A4

Yuan Tien

4/9/2022

```r
rm(list = ls())
getwd()
```

```
## [1] "/Users/yuantien/Desktop/R/613/A4"
```

```r
setwd("/Users/yuantien/Desktop/R/613/A4")
library(tidyverse)
```

```
## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.1 ──
```

```
## ✓ ggplot2 3.3.5     ✓ purrr   0.3.4
## ✓ tibble  3.1.6     ✓ dplyr   1.0.7
## ✓ tidyr   1.1.4     ✓ stringr 1.4.0
## ✓ readr   2.1.1     ✓ forcats 0.5.1
```

```
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```r
library(data.table)
```

```
##
## Attaching package: 'data.table'
```

```
## The following objects are masked from 'package:dplyr':
##
##     between, first, last
```

```
## The following object is masked from 'package:purrr':
##
##     transpose
```

```r
dat <- fread("dat_A4.csv")
```

Exercise I

1.1 To my understanding, the latest interview is in 2019, so we compute age base on 2019 - birth year

```r
dat97 <- dat %>%
  mutate(age = 2019 - KEY_BDATE_Y_1997)
dat97$expweek <- rowSums(dat97[,18:28], na.rm = T)

#Since a year has 52.1429 weeks approximately, I will divide weeks by this number
dat97$work_exp <- dat97$expweek/52.1429
```

1.2 Use YSCH-3113 to compute

GED is equivalent to 12th grade (12 years of schooling) see: http://usgei.org/high-school-equivalency-ged/# (http://usgei.org/high-school-equivalency-ged/#):~:text=GEI%20offers%20international%20students%20the,globally%20have%20earned%20GED%20diplomas.

I assume degree earned = None is 0 years of schooling Associate degree and junior college normally takes two years. I assume they take 12+2 = 14 years of schooling Assume normal college degree takes 4 years (12+4 = 16) I assume master's degree takes additional 2 years out of college (12+4+2), and PhD takes 4 years out of college (12+ 4 +4) =20

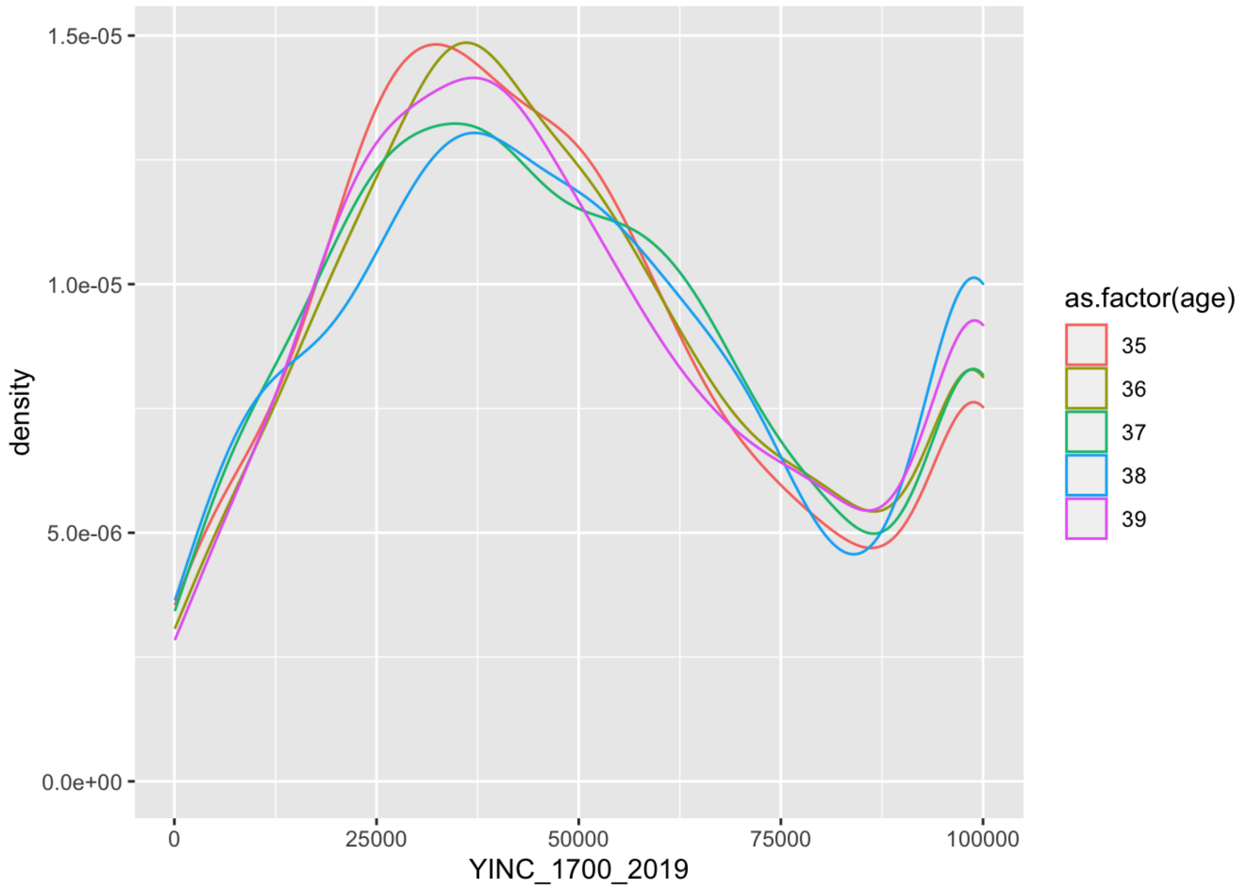
I assume DDS, JD and MD takes 4 years out of college (12+4+4 = 20)

```
s = dat97$YSCH.3113_2019
news <- recode(s, '1' = 0, "2" = 12, "3" = 12, "4" = 14, "5" = 16, "6" = 18 , "7" = 20, "8"
= 20, "-1" = 0, "-2" = 0) #here I assume those refuse to answer and those who don't know th
eir schooling as 0 year of schooling.
dat97$schooling = news

dat97$biodad_school <- dat97$CV_HGC_BIO_DAD_1997
dat97$biomom_school <- dat97$CV_HGC_BIO_MOM_1997
dat97$resdad_school <- dat97$CV_HGC_RES_DAD_1997
dat97$resmom_school <- dat97$CV_HGC_RES_MOM_1997

#I leave "Ungraded = 95" as it be for now

# dat97$biodad_school <- recode(dat97$biodad_school, "95" = 0)
# dat97$biomom_school <- recode(dat97$biomom_school, "95" = 0)
# dat97$resdad_school <- recode(dat97$resdad_school, "95" = 0)
# dat97$resmom_school <- recode(dat97$resmom_school, "95" = 0)
```

