Kaicheng Luo

# Assignment 1

Stat 157
September 23, 2019

## 1. Specification searches

### 1.1 Data Prep

```
dat <- read.table("cps1re74.csv",header=T)
# unemployed
dat$u74 <- as.numeric(dat$re74==0)
dat$u75 <- as.numeric(dat$re75==0)
## linear regression on the outcome
lmoutcome = lm(re78 ~ ., data = dat)
lmoutcome$coefficients['treat']
```

```
##    treat
## 1067.546
```

```
lmoutcome = lm(re78 ~ treat, data = dat)
```

### 1.2 Function Design

Run 1024 linear regressions with subsets of covariates, and report the regression coefficients of the treatment. How many are positively significant, how many are negatively significant, and how many are not significant?

```
alpha <- 0.05
positive <- 0
negative <- 0
R2 <- rep(0,1024)

# filter the dependent variable and treatment to ensure the 1-1 mapping
# between our index and the variable
ind_dat <- dat %>%
  select(-"re78", -"treat")

# The main idea here is to use bit-operation to perform the loop.
# For each variable j, the index we created in listI[[1]][j]
# determined whether it shall be included in our lm.
# Besides the statistical inference of the coefficient, we also report R^2
for (i in 0:1023){
  binaryI <- intToBin(i)
  strI <- as.character(binaryI)
  listI <- strsplit(strI, "")
  modelSpecification <- "re78 ~ treat"

  # The model specification is hence decided and updated
  # according to the different values of i
  for (j in 1:nchar(strI)){
    if (listI[[1]][j] == "1"){
      modelSpecification <- paste(modelSpecification, "+", colnames(ind_dat)[j])
    }
  }
  lmtemp <- lm(modelSpecification, data = dat)
  R2[i+1] <- as.numeric(summary(lmtemp)['r.squared'])
  if (summary(lmtemp)$coefficients['treat','Pr(>|t|)'] < alpha
      & lmtemp$coefficients['treat'] > 0){
    positive <- positive + 1
  }
  if (summary(lmtemp)$coefficients['treat','Pr(>|t|)'] < alpha
      & lmtemp$coefficients['treat'] < 0){
    negative <- negative + 1
  }
}
negative
```

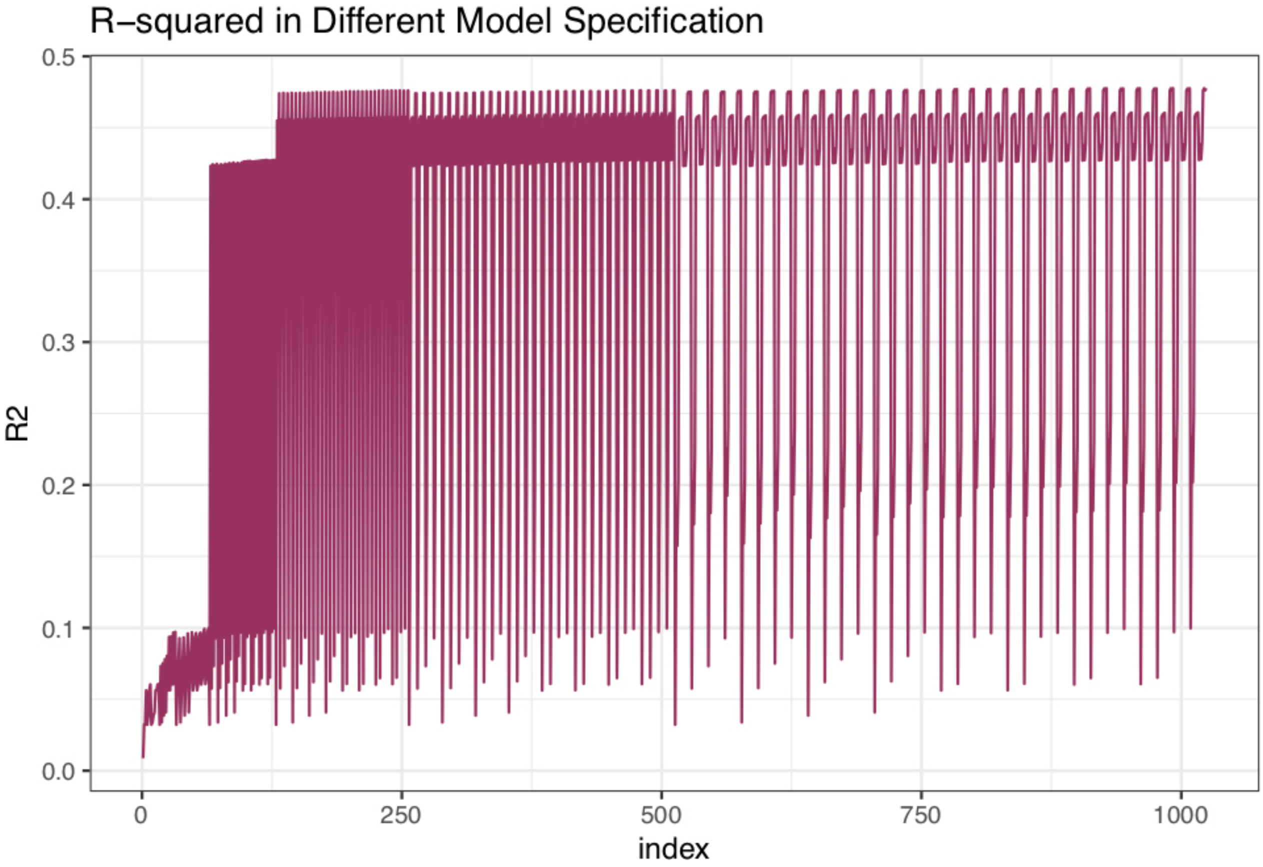
```
## [1] 198
```

```
positive
```

```
## [1] 125
```

