, one may argue that enrollment might affect teacher's quality (or the faculty size), which then affects students' performance on the reading comprehension exams.

Appendix: `R` Code for Problem Set 3

```
#### 200D Problem Set 3
#### Chao-yo Cheng

rm(list=ls())
setwd("C:/Users/CYCheng/Desktop/Dropbox/2015 Spring/Coursework/200D/Problem Set/Pset 3")
setwd("/Users/chaoyocheng/Dropbox/2015 Spring/Coursework/200D/Problem Set/Pset 3")

##### Question 1 #####
rm(list=ls())
load("card_krueger.Rdata")
dta = d; rm(d)
#head(dta)

### Part (b)
treat = dta[dta$state %in% "nj",] # D=1
pre.treat = treat$emp_pre # D=1 and T=0
post.treat = treat$emp_post # D=1 and T=1

control = dta[dta$state %in% "pa",] # D=0
pre.control = control$emp_pre # D=0 and T=0
post.control = control$emp_post # D=0 and T=1

## Table 1
alpha = mean(pre.treat, na.rm=T)
beta = mean(post.treat, na.rm=T)
gamma = mean(pre.control, na.rm=T)
delta = mean(post.control, na.rm=T)

res = matrix(c(alpha, gamma, beta, delta), 2, 2)
rownames(res) = c("NJ", "PA")
colnames(res) = c("Feb", "Nov")
round(res, digits=3)

## Table 2
alpha.var = var(pre.treat, na.rm=T)/nrow(treat)
beta.var = var(post.treat, na.rm=T)/nrow(treat)
gamma.var = var(pre.control, na.rm=T)/nrow(control)
delta.var = var(post.control, na.rm=T)/nrow(control)
ab = cov(dta$emp_pre[dta$state %in% "nj"], dta$emp_post[dta$state %in% "nj"], use="
    pairwise.complete.obs")/nrow(treat)
gd = cov(dta$emp_pre[dta$state %in% "pa"], dta$emp_post[dta$state %in% "pa"], use="
    pairwise.complete.obs")/nrow(control)
a.row = c(alpha.var, ab, 0, 0)
b.row = c(ab, beta.var, 0, 0)
g.row = c(0, 0, gamma.var, gd)
d.row = c(0, 0, gd, delta.var)
vcov = rbind(a.row, b.row, g.row, d.row)
rownames(vcov) = c("alpha", "beta", "gamma", "delta")
colnames(vcov) = c("alpha", "beta", "gamma", "delta")
round(vcov, digits=3)

### Part (c)
alpha = mean(pre.treat, na.rm=T)
beta = mean(post.treat, na.rm=T)
gamma = mean(pre.control, na.rm=T)
delta = mean(post.control, na.rm=T)
```

```r
did = beta - alpha - delta + gamma; did

alpha.var = var(pre.treat, na.rm=T)/nrow(treat)
beta.var = var(post.treat, na.rm=T)/nrow(treat)
gamma.var = var(pre.control, na.rm=T)/nrow(control)
delta.var = var(post.control, na.rm=T)/nrow(control)
ab = cov(dta$emp_pre[dta$state %in% "nj"], dta$emp_post[dta$state %in% "nj"], use="
    pairwise.complete.obs")/nrow(treat)
gd = cov(dta$emp_pre[dta$state %in% "pa"], dta$emp_post[dta$state %in% "pa"], use="
    pairwise.complete.obs")/nrow(control)
var = alpha.var + beta.var + gamma.var + delta.var - 2*ab - 2*gd; var

z = did/sqrt(var)
2*pnorm(-abs(z))

did + 1.96*sqrt(var)
did - 1.96*sqrt(var)

##### Question 3 #####
rm(list=ls())

library(foreign)
dta = read.dta("arj.dta"); ls(dta)

### Part (a)
library(lmtest); library(AER)

## First-stage
mod.f1 = lm(avexpr ~ logem4, data=dta)
mod.f1r = coeftest(mod.f1, vcov = vcovHC(mod.f1, type = "HC2"))

mod.f2 = lm(avexpr ~ logem4 + lat_abst, data=dta)
mod.f2r = coeftest(mod.f2, vcov = vcovHC(mod.f2, type = "HC2"))

library(stargazer)
stargazer(list(mod.f1r, mod.f2r),
          column.labels= c("No Covariates", "Including Latitude"))

## Reduced form
mod.r1 = lm(logpgp95 ~ logem4, data=dta)