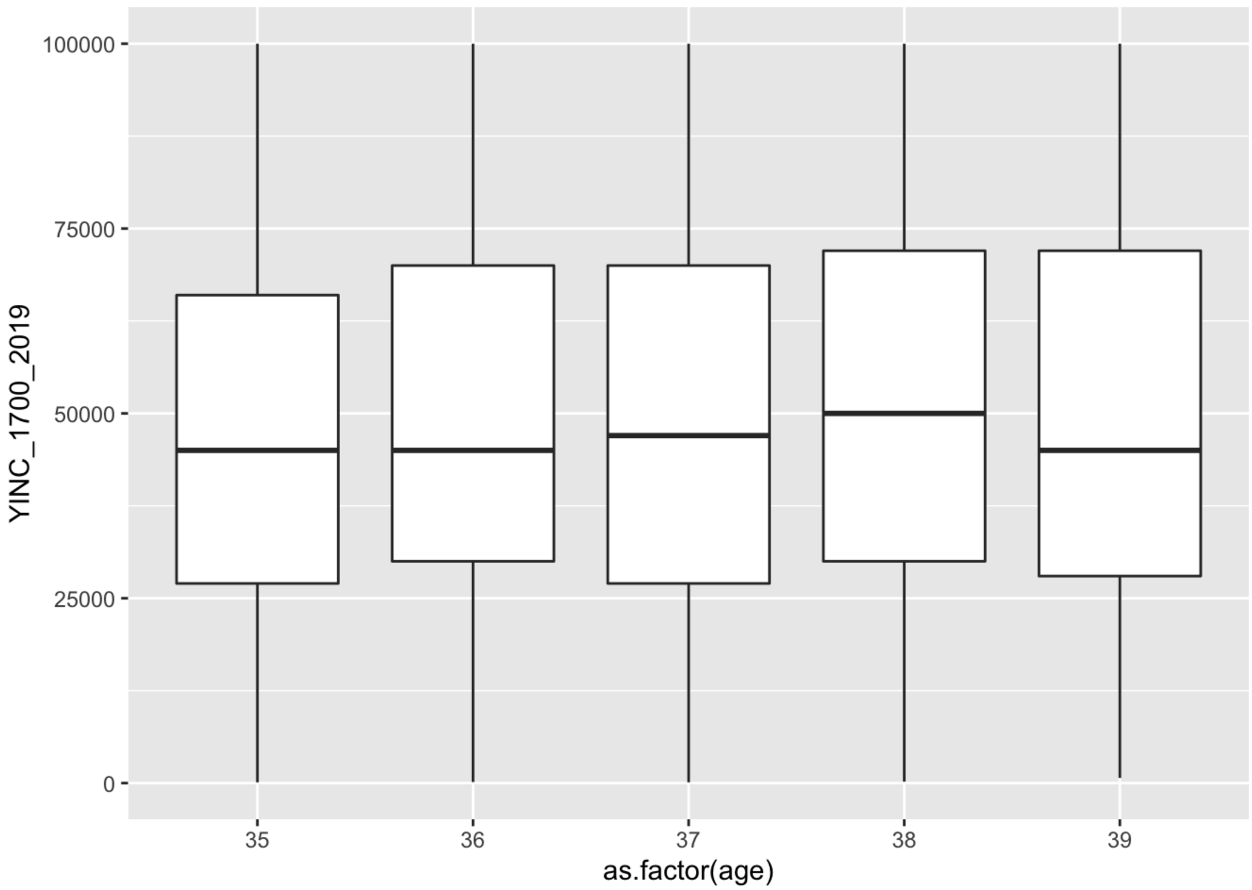
1.3.1

*Note that I provide interpretation right after producing the graphs and also after all the graphs are produced.

```
#Income distribution by age group
dat97%>%
  filter(YINC_1700_2019 > 0) %>%
  ggplot(aes(x = YINC_1700_2019, color = as.factor(age))) + geom_density()
```

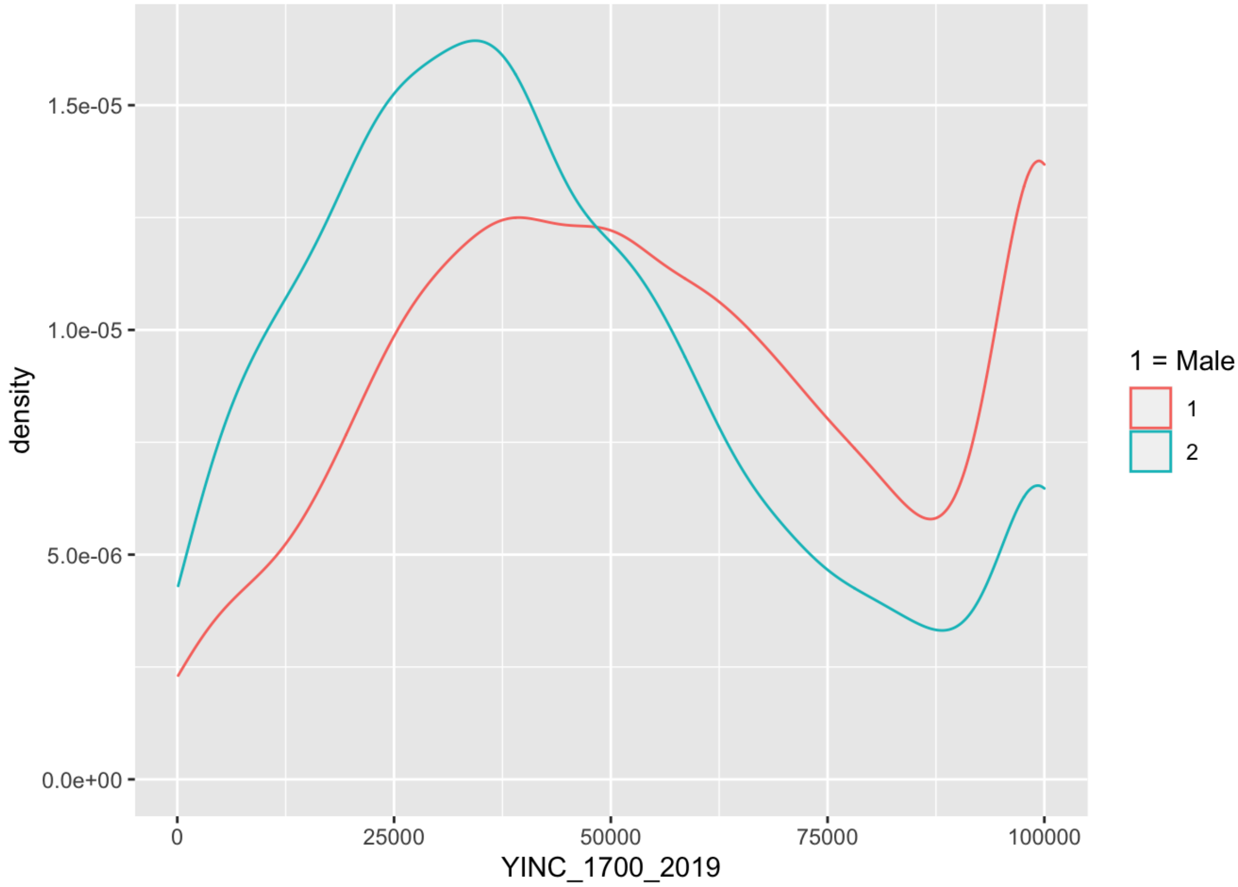



```
dat97%>%
  filter(YINC_1700_2019 > 0) %>%
  ggplot(aes(x = as.factor(age), y = YINC_1700_2019, )) + geom_boxplot()
```

