ARE213 Problem Set #1B

Peter Alstone & Frank Proulx

October 14, 2013

1 Problem #1

1.1 Part A

Under the assumption of random assignment conditional on the observables, what are the sources of misspecification bias in the estimates generated by the linear model estimated in Problem Set 1a?

Wrong functional form. In Problem Set 1A we used linear (i.e., $y = \beta x + \epsilon$) estimators to make "corrections" while the true functional form of the relationships between the covariates we included in the modern were certainly not linear. By imposing a linear function on a non-linear data generating process (described by the CEF), we introduce misspecification bias in the model.

Omitted Variables Bias. We were able to use variables included in the dataset in our linear model, but not the unobserved variables that may be important for control. If omitted variables exist that both determine outcomes related to birth weight and are correlated with smoking status we will over- or under-estimate the effect (depending on the characteristics of the omission).

1.2 Part B

Now, consider a series estimator. Estimate the smoking effects using a flexible functional form for the control variables (e.g., higher order terms and interactions). What are the benefits and drawbacks to this approach?

Using a

2 Problem #2

2.1 Part A

To calculate the propensity score, we estimated a logit model of mother's tobacco use (0=non-smoker, 1=smoker) as determined by the predetermined covariates shown in Table **??**. Model #1 shows the full model using all of the covariates suspected to be predetermined. Model #2 is a reduced form of the same model, preserving just those covariates that were significant at the 1% level in Model #1.

To test whether the propensity scores predicted by these two models are significantly different we perform a likelihood ratio test between the full and reduced model. This test yields the following output:

NOTE from class: including insignificant terms can be beneficial in terms of getting better fit (by reducing

2.2 Part B

This estimation assumes unconfoundedness. and conditional independence

2.3 Part C

We used R to complete this assignment. The code is below:

```
1  # PROBLEM SET 1B
2  # ARE 213 Fall 2013
3
4
5  # Frank's Directory
6  #setwd("/media/frank/Data/documents/school/berkeley/fall13/are213/are213/ps1
       ")
7
8  # Peter's Directory
9  #setwd("~/Google Drive/ERG/Classes/ARE213/are213/ps1")
10
11
12 # Packages
13 library(foreign) #this is to read in Stata data
14 library(Hmisc)
15 library(psych)
16 library(stargazer)
17 library(ggplot2) # for neato plotting tools
```

Table 1: Propensity scores calculated for mother's smoking status

	Mother Tobacco-Use Status	
	(1)	(2)
Mother's Race not White or Black	−1.956***	−1.954***
	(0.134)	(0.133)
Mother's Years of Education	−0.817***	−0.818***
	(0.028)	(0.028)
Marital status	−0.205***	−0.204***
	(0.005)	(0.005)
Father's age	−1.256***	−1.251***
	(0.022)	(0.021)
Father's Years of Education	0.029***	0.030***
	(0.002)	(0.001)
Father Mexican	−0.131***	−0.131***
	(0.005)	(0.005)
Father Puerto Rican	−1.961***	−1.957***
	(0.173)	(0.173)
Father Cuban	−1.267***	−1.268***
	(0.058)	(0.058)
Father Central or South American	−0.567	−0.567
	(0.364)	(0.364)
Father Race Other or Unknown Hispanic	−1.933***	−1.932***
	(0.205)	(0.205)
Plurality of Infant	−0.890***	−0.889***
	(0.120)	(0.120)
Sex of Infant	−0.148***	
	(0.054)	
Mother's age	−0.019	
	(0.017)	
dmage	0.003	
	(0.002)	
Constant	2.873***	2.707***
	(0.088)	(0.064)
N	114,610	114,610
Log Likelihood	-44,310.690	-44,315.790
Akaike Inf. Crit.	88,651.370	88,655.580