mod.r1r = coeftest(mod.r1, vcov = vcovHC(mod.r1, type = "HC2"))

mod.r2 = lm(logpgp95 ~ logem4 + lat_abst, data=dta)
mod.r2r = coeftest(mod.r2, vcov = vcovHC(mod.r2, type = "HC2"))

library(stargazer)
stargazer(list(mod.r1r, mod.r2r),
          column.labels= c("No Covariates", "Including Latitude"))

## 2SLS
mod.i1 = ivreg(logpgp95 ~ avexpr | logem4, data=dta)
mod.i1r = coeftest(mod.i1, vcov = vcovHC(mod.i1, type = "HC2"))

mod.i2 = ivreg(logpgp95 ~ avexpr + lat_abst | logem4 + lat_abst, data=dta)
mod.i2r = coeftest(mod.i2, vcov = vcovHC(mod.i2, type = "HC2"))

library(stargazer)
stargazer(list(mod.i1r, mod.i2r),
```

```
          column.labels= c("No Covariates", "Including Latitude"))

### Part (b)
library(lmtest)
pa1 = lm(logpgp95 ~ lat_abst, data=dta)#; coeftest(pa1)
pa2 = lm(avexpr ~ lat_abst, data=dta)#; coeftest(pa2)
pa3 = lm(logem4 ~ lat_abst, data=dta)#; coeftest(pa3)

dta$logpgp95.re = pa1$residuals
dta$avexpr.re = pa2$residuals
dta$logem4.re = pa3$residuals

mod.ire = ivreg(logpgp95.re ~ avexpr.re | logem4.re, data=dta)
mod.irer = coeftest(mod.ire, vcov = vcovHC(mod.ire, type = "HC2"))
round(mod.irer, digits=3) # Results from partialling out the effect of latitude
round(mod.i2r, digits=3) # Results from controlling for the effect of latitude

##### Question 4 #####
rm(list=ls())

library(foreign)
dta = read.dta("angrist_lavy.dta"); ls(dta)

### Part (a)
library(lmtest)
mod = lm(avgverb ~ classize + as.factor(schlcode), data=dta)
round(coeftest(mod)[1:2,], digits=3)

### Part (b)
library(base)
e = sort(unique(dta$enrollment))
f.mai =  e/ (floor((e-1)/40) + 1); length(f.mai)
f.avg = tapply(dta$classize, dta$enrollment, mean, na.rm=T); length(f.avg)

res = cbind(f.avg, f.mai)
plot(e, res[,1], type="l", xlab="Enrollment count", ylab="Class size")
lines(e, res[,2], lty=2)
legend("bottomright", legend=c("Actual class size","Maimonides Rule"), lty=c(1,2), lwd
    =2)

### Part (c)
rdd = function(threshold, window=5){
  dta.rdd = dta[dta$enrollment >= threshold - window,]
  dta.rdd = dta.rdd[dta.rdd$enrollment <= threshold + window,]
  dta.rdd$forcing = dta.rdd$enrollment - threshold
  dta.rdd$treat = ifelse(dta.rdd$forcing >0, 1, 0)
  dta.rdd$threshold = threshold
  plot(dta.rdd$forcing, dta.rdd$classize, type="p", xlab="Forcing Variable", ylab="Class
       Size", main=paste("Threshold=", threshold))
  abline(v=0, lty=2, col="red")
  return(dta.rdd = dta.rdd)
}
par(mfrow=c(1,3))
rdd.41 = rdd(41, 5)
rdd.81 = rdd(81, 5)
rdd.121 = rdd(121, 5)
#rdd.161 = rdd(161, 5)
#dta.rdd = rbind(rdd.41, rdd.81, rdd.121, rdd.161)
```

```
dta.rdd = rbind(rdd.41, rdd.81, rdd.121)
dev.off()

rdd = lm(classize ~ treat + forcing + treat * forcing, data = dta.rdd)
rdd2 = lm(avgverb ~ treat + forcing + treat * forcing, data = dta.rdd)

## Robust standard errors
library(lmtest)
rse = coeftest(rdd, vcov = vcovHC(rdd, type = "HC2"))
rse2 = coeftest(rdd2, vcov = vcovHC(rdd2, type = "HC2"))

coef = c(coef(rdd)[2], coef(rdd2)[2]) # coefficients/center point
se = c(rse[2,2], rse2[2,2])
ub = coef + 1.96*se # upper bound
lb = coef - 1.96*se # lower bound
var = c("Class Size", "Verbal Exam") # label vars etc.