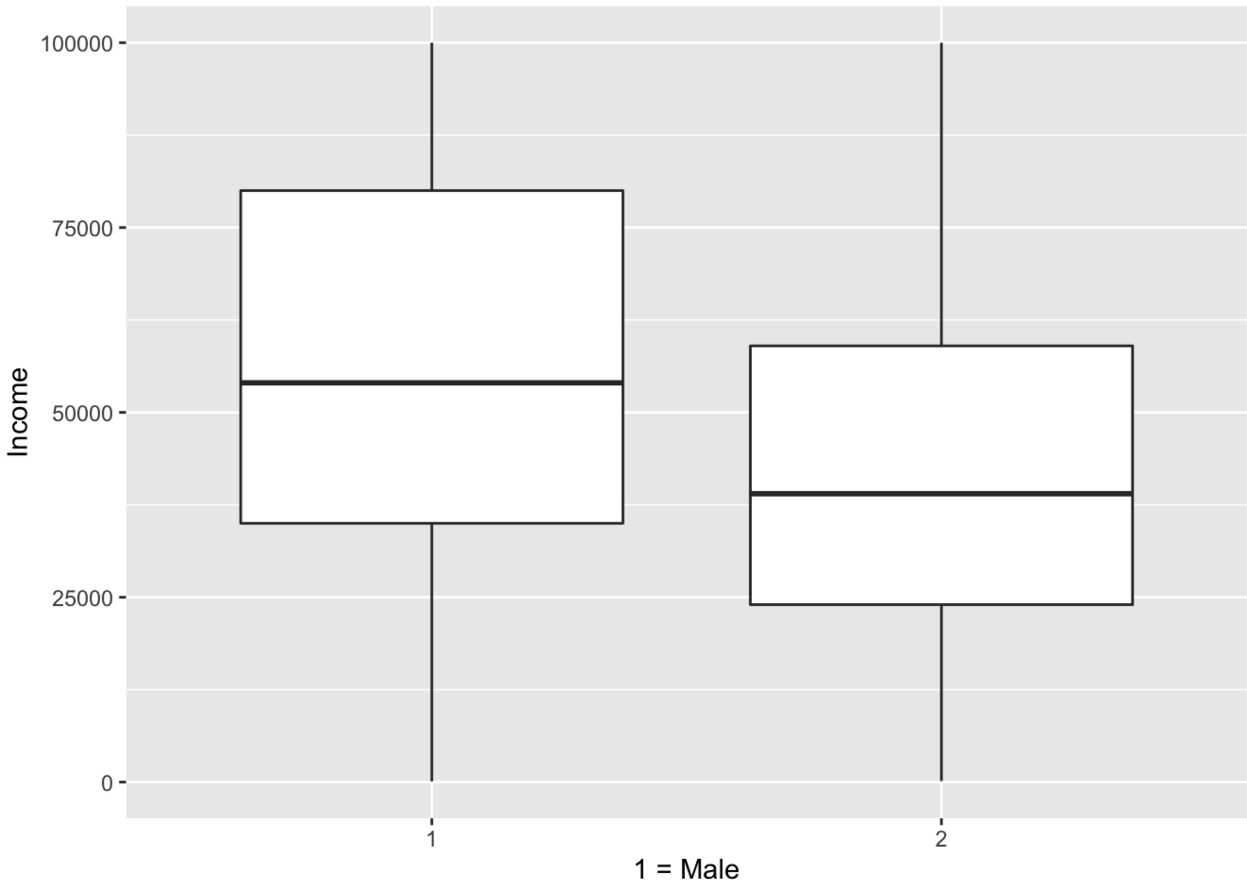


The graph shows that the income seems to be increasing with age but not much. The mean income of age 39 is actually less than mean of age group 38

```
dat97%>%
  filter(YINC_1700_2019 > 0) %>%
  ggplot(aes(x = YINC_1700_2019, color = as.factor(KEY_SEX_1997))) + geom_density() + labs
(color = "1 = Male")
```