Notes: ***Significant at the 1 percent level.

```
library(plyr) # for nice data tools like ddply
library(epicalc) # For likelihood ratio test
library(car) # "companion for applied regression" - recode fxn, etc.
library(gmodels) #for Crosstabs
library(splines) # for series regression

# Homebrewed functions
source("../util/are213-func.R")
source("../util/watercolor.R") # for watercolor plots

# Data
ps1.data <- read.dta(file="ps1.dta")

# Data Cleaning Step
full.record.flag <- which(ps1.data$cardiac != 9 &
                            ps1.data$cardiac != 8 &
                            ps1.data$lung != 9 &
                            ps1.data$lung != 8 &
                            ps1.data$diabetes !=9 &
                            ps1.data$diabetes !=8 &
                            ps1.data$herpes != 9 &
                            ps1.data$herpes != 8 &
                            ps1.data$chyper != 9 &
                            ps1.data$chyper != 8 &
                            ps1.data$phyper != 9 &
                            ps1.data$phyper != 8 &
                            ps1.data$pre4000 !=9 &
                            ps1.data$pre4000 !=8 &
                            ps1.data$preterm != 9 &
                            ps1.data$preterm != 8 &
                            ps1.data$tobacco != 9 &
                            ps1.data$cigar != 99 &
                            ps1.data$cigar6 !=6 &
                            ps1.data$alcohol != 9 &
                            ps1.data$drink != 99 &
                            ps1.data$drink5 !=5 &
                            ps1.data$wgain !=99
                          )

ps1.data$full.record <- FALSE # initialize column as F
ps1.data$full.record[full.record.flag] <- TRUE #reassign level to T for full
     records

ps1.data.clean <- subset (ps1.data, full.record == TRUE)
ps1.data.missingvalues <- subset(ps1.data, full.record == FALSE)

# Problem 1a is only in ps1b.tex
# Problem 1b
# This is using a series estimator. I think smooth.spline() is the right
    function to use, but let me know if you think we should be doing kernel
    regression instead. I'm also not sure how to go about adding interaction
     terms



sm.flex <- with(ps1.data.clean, smooth.spline(cigar, y=dbrwt, nknots=10,
    spar = 0.7, tol = 0.0001)) # Fits a smooth line to the data
```

```
sm.flex.df <- data.frame(sm.flex$x, sm.flex$y) #converts the fitted values into a data frame for ggplot

splineplot <- ggplot(sm.flex.df, aes(x = sm.flex.x, y=sm.flex.y))
splineplot <- splineplot +
  geom_point(data=ps1.data.clean, aes(x = cigar, y = dbrwt), pch = 1) +
  geom_line(color='red') +
  labs(x = 'Cigarettes smoked per day by mother', y= 'Birthweight')
splineplot

ggsave(filename = 'img/splineplot.pdf')


# Problem 2a
ps1.data.clean$tobacco.rescale <- with(ps1.data.clean, recode(tobacco, "2='0'", as.numeric.result=TRUE)) #rescales the tobacco use variable to be 0/1, where 0=no and 1 = yes
ps1.data.clean$dmar.rescale <- with(ps1.data.clean, recode(dmar, "2='0'"))

smoke.propensity.all <- glm(tobacco.rescale ~ as.factor(mrace3) + dmeduc + dmar.rescale + dfage + dfeduc + as.factor(orfath) + dplural + csex + dmage, data=ps1.data.clean, family = binomial()) ## Did I miss any predetermined covariates here?

smoke.propensity.reduced <- glm(tobacco.rescale ~ as.factor(mrace3) + dmeduc + dmar.rescale + dfage + dfeduc + as.factor(orfath), data=ps1.data.clean, family = binomial())


stargazer(smoke.propensity.all, smoke.propensity.reduced,
          type = "latex",
          covariate.labels = c("Mother's Race not White or Black", "Mother's Years of Education", "Marital status", "Father's age", "Father's Years of Education", "Father Mexican", "Father Puerto Rican", "Father Cuban", "Father Central or South American", "Father Race Other or Unknown Hispanic", "Plurality of Infant", "Sex of Infant", "Mother's age"),
          style ="qje",
          align = TRUE,
         font.size = "scriptsize",
         label = "tab:propensities",
         title = "Propensity scores calculated for mother's smoking status \n
logit model specification",
          dep.var.labels = "Mother Tobacco-Use Status",
          out = "propensityscores.tex"
          )

ps1.data.clean$propensityfull <- predict(smoke.propensity.all, type = "response")
ps1.data.clean$propensityreduced <- predict(smoke.propensity.reduced, type = "response")

sink(file = "lrtest.tex", append = FALSE)