There’re 198 negative significant coefficients of treatment and 125 positive ones.

```
# Also some interesting results concerning the choice of parameters.
R2<- as.data.frame(R2)
R2 %>%
  mutate(index = 1:length(R2)) %>%
  ggplot() + theme_bw() +
  geom_line(aes(x = index, y = R2),color = 'maroon') +
  labs(
    title = "R-squared in Different Model Specification"
  )
```



```
# The model with the best prediction, in terms of maximizing the square of correlation
which(R2 == max(R2))
```

```
## [1] 1024
```

```
# which is the full model with all covariates!
```

But $R^2$ remains almost unchanged as we include the second covariate in our model. Some severe multi-collinearity is actually occuring in our estimation.

## 2. More on racial discrimination

Conduct two subgroup analyses:

```
resume = read.csv("resume.csv")

# Subsetting
male_resume <- resume %>%
  filter(sex == "male")
```

```
female_resume <- resume %>%
  filter(sex == "female")

male_table = table(male_resume$race, male_resume$call)
male_table
```

```
##
##           0   1
##   black 517  32
##   white 524  51
```

```
fisher.test(male_table)
```

```
##
##  Fisher's Exact Test for Count Data
##
## data:  male_table
## p-value = 0.05317
## alternative hypothesis: true odds ratio is not equal to 1
## 95 percent confidence interval:
##  0.9730068 2.5718680
## sample estimates:
## odds ratio
##   1.571824
```

```
female_table = table(female_resume$race, female_resume$call)
female_table
```

```
##
##            0    1
##   black 1761  125
##   white 1676  184
```

```
fisher.test(female_table)
```

```
##
##  Fisher's Exact Test for Count Data
##
## data:  female_table
## p-value = 0.0002856
## alternative hypothesis: true odds ratio is not equal to 1
## 95 percent confidence interval:
##  1.213027 1.976560
## sample estimates:
```

```
## odds ratio
##   1.546461
```

Within 95 % confident level, racial discrimination for men-employers are insignificant, with a p-value around 0.53. Yet cincerning women employers, the observed odds ratio is significantly different from 0 (p<0.001). Though it might be irrigorous to jump to the conclusion that racial discrimination exists more in women's job market (Due to ommitted variable bias.) It's a clear indicator that women are, directly or indirectly, suffering more from the prejudice.

## 3. Regression adjustment in the Fisher Randomization Test

### 3.1 Using the residuals

```
library(Matching)
data(lalonde)
result <- lm(re78~ -treat, data = lalonde)
z <- lalonde$treat
y <- result$residuals

# Monte-Carlo Simulation of data
MC = 10^3
Tauhat   = rep(0, MC)
Student  = rep(0, MC)
Wilcox   = rep(0, MC)
Ks       = rep(0, MC)
tau = t.test(y ~ z, var.equal = TRUE)$statistic
t = t.test(y ~ z, var.equal = FALSE)$statistic
w = wilcox.test(y ~ z)$statistic
ks = ks.test(y[z == 1], y[z == 0])$statistic

extreme_tau = 0
extreme_t = 0
extreme_w = 0
extreme_ks = 0
for(mc in 1:MC){
   zperm = sample(z)
   temptau = t.test(y ~ zperm, var.equal = TRUE)$statistic
   tempt = t.test(y ~ zperm, var.equal = FALSE)$statistic
   tempw = wilcox.test(y ~ zperm)$statistic
```

```
    tempks = ks.test(y[zperm == 1], y[zperm == 0])$statistic
    if (abs(temptau) > abs(tau)){
      extreme_tau <- extreme_tau + 1
    }
    if (abs(tempt) > abs(t)){
      extreme_t <- extreme_t + 1
    }
    if (abs(tempw) < abs(w)){
      extreme_w <- extreme_w + 1
    }
    if (abs(tempks) > abs(ks)){
      extreme_ks <- extreme_ks + 1
    }
}
```