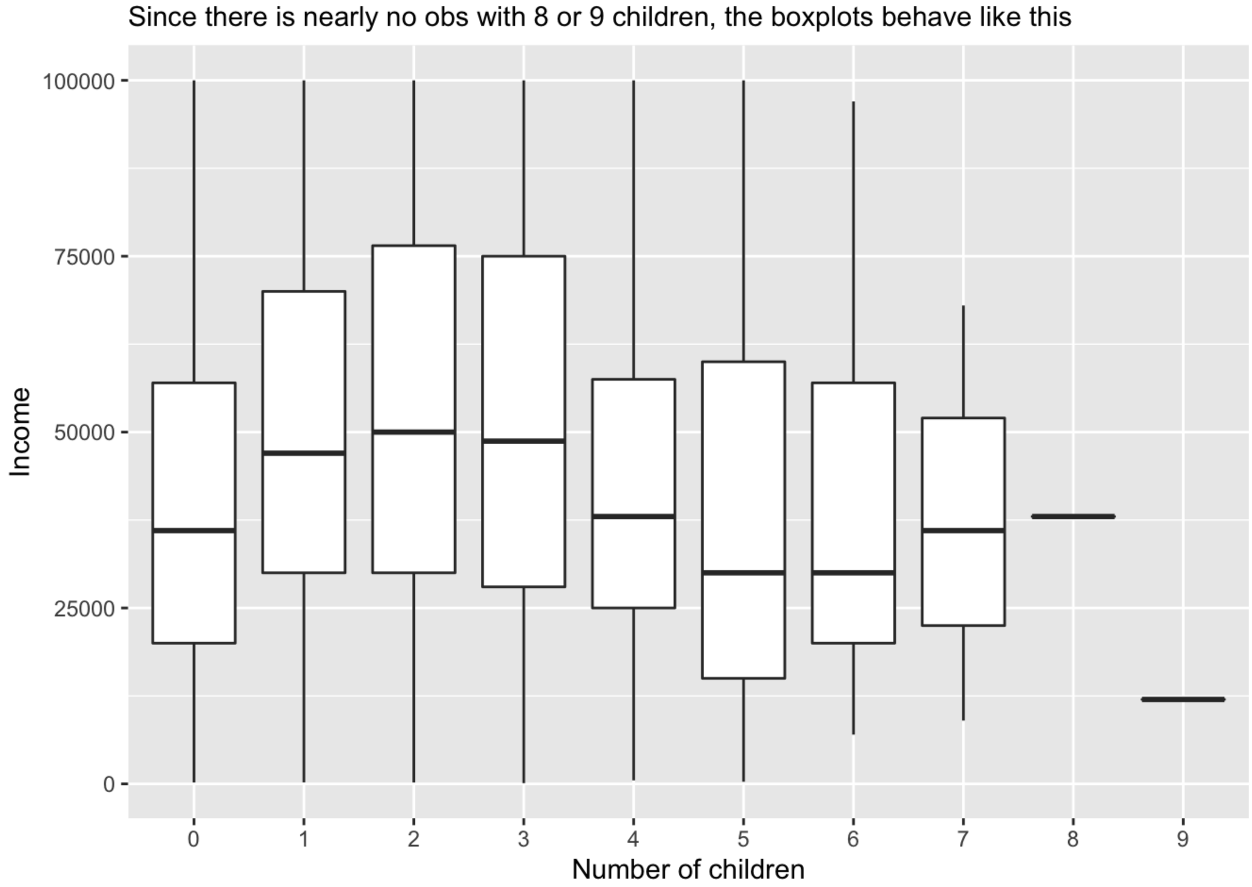



```
dat97%>%
  filter(YINC_1700_2019 > 0) %>%
  ggplot(aes(x = as.factor(KEY_SEX_1997), y = YINC_1700_2019)) + geom_boxplot() + xlab("1 =
Male") +ylab("Income")
```



The graph shows that male tends to earn more than female in our sample with higher mean and distribution.

```
dat97%>%
  filter(YINC_1700_2019 > 0) %>%
  filter(CV_BIO_CHILD_HH_U18_2019 >= 0) %>%
  ggplot(aes(x = as.factor(CV_BIO_CHILD_HH_U18_2019), y = YINC_1700_2019)) + geom_boxplot()
+ xlab("Number of children")+ylab("Income") + labs(subtitle = "Since there is nearly no obs
with 8 or 9 children, the boxplots behave like this")
```



The graph shows that small to middle families on average earn more than no children or many children families.

1.3.2 Income Age

```
income_age <- as.data.frame.matrix(table(dat97$age, dat97$YINC_1700_2019))
income_age$share_of_zero <- (income_age[,1])/rowSums(income_age)

tibble(age = 35:39, share_of_zero = income_age$share_of_zero)
```

```
## # A tibble: 5 × 2
##     age share_of_zero
##   <int>         <dbl>
## 1    35       0.00929
## 2    36       0.00630
## 3    37       0.00542
## 4    38       0.00896
## 5    39       0.00299
```

The table shows that age group 35 has the most share of zero income (maybe those unemployed) among all the age groups.

Gender Income

```
  income_gender <- as.data.frame.matrix(table(dat97$KEY_SEX_1997, dat97$YINC_1700_2019))
  income_gender$share_of_zero <- (income_gender[,1])/rowSums(income_gender)

 tibble(sex = c("male", "female"), share_of_zero = income_gender$share_of_zero)
```

```
## # A tibble: 2 × 2
##   sex    share_of_zero
##   <chr>          <dbl>
## 1 male          0.0075
## 2 female        0.00574
```

The table shows that there is a larger share of no income men than the share of no income women in our sample.

```
income_child <- as.data.frame.matrix(table(dat97$CV_BIO_CHILD_HH_U18_2019, dat97$YINC_1700_
2019))
income_child$share_of_zero <- (income_child[,1])/rowSums(income_child)

tibble(number_of_children = 0:9, share_of_zero = income_child$share_of_zero)
```

```
## # A tibble: 10 × 2
##    number_of_children share_of_zero
##                 <int>         <dbl>
##  1                  0       0.0149
##  2                  1       0.00785
##  3                  2       0.00574
##  4                  3       0.00803
##  5                  4       0
##  6                  5       0
##  7                  6       0
##  8                  7       0
##  9                  8       0
## 10                  9       0
```

The table shows that respondents with no children have the most share of zero income (maybe unemployed) people.

```
marry_income <- as.data.frame.matrix(table(dat97$CV_MARSTAT_COLLAPSED_2019, dat97$YINC_1700
_2019))
marry_income$share_of_zero <- (marry_income[,1])/rowSums(marry_income)

tibble(marital_status = c("Never_married", "Married", "Separated", "Divorced", "Widowed"),
 share_of_zero = marry_income$share_of_zero)
```

```
## # A tibble: 5 × 2
##   marital_status share_of_zero
##   <chr>                  <dbl>
## 1 Never_married        0.00565
## 2 Married              0.00745
## 3 Separated            0.0430
## 4 Divorced             0.00154
## 5 Widowed              0
```

The table shows that respondents with separated status have the largest share of people who don’t have income.

1.3.3

Concluding interpretation

1. The age of the respondents seems not to be associated with income.
2. Male appears to earn more than female.
3. People with small family size (i.e, 1-3 children) seems to earn more than people with no children and more than 3 children.
4. 35 & 38 years old age group has more people with 0 income than other groups in proportion.
5. There are more male with 0 income than female in proportion.

6. People with less children has a larger share of people with 0 income.

Exercise II

2.1

```
#Proposed model: income_i = b0+ b1 * work_experience_i + b2 * schooling_i

dat97 %>%
  filter(YINC_1700_2019 >0) %>%
  lm(YINC_1700_2019 ~ work_exp + schooling, data =.) %>%
  summary()
```

```
##
## Call:
## lm(formula = YINC_1700_2019 ~ work_exp + schooling, data = .)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -68169 -19413  -3518  18052  87171
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 11124.13    1290.65   8.619   <2e-16 ***
## work_exp     1071.26      66.28  16.163   <2e-16 ***
## schooling    2341.17      88.53  26.443   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 26080 on 5369 degrees of freedom
##   (4 observations deleted due to missingness)
## Multiple R-squared:  0.1621, Adjusted R-squared:  0.1617
## F-statistic: 519.2 on 2 and 5369 DF,  p-value: < 2.2e-16
```

This OLS model, however, may suffer from selection problem since people who report 0 income or those that don’t report income is not random. There might be ommitted variables that influence the reporting bias and also work experience and years of schooling.

Hence, we could consider Heckman’s two step estimation.