lrtest(smoke.propensity.all, smoke.propensity.reduced) #Test whether the two scores are statistically different
sink()
```

```
110
111 #Problem 2b - Estimating a regression model using propensity scores
112
113 sm.propensityregression <- lm(dbrwt ~ tobacco.rescale + (propensityreduced *
      tobacco.rescale) + propensityreduced, ps1.data.clean)
114
115 #calculation of average treatment effect:
116 coefficients(sm.propensityregression)[2] + coefficients(sm.
    propensityregression)[4]*mean(ps1.data.clean$propensityreduced)
117
118
119 stargazer(sm.propensityregression,
120           type = "latex",
121           covariate.labels = c("Delta1", "Beta", "Delta2", "Constant"),
122           style ="qje",
123           align = TRUE,
124           font.size="footnotesize",
125          label = "tab:propensitymodel",
126          title = "Model of effects of tobacco use on birthweight using
               propensity score as a control",
127           dep.var.labels = "Mother Tobacco-Use Status",
128           out = "propensityscoremodel.tex"
129           )
130
131 #Problem 2c - Using reweighting with propensity scores
132
133 term1 <- with(ps1.data.clean, sum((tobacco.rescale*dbrwt)/propensityreduced)
    /sum(tobacco.rescale/propensityreduced))
134 term2 <- with(ps1.data.clean, sum(((1-tobacco.rescale)*dbrwt)/(1-
    propensityreduced))/sum((1-tobacco.rescale)/(1-propensityreduced)))
135
136 weightingestimator <- term1-term2 #This should be the average treatment
    effect
137
138 term1.T <- with(subset(ps1.data.clean, tobacco.rescale=1), sum((tobacco.
    rescale*dbrwt)/propensityreduced)/sum(tobacco.rescale/propensityreduced)
    )
139 #term2.T <- with(subset(ps1.data.clean, tobacco.rescale=1), sum(((1-tobacco.
    rescale)*dbrwt)/(1-propensityreduced))/sum((1-tobacco.rescale)/(1-
    propensityreduced)))
140
141 weightingestimator.T <- term1.T#-term2.T #This should be the average
    treatment on treated
142
143
144
145 # Problem 2d - Kernel Density Estimator
146 tot.propensity.nosm <- with(subset(ps1.data.clean, tobacco.rescale == 0),
    sum(propensityreduced))
147 tot.propensity.sm <- with(subset(ps1.data.clean, tobacco.rescale == 1), sum(
    propensityreduced))
148 h <- 35 # This is to play with the bandwidth
149
150 kerndensity.nosm <- with(subset(ps1.data.clean, tobacco.rescale == 0),
    density(dbrwt, #if nobody smoked
151         kernel = "epanechnikov",
152         bw = h,
```

```
        weights = propensityreduced/tot.propensity.nosm))
kerndensity.nosm.df <- data.frame(kerndensity.nosm[1], kerndensity.nosm[2])

kerndensity.sm <- with(subset(ps1.data.clean, tobacco.rescale == 1), density
    (dbrwt, #if everybody smoked
        kernel = "epanechnikov",
        bw = h,
        weights = propensityreduced/tot.propensity.sm))
kerndensity.sm.df <- data.frame(kerndensity.sm[1], kerndensity.sm[2])

kerndensity.plot <- ggplot(kerndensity.nosm.df, aes(x, y))
kerndensity.plot <- kerndensity.plot +
  geom_line(linetype = 'dotted') +
  geom_line(data = kerndensity.sm.df, aes(x, y)) +
  labs(title = paste("Density of birthweights estimated using \n propensity
      score-weighted kernel regression \n Bandwidth=", as.factor(h)), x = "
      Birthweight (grams)", y = "Density") +
  guides(linetype = "Legend") # Having trouble getting a legend.

kerndensity.plot

ggsave(file = 'img/kerndensity.pdf', plot = kerndensity.plot)

#2d - calculating kernel value by 'hand' at dbrwt = 3000
# y_pred = sum_onj(K(pi-pj/h)*yj)/sum_onj(K(pi-pj/h))

#h <- 30
#kernel.epa <- function(u){
#return(if (abs(u) < 1){0.75*(1-u*u))}
#}
#propensity3000 <- # I'm not sure how we get the estimated propensity score
    at dbrwt = 3000


#Problem 2e
## This is in progress
#kerndensity.plot.bws <- ggplot()
#for(h in seq(from = 15, to = 40, by = 5)) {
#paste0('kerndensity.sm.', h) <- with(subset(ps1.data.clean, tobacco.rescale
     == 1), density(dbrwt, #if everybody smoked
#         kernel = "epanechnikov",
#         bw = h,
#         weights = propensityreduced/tot.propensity.sm))
#paste0('kerndensity.sm.df.', h) <- data.frame(paste0('kerndensity.sm.', h
    ,'[1]'), paste0('kerndensity.sm.', h, '[2]'))
#kerndensity.plot.bws <- kerndensity.plot.bws +
#  geom_line(data = paste0('kerndensity.sm.df.', h), aes(x, y)) +
#  labs(title = "Density of birthweights estimated using \n propensity score
    -weighted kernel regression", x = "Birthweight (grams)", y = "Density")
#}

#kerndensity.plot.bws
```

ps1b.R

```
# Econometrics helper functions for [R]
#
```