The exact p-value using tau-statistic is 0.002
The exact p-value using t-statistic is 0.004
The exact p-value using w-statistic is 0.004
The exact p-value using ks-statistic is 0.034
Those four p-values are jusified because the taking the residual in a regression is simply a de-mean operation, subtrating the conditional mean of the dependent variable given all the covariates other than treatment itself. The imbalance of covariates are hence controlled after the substraction. Almost always, it reduces the variance of our estimation.

### 3.2 Using the residuals

```
library(Matching)
data(lalonde)
z <- lalonde$treat
y <- lalonde$re78

# Monte-Carlo Simulation of data
MC = 10^3
current_coef <- lm(re78~., data = lalonde)$coefficients['treat']
lm(re78~treat:., data = lalonde)$coef
```

```
##   (Intercept)         treat     treat:age    treat:educ   treat:black
##  4.554802e+03 -7.683564e+03  5.798286e+01  5.482296e+02 -5.904062e+02
##    treat:hisp treat:married  treat:nodegr    treat:re74    treat:re75
##  1.262090e+03  9.356968e+02 -7.706816e+02  2.925472e-01  1.192415e-02
##     treat:u74     treat:u75
##  6.920888e+03 -4.052482e+03
```

```r
extreme_coef <- 0
for(mc in 1:MC){
  zperm = sample(z)
  data_copy <- lalonde
  data_copy$treat <- zperm
  coef <- lm(re78~., data = data_copy)$coefficients['treat']
  if (coef > current_coef){
    extreme_coef = extreme_coef + 1
  }
}
```

The exact p-value using coef-statistic is 0.005
It's plausible to use coefficients as our statistic because it simply denotes the average treatment effect controlling all those covariates constant. It can be considered as the subtraction of two regression coefficients, indicating the difference in means. (See Lin(2013) for detailed discussion)

# 4. Correlation and Partial Correlation

### 4.1 Express $\rho_{XY|Z}$ using $\rho_{XY}, \rho_{YZ}, \rho_{XZ}$

First, standardize our data,

$$let \quad x = \frac{X - \mu_X}{se(X)}, y = \frac{Y - \mu_Y}{se(Y)}, z = \frac{Z - \mu_Z}{se(Z)} \tag{1}$$

We know that in a univariate regression, the coefficient $\beta$ of the treatment variable equals the correlation index $\rho_{XY}$. To calculate the conditional correlation, we shall first regress x on z and y on z, respectively, to calculate the residuals.

$$x = z\rho_{XZ} + u_x \tag{2}$$

where $var(x) = var(z) = 1$ (standardized), and $var(u_x) = (1 - \rho^2_{XZ})$

$$y = z\rho_{YZ} + u_y \tag{3}$$

where $var(y) = var(z) = 1$ , and $var(u_y) = (1 - \rho^2_{YZ})$
Regress $u_y$ on $u_x$

$$\rho_{XY|Z} = \frac{cov(u_x, u_y)}{\sqrt{var(u_x)var(u_y)}} \tag{4}$$

Implementing out results in (2) and (3),

$$\begin{aligned}\rho_{XY|Z} &= \frac{\sum u_x u_y}{n\sqrt{(1-\rho_{XZ}^2)(1-\rho_{YZ}^2)}} \\ &= \frac{\sum (x - z\rho_{XZ})(y - z\rho_{YZ})}{n\sqrt{(1-\rho_{XZ}^2)(1-\rho_{YZ}^2)}} \\ &= \frac{\rho_{XY} - 2\rho_{XZ}\rho_{ZY} + \rho_{XZ}\rho_{ZY}}{\sqrt{(1-\rho_{XZ}^2)(1-\rho_{YZ}^2)}} \\ &= \frac{\rho_{XY} - \rho_{XZ}\rho_{ZY}}{\sqrt{(1-\rho_{XZ}^2)(1-\rho_{YZ}^2)}}\end{aligned} \tag{5}$$