2.2

Heckman’s Two-Step estimator consider the selection problem by estimating the part that determines the dependent variable (here is income) but is not our explanatory variable. Using that part as a regressor for the second stage regression, we can obtain consistent estimates of x ruling out the effect of selection.

2.3

Hung-Wei said on Slack that we can use glm() to estimate the first stage probit

```
#create a dummy for income >0

dat97$nonmiss <- ifelse(dat97$YINC_1700_2019 > 0, 1,0)

dat97$nonmiss[is.na( dat97$nonmiss ) == T] = 0 #make missing value = 0

first <- glm(formula =  nonmiss ~ work_exp + schooling, family = binomial(link = "probit"),
data = dat97)
summary(first)
```

```
##
## Call:
## glm(formula = nonmiss ~ work_exp + schooling, family = binomial(link = "probit"),
##     data = dat97)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.9617   0.1301   0.4936   0.7566   1.4750
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.420733   0.051752   -8.13  4.3e-16 ***
## work_exp     0.113388   0.004697   24.14  < 2e-16 ***
## schooling    0.050049   0.003816   13.11  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 7404.4  on 6935  degrees of freedom
## Residual deviance: 6203.0  on 6933  degrees of freedom
##   (2048 observations deleted due to missingness)
## AIC: 6209
##
## Number of Fisher Scoring iterations: 7
```

```
first_predict <- - predict(first) #remember a negative sign

inmills <- dnorm(first_predict)/ (1- pnorm(first_predict)) #pdf/cdf
summary(inmills)
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
## 0.0000055 0.1771907 0.3704961 0.3848584 0.5527985 1.0836118
```

```
#Here is just second stage linear regression using MLE
Hecloglik <- function (par, work_exp, schooling, inmills)  {
  XB   = par[1] + par[2]* work_exp + par[3]* schooling + par[4] * inmills
  Prob = dnorm(XB)
  Prob[Prob>0.999999] = 0.999999
  Prob[Prob<0.000001] = 0.000001
  Like = log(Prob)
  return( - sum(Like) )
}
```

Use lm() to help me find good starting value

```
regdat <- dat97 %>%
  filter(is.na(work_exp) == F, is.na(schooling) == F) %>%
  cbind(inmills)

cheat <- lm(YINC_1700_2019 ~ work_exp + schooling + inmills, data = regdat)
as.vector(cheat$coefficients)
```

```
## [1]  44872.0617   -194.4803   1441.6851 -40057.2311
```

```r
startv <- as.vector(cheat$coefficients)
#noisestartv <- jitter(startv) #add noise
#noisestartv

work_exp = regdat$work_exp
schooling = regdat$schooling
inmills = regdat$inmills
```

```r
results_2stage <- optim(startv, fn = Hecloglik, method = "BFGS",
                  control = list(trace = 6, maxit = 3000),
                  work_exp = work_exp, schooling = schooling, inmills = inmills)
```

```
## initial  value 95824.381230
## final  value 95824.381230
## converged
```

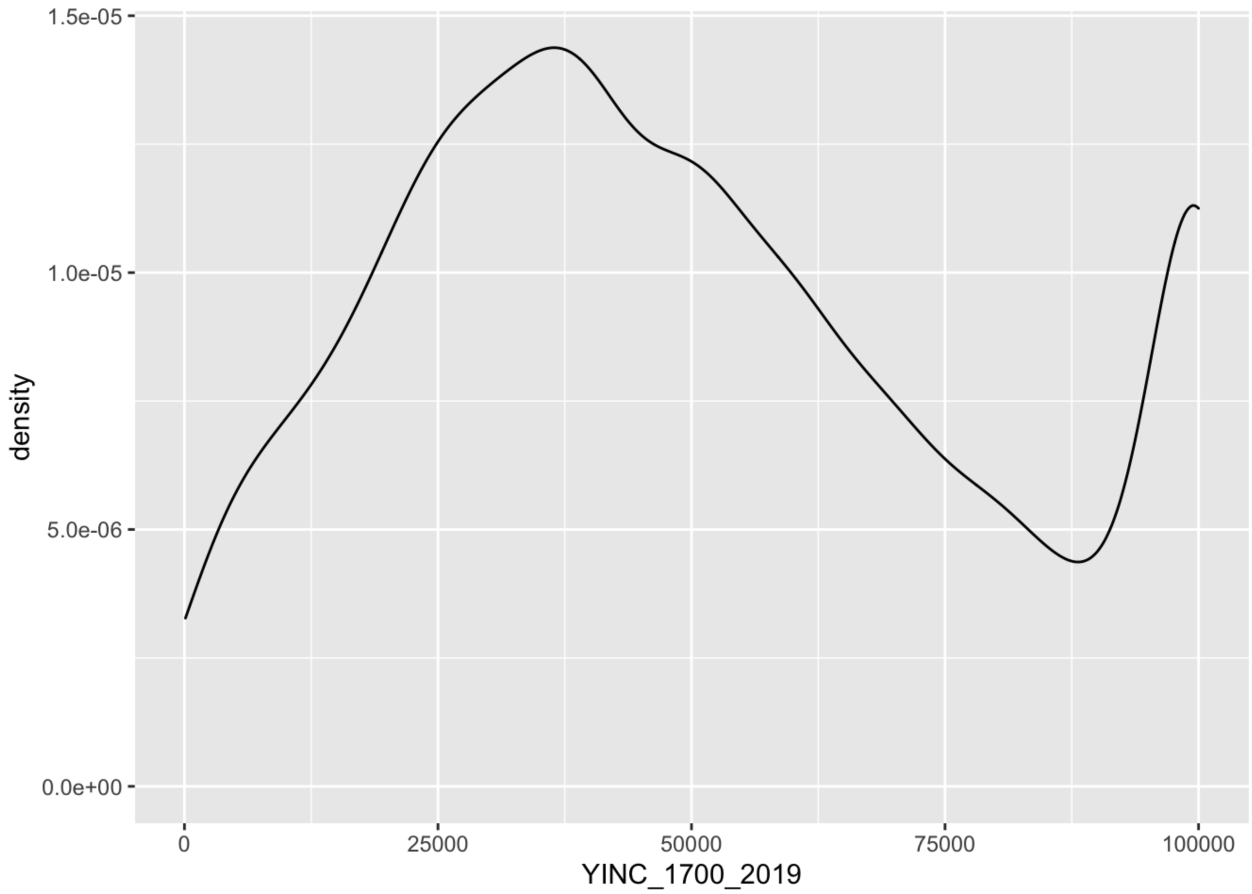
```r
results_2stage$par
```

```
## [1]  44872.0617   -194.4803   1441.6851 -40057.2311
```

The results change a lot. Perhaps there is ability (productivity) bias. For example, people with high capability receive more schooling and work longer while their talents grant them more wages. This could create overestimation using OLS. On the other hand, people with low ability could earn very little wage or unemployed, and they also could receive less schooling and work experience.