all = data.frame(var, coef, lb, ub)

require(ggplot2)
pd = position_dodge(.1)
ggplot(all, aes(coef, var)) +
  geom_point(size=3, position=pd) +
  geom_vline(xintercept = 0, colour="red", linetype = "longdash") +
  xlab("Estimated RDD Treatment Effect") + ylab("") +
  geom_errorbarh(aes(xmin=lb, xmax=ub), height = 0.1) + theme_bw() +
  theme(text = element_text(size=20))

## Cluster standard errors
clse = function(model, cluster){
  require(sandwich); require(lmtest)

  # Calculate degree of freedom adjustment
  M = length(unique(cluster))
  N = length(cluster)
  K = model$rank
  dfc = (M/(M-1))*((N-1)/(N-K))

  # Calculate the uj's
  uj  = apply(estfun(model),2, function(x) tapply(x, cluster, sum))

  # Use sandwich to get the var-covar matrix
  vcovCL = dfc*sandwich(model, meat=crossprod(uj)/N)
  return(vcovCL)
}
clse = clse(rdd, dta.rdd$schlcode)[2,2]^0.5
clse2 = clse(rdd2, dta.rdd$schlcode)[2,2]^0.5

coef = c(coef(rdd)[2], coef(rdd2)[2]) # coefficients/center point
se = c(clse, clse2)
ub = coef + 1.96*se # upper bound
lb = coef - 1.96*se # lower bound
var = c("Class Size", "Verbal Exam") # label vars etc.
all = data.frame(var, coef, lb, ub)

require(ggplot2)
pd = position_dodge(.1)
ggplot(all, aes(coef, var)) +
  geom_point(size=3, position=pd) +
```

```r
  geom_vline(xintercept = 0, colour="red", linetype = "longdash") +
  xlab("Estimated RDD Treatment Effect") + ylab("") +
  geom_errorbarh(aes(xmin=lb, xmax=ub), height = 0.1) + theme_bw() +
  theme(text = element_text(size=20))

### Part (d)
frdd = function(threshold, window=5){
  dta.frdd = dta[dta$enrollment >= threshold - window,]
  dta.frdd = dta.frdd[dta.frdd$enrollment <= threshold + window,]
  dta.frdd$forcing = dta.frdd$enrollment - threshold
  dta.frdd$z = ifelse(dta.frdd$enrollment >= threshold, 1, 0)
  return(dta.frdd = dta.frdd)
}
frdd.41 = frdd(41, 5)
frdd.81 = frdd(81, 5)
frdd.121 = frdd(121, 5)
dta.frdd = rbind(frdd.41, frdd.81, frdd.121)

library(AER); library(lmtest)
frdd = ivreg(avgverb ~ classize + forcing + forcing * classize | z + forcing + forcing *
     z, data=dta.frdd)
coeftest(frdd, vcov = vcovHC(frdd, type = "HC2"))
```