### 4.2 Give an example with $\rho_{XY} > 0$ and $\rho_{XY|Z} < 0$.

If $\rho_{XY} = 0.5$, $\rho_{YZ} = 0.7$ $\rho_{XZ} = 0.8$, then $\rho_{XY|Z} = -0.071 < 0$

## 5. Nonlinear causal estimands

### 5.1 Examples of the differents in estimands

Example for $\sigma_1 = \sigma_2$

```
x <- seq(1,10,1)
y <- seq(2,20,2)
sigma_1 <- median(y) - median(x)
sigma_2 <- median(y-x)
sigma_1-sigma_2
```

```
## [1] 0
```

```
# As long as the sign of x', y' are the same, the equation holds.
```

Example for $\sigma_1 < \sigma_2$

```
x <- seq(1,10,1)
y[10] <- 2
sigma_1 <- median(y) - median(x)
sigma_2 <- median(y-x)
sigma_1-sigma_2
```

```
## [1] -1
```

Example for $\sigma_1 > \sigma_2$

```
y <- seq(2,20,2)
x[10] <- 1
sigma_1 <- median(y) - median(x)
sigma_2 <- median(y-x)
sigma_1-sigma_2
```

```
## [1] 1
```

### Find the better one

Intuitively, the medium of differences, instead of the difference in medians makes more sense. When we're substracting the difference of the medians, the pre-treatment value and post-treatment value possibly will not be attributed to the same observation. In other words, we could be deriving a statistic based on some data that is meaningless in reality. So in most cases, chosing the first statistic is always a safe choice. However, it still depends on the scenario that we apply the statistic. The following example shows that, if our goal is, say, simply to get rid of some outliers. Given normal distributions of our original data, the difference-in-medians statistic show a better property – with less bias and less variance as well.

```
# Monte-Carlo Simulation
# Assume that we already have the science table
# Also assume some normal distributions of our treatment effect
MC <- 1e4
sum_of_squares_1 <- 0
sum_of_squares_2 <- 0
sum_1 <- 0
sum_2 <- 0
for (mc in 1:MC){
  Before <- rnorm(1000,0,1)
  treat <- c(rep(1,500), rep(0,500))
  Treatment_effect <- rnorm(1000,1,1) * treat
  Control_effect <- rnorm(1000,0,1) * (1-treat)
  After <- Before + Treatment_effect + Control_effect
  y <- After[1:500]
  x <- After[501:1000]
  sigma_1 <- median(y) - median(x)
  sigma_2 <- median(y-x)
  sum_of_squares_1 <- sum_of_squares_1 + (sigma_1-1)^2
  sum_of_squares_2 <- sum_of_squares_2 + (sigma_2-1)^2
  sum_1 <- sum_1 + sigma_1
  sum_2 <- sum_2 + sigma_2
}
```

## 6. A better bound of the variance formula

Take the subtraction,

$$\begin{aligned} RHS - LHS &= \frac{n_0}{nn_1}S_1^2 + \frac{n_1}{nn_0}S_0^2 + \frac{2}{n}S_0S_1 \\ &= \frac{S_1^2}{n} + \frac{S_0^2}{n} + \frac{2S_0S_1}{n} - \frac{S_\tau^2}{n} \\ &= [(S_0+S_1)^2 - S_\tau^2]/n \end{aligned} \tag{6}$$

Note that

$$\begin{aligned} S_0^2 &= \frac{1}{N-1}\sum_{i=1}^{n}(Y_i(0) - Y_i\bar{(0)})^2 \\ S_1^2 &= \frac{1}{N-1}\sum_{i=1}^{n}(Y_i(1) - Y_i\bar{(1)})^2 \\ S_\tau^2 &= \frac{1}{N-1}\sum_{i=1}^{n}(Y_i(1) - Y_i\bar{(1)} - (Y_i(0) - Y_i\bar{(0)}))^2 \end{aligned} \tag{7}$$

To simplify our notation, the equation above can be re-written as

$$\sum(a+b)^2 - (\sqrt{\sum a^2} + \sqrt{\sum b^2})^2 \tag{8}$$

Here we use Cauchy inequality

$$\sum a^2 \sum b^2 \geq \sum |ab|^2 \tag{9}$$

$$LHS = RHS \iff \frac{Y_i(0) - Y_i\bar{(0)}}{Y_i(1) - Y_i\bar{(1)}} = \frac{Y_j(0) - Y_j\bar{(0)}}{Y_j(1) - Y_j\bar{(1)}} \qquad \forall i,j \in [1,N] \tag{10}$$

Q.E.D.