Exercise 3 3.1

```r
dat97%>%
  filter(YINC_1700_2019 > 0) %>%
  ggplot(aes(x = YINC_1700_2019)) + geom_density()
```




```
#Income should be top-coded at 100,000
```

The censored value is 100,000

3.2 & 3.3

I propose the two stage sample selection model to deal with censoring problem. I first explain top-coded incidents and then use the inverse mills ratio for the second stage estimation.

Since glm & lm are allowed for two-stage test, I use them again here.

```
#First stage: explaining top-coded income

dat97$topcode <- ifelse(dat97$YINC_1700_2019 == 100000, 1,0)

sum(is.na( dat97$topcode) ) #3572 NA values
```

```
## [1] 3572
```

```
datex3 <- dat97 %>%
  dplyr::select(topcode, YINC_1700_2019, work_exp, schooling) %>%
  filter(is.na(topcode) == F, is.na(work_exp) == F, is.na(schooling) == F)

#clear NA before going in estimation

topfirst <- glm(formula =  topcode ~ work_exp + schooling, family = binomial(link = "probi
t"), data = datex3)
summary(topfirst)
```

```
##
## Call:
## glm(formula = topcode ~ work_exp + schooling, family = binomial(link = "probit"),
##     data = datex3)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.1299  -0.5519  -0.4070  -0.3348   3.7012
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -3.111288   0.136645 -22.769  < 2e-16 ***
## work_exp     0.024135   0.004109   5.874 4.25e-09 ***
## schooling    0.121363   0.008858  13.700  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 3920.7  on 5407  degrees of freedom
## Residual deviance: 3599.5  on 5405  degrees of freedom
## AIC: 3605.5
##
## Number of Fisher Scoring iterations: 7
```

```
top_predict <- - predict(topfirst) #remember a negative sign

topmills <- dnorm(top_predict)/ (1- pnorm(top_predict)) #pdf/cdf
summary(topmills)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.8435  1.5174  1.7795  1.7928  1.9789  3.3867
```

```
datex3 <- cbind(datex3, topmills)
#2nd stage

topsec<- lm(YINC_1700_2019 ~ work_exp + schooling + topmills, data = datex3)
summary(topsec)
```

```
##
## Call:
## lm(formula = YINC_1700_2019 ~ work_exp + schooling + topmills,
##     data = datex3)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -96211 -18821  -3368  18082  73725
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -908772.6    65194.2  -13.94   <2e-16 ***
## work_exp       6538.2      392.6   16.65   <2e-16 ***
## schooling     31155.0     2044.1   15.24   <2e-16 ***
## topmills     275936.9    19561.1   14.11   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 25810 on 5404 degrees of freedom
## Multiple R-squared:  0.1908, Adjusted R-squared:  0.1904
## F-statistic: 424.8 on 3 and 5404 DF,  p-value: < 2.2e-16
```

3.4

```
#Results indicate that work experience and schooling are positively correlated with income
 as expected.

ols <-  lm(YINC_1700_2019 ~ work_exp + schooling, data = datex3)
summary(ols)
```

```
##
## Call:
## lm(formula = YINC_1700_2019 ~ work_exp + schooling, data = datex3)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -68126 -19407  -3514  18242  87576
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 10708.25    1294.51   8.272   <2e-16 ***
## work_exp     1077.63      66.57  16.189   <2e-16 ***
## schooling    2346.03      88.87  26.399   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 26280 on 5405 degrees of freedom
## Multiple R-squared:  0.161,  Adjusted R-squared:  0.1607
## F-statistic: 518.8 on 2 and 5405 DF,  p-value: < 2.2e-16
```

```
#The effect size is smaller when we ignore censoring issue. That is, we may underestimate t
he effect of education and work experience if we just use the censor data.
```

Exercise 4

Goal: the effect of education, marital status, experience and education on wages

```r
list.files()
```

```
## [1] "A4_files"                          "A4.html"
## [3] "A4.Rmd"                            "dat_A4_panel_variables_doc.pdf"
## [5] "dat_A4_panel.csv"                  "dat_A4_variables_doc.pdf"
## [7] "dat_A4.csv"                        "longp.rds"
```

```r
library(data.table)
library(tidyverse)
dat2 <- fread("dat_A4_panel.csv")


panel <- dat2 #just in case
```

4.1

The association between education and wage, marital status and wage, and experience and wage could have ability selection problem. People with better ability could be encouraged to study longer, and they could also earn more because of their productivity. Likewise, gifted people could do better in marriage market, and they could also work more years because of their ability.

4.2

Prepare the data for Between and Within Estimator

mutate mean income by individual, mutate mean independent variable by individual

```r
str_subset( colnames(panel), "YINC")
```

```
##  [1] "YINC-1700_1997" "YINC-1700_1998" "YINC-1700_1999" "YINC-1700_2000"
##  [5] "YINC-1700_2001" "YINC-1700_2002" "YINC-1700_2003" "YINC-1700_2004"
##  [9] "YINC-1700_2005" "YINC-1700_2006" "YINC-1700_2007" "YINC-1700_2008"
## [13] "YINC-1700_2009" "YINC-1700_2010" "YINC-1700_2011" "YINC-1700_2013"
## [17] "YINC-1700_2015" "YINC-1700_2017" "YINC-1700_2019"
```

```r
str_subset( colnames(panel), "DEGREE")
```

```
##  [1] "CV_HIGHEST_DEGREE_9899_1998"     "CV_HIGHEST_DEGREE_9900_1999"
##  [3] "CV_HIGHEST_DEGREE_0001_2000"     "CV_HIGHEST_DEGREE_0102_2001"
##  [5] "CV_HIGHEST_DEGREE_0203_2002"     "CV_HIGHEST_DEGREE_0304_2003"
##  [7] "CV_HIGHEST_DEGREE_0405_2004"     "CV_HIGHEST_DEGREE_0506_2005"
##  [9] "CV_HIGHEST_DEGREE_0607_2006"     "CV_HIGHEST_DEGREE_0708_2007"
## [11] "CV_HIGHEST_DEGREE_0809_2008"     "CV_HIGHEST_DEGREE_0910_2009"
## [13] "CV_HIGHEST_DEGREE_EVER_EDT_2010" "CV_HIGHEST_DEGREE_1011_2010"
## [15] "CV_HIGHEST_DEGREE_EVER_EDT_2011" "CV_HIGHEST_DEGREE_1112_2011"
## [17] "CV_HIGHEST_DEGREE_EVER_EDT_2013" "CV_HIGHEST_DEGREE_1314_2013"
## [19] "CV_HIGHEST_DEGREE_EVER_EDT_2015" "CV_HIGHEST_DEGREE_EVER_EDT_2017"
## [21] "CV_HIGHEST_DEGREE_EVER_EDT_2019"
```

```
panel <- panel %>%
  rowwise() %>%
  mutate(mincome = sum(c (`YINC-1700_1997` + `YINC-1700_1998` + `YINC-1700_1999` + `YINC-1700_2000` + `YINC-1700_2001` +
         `YINC-1700_2002`+ `YINC-1700_2003` + `YINC-1700_2004`+ `YINC-1700_2005`+ `YINC-1700_2006`+ `YINC-1700_2007`+
         `YINC-1700_2008`+ `YINC-1700_2009`+  `YINC-1700_2010`+ `YINC-1700_2011` + `YINC-1700_2013`+ `YINC-1700_2015`+
         `YINC-1700_2017`+ `YINC-1700_2019`), na.rm =T) /19 ) %>%
  ungroup() %>%
  mutate( across(starts_with("CV_HIGHEST_DEGREE"), recode, "0" = 0, "1" = 12, "2" = 12, "3" = 14, "4" = 16, "5" = 18, "6"=  20 , "7" = 20, "-1" = 0, "-2" = 0, "-3" = 0)) %>%   #here I assume years of education for "Invalid Skip" = 0
  select(- c (CV_HIGHEST_DEGREE_EVER_EDT_2010, CV_HIGHEST_DEGREE_EVER_EDT_2011, CV_HIGHEST_DEGREE_EVER_EDT_2013))
  #drop these two columns to remain consistent with 19 years 19 columns

#checking
panel %>%
  select(starts_with("CV_HIGHEST_DEGREE")) %>%
  glimpse()
```

```
## Rows: 8,984
## Columns: 18
## $ CV_HIGHEST_DEGREE_9899_1998     <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,…
## $ CV_HIGHEST_DEGREE_9900_1999     <dbl> 12, 0, 0, 12, 0, 0, 0, 12, 0, 0, 0, 0,…
## $ CV_HIGHEST_DEGREE_0001_2000     <dbl> 12, 12, 0, 12, 12, 12, 0, 12, 12, 0, 1…
## $ CV_HIGHEST_DEGREE_0102_2001     <dbl> 12, 12, 12, 12, 12, 12, 0, 12, 12, 0, …
## $ CV_HIGHEST_DEGREE_0203_2002     <dbl> 12, 12, 12, 12, 12, 12, 0, 12, 12, 12,…
## $ CV_HIGHEST_DEGREE_0304_2003     <dbl> 16, 12, 14, 12, 12, 12, NA, 16, 12, 12…
## $ CV_HIGHEST_DEGREE_0405_2004     <dbl> 16, 12, 14, 12, 12, 12, NA, 16, 12, 12…
## $ CV_HIGHEST_DEGREE_0506_2005     <dbl> 16, 12, NA, 12, 12, 12, 0, 16, 16, 12,…
## $ CV_HIGHEST_DEGREE_0607_2006     <dbl> 16, NA, NA, 12, 12, 12, 0, 16, 16, 16,…
## $ CV_HIGHEST_DEGREE_0708_2007     <dbl> 16, NA, NA, 12, 12, 12, NA, 16, 16, 16…
## $ CV_HIGHEST_DEGREE_0809_2008     <dbl> 16, 12, NA, 12, 12, 12, NA, NA, 16, 16…
## $ CV_HIGHEST_DEGREE_0910_2009     <dbl> 16, 12, 14, 12, 12, 12, 12, 18, 16, 16…
## $ CV_HIGHEST_DEGREE_1011_2010     <dbl> 16, 12, NA, 12, 12, 12, 12, NA, 16, NA…
## $ CV_HIGHEST_DEGREE_1112_2011     <dbl> 16, 12, 14, 12, 12, 12, 12, 18, 18, NA…
## $ CV_HIGHEST_DEGREE_1314_2013     <dbl> 16, 12, 14, 12, 12, 12, NA, NA, 18, NA…
## $ CV_HIGHEST_DEGREE_EVER_EDT_2015 <dbl> 16, 12, NA, 12, 12, 12, 12, 18, 18, NA…
## $ CV_HIGHEST_DEGREE_EVER_EDT_2017 <dbl> NA, 12, 16, 12, 12, 12, NA, NA, 18, 16…
## $ CV_HIGHEST_DEGREE_EVER_EDT_2019 <dbl> NA, 12, 16, 12, 12, 12, 12, 18, 18, 16…
```

```
panel$mschool <- panel %>%
  select(starts_with("CV_HIGHEST_DEGREE")) %>%
  rowMeans(na.rm = T)

#Now to deal with marital status

p2 <- panel #just in case

p2 <- p2 %>%
  mutate( across(starts_with("CV_MARSTAT"), recode, "0" = 0, "1" = 1, "2" = 0, "3" = 0, "4"
= 0, "-1" = 0, "-2" = 0))
#Here I treat Separated Divorced and Widowed as not married, along side "never married"

p2$mmar <- p2 %>%
  select(starts_with("CV_MARSTAT")) %>%
  rowMeans(na.rm = T)

# Finally, I have to deal with work experience

p2$work_exp_1997 <- p2 %>% select(starts_with("CV_WKSWK_JOB") & ends_with("1997")) %>% rowS
ums(na.rm = T)/52.1429
p2$work_exp_1998 <- p2 %>% select(starts_with("CV_WKSWK_JOB") & ends_with("1998")) %>% rowS
ums(na.rm = T)/52.1429
p2$work_exp_1999 <- p2 %>% select(starts_with("CV_WKSWK_JOB") & ends_with("1999")) %>% rowS
ums(na.rm = T)/52.1429
p2$work_exp_2000 <- p2 %>% select(starts_with("CV_WKSWK_JOB") & ends_with("2000")) %>% rowS
ums(na.rm = T)/52.1429
p2$work_exp_2001 <- p2 %>% select(starts_with("CV_WKSWK_JOB") & ends_with("2001")) %>% rowS
ums(na.rm = T)/52.1429

p2$work_exp_2002 <- p2 %>% select(starts_with("CV_WKSWK_JOB") & ends_with("2002")) %>% rowS
ums(na.rm = T)/52.1429
p2$work_exp_2003 <- p2 %>% select(starts_with("CV_WKSWK_JOB") & ends_with("2003")) %>% rowS
ums(na.rm = T)/52.1429
p2$work_exp_2004 <- p2 %>% select(starts_with("CV_WKSWK_JOB") & ends_with("2004")) %>% rowS
ums(na.rm = T)/52.1429
p2$work_exp_2005 <- p2 %>% select(starts_with("CV_WKSWK_JOB") & ends_with("2005")) %>% rowS
ums(na.rm = T)/52.1429
p2$work_exp_2006 <- p2 %>% select(starts_with("CV_WKSWK_JOB") & ends_with("2006")) %>% rowS
ums(na.rm = T)/52.1429

p2$work_exp_2007 <- p2 %>% select(starts_with("CV_WKSWK_JOB") & ends_with("2007")) %>% rowS
ums(na.rm = T)/52.1429
p2$work_exp_2008 <- p2 %>% select(starts_with("CV_WKSWK_JOB") & ends_with("2008")) %>% rowS
ums(na.rm = T)/52.1429
p2$work_exp_2009 <- p2 %>% select(starts_with("CV_WKSWK_JOB") & ends_with("2009")) %>% rowS
ums(na.rm = T)/52.1429
p2$work_exp_2010 <- p2 %>% select(starts_with("CV_WKSWK_JOB") & ends_with("2010")) %>% rowS
ums(na.rm = T)/52.1429
p2$work_exp_2011 <- p2 %>% select(starts_with("CV_WKSWK_JOB") & ends_with("2011")) %>% rowS
ums(na.rm = T)/52.1429

p2$work_exp_2013 <- p2 %>% select(starts_with("CV_WKSWK_JOB") & ends_with("2013")) %>% rowS
ums(na.rm = T)/52.1429
p2$work_exp_2015 <- p2 %>% select(starts_with("CV_WKSWK_JOB") & ends_with("2015")) %>% rowS
ums(na.rm = T)/52.1429
p2$work_exp_2017 <- p2 %>% select(starts_with("CV_WKSWK_JOB") & ends_with("2017")) %>% rowS
ums(na.rm = T)/52.1429
p2$work_exp_2019 <- p2 %>% select(starts_with("CV_WKSWK_JOB") & ends_with("2019")) %>% rowS
```

```
ums(na.rm = T)/52.1429

str_which( colnames(p2), "work_exp" )
```

```
##  [1] 250 251 252 253 254 255 256 257 258 259 260 261 262 263 264 265 266 267 268
```

```
p2$mexp <- rowMeans(p2[, 250:268], na.rm = T) #column 252:270 are work_exp_1997 - work_exp_
2019
```

Between Estimator

Note that I first deal with between estimator since I already got the means

```
between <- p2 %>%
  select(mincome, mschool, mmar, mexp) %>%
  na.omit() %>%
  lm(mincome ~ mmar + mexp + mschool, data =.)

summary(between)
```

```
##
## Call:
## lm(formula = mincome ~ mmar + mexp + mschool, data = .)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##  -8069  -1086   -463     87  94219
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1060.58     143.09  -7.412 1.36e-13 ***
## mmar          767.22     206.61   3.713 0.000206 ***
## mexp          481.70      34.08  14.135  < 2e-16 ***
## mschool        52.66      14.55   3.619 0.000297 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5085 on 8867 degrees of freedom
## Multiple R-squared:  0.03479,    Adjusted R-squared:  0.03446
## F-statistic: 106.5 on 3 and 8867 DF,  p-value: < 2.2e-16
```

Preparing data for within estimator idea: yit - mean(yi) = beta * (xit - mean(xi))

```
str_subset( colnames(p2), "work_exp")
```

```
##  [1] "work_exp_1997" "work_exp_1998" "work_exp_1999" "work_exp_2000"
##  [5] "work_exp_2001" "work_exp_2002" "work_exp_2003" "work_exp_2004"
##  [9] "work_exp_2005" "work_exp_2006" "work_exp_2007" "work_exp_2008"
## [13] "work_exp_2009" "work_exp_2010" "work_exp_2011" "work_exp_2013"
## [17] "work_exp_2015" "work_exp_2017" "work_exp_2019"
```

```
str_subset( colnames(p2), "MAR")
```

```
##  [1] "CV_MARSTAT_COLLAPSED_1997" "CV_MARSTAT_COLLAPSED_1998"
##  [3] "CV_MARSTAT_COLLAPSED_1999" "CV_MARSTAT_COLLAPSED_2000"
##  [5] "CV_MARSTAT_COLLAPSED_2001" "CV_MARSTAT_COLLAPSED_2002"
##  [7] "CV_MARSTAT_COLLAPSED_2003" "CV_MARSTAT_COLLAPSED_2004"
##  [9] "CV_MARSTAT_COLLAPSED_2005" "CV_MARSTAT_COLLAPSED_2006"
## [11] "CV_MARSTAT_COLLAPSED_2007" "CV_MARSTAT_COLLAPSED_2008"
## [13] "CV_MARSTAT_COLLAPSED_2009" "CV_MARSTAT_COLLAPSED_2010"
## [15] "CV_MARSTAT_COLLAPSED_2011" "CV_MARSTAT_COLLAPSED_2013"
## [17] "CV_MARSTAT_COLLAPSED_2015" "CV_MARSTAT_COLLAPSED_2017"
## [19] "CV_MARSTAT_COLLAPSED_2019"
```

```
str_subset( colnames(p2), "DEGREE") #note that we only have 18 rounds of education informat
ion (no such data for 1997)
```

```
##  [1] "CV_HIGHEST_DEGREE_9899_1998"     "CV_HIGHEST_DEGREE_9900_1999"
##  [3] "CV_HIGHEST_DEGREE_0001_2000"     "CV_HIGHEST_DEGREE_0102_2001"
##  [5] "CV_HIGHEST_DEGREE_0203_2002"     "CV_HIGHEST_DEGREE_0304_2003"
##  [7] "CV_HIGHEST_DEGREE_0405_2004"     "CV_HIGHEST_DEGREE_0506_2005"
##  [9] "CV_HIGHEST_DEGREE_0607_2006"     "CV_HIGHEST_DEGREE_0708_2007"
## [11] "CV_HIGHEST_DEGREE_0809_2008"     "CV_HIGHEST_DEGREE_0910_2009"
## [13] "CV_HIGHEST_DEGREE_1011_2010"     "CV_HIGHEST_DEGREE_1112_2011"
## [15] "CV_HIGHEST_DEGREE_1314_2013"     "CV_HIGHEST_DEGREE_EVER_EDT_2015"
## [17] "CV_HIGHEST_DEGREE_EVER_EDT_2017" "CV_HIGHEST_DEGREE_EVER_EDT_2019"
```

```
p3 <- p2 #just in case

p3 <- rename(p3, DEGREE_2015 = CV_HIGHEST_DEGREE_EVER_EDT_2015, DEGREE_2017 = CV_HIGHEST_DE
GREE_EVER_EDT_2017, DEGREE_2019 = CV_HIGHEST_DEGREE_EVER_EDT_2019)

p3 <- rename(p3, DEGREE_1998 = CV_HIGHEST_DEGREE_9899_1998, DEGREE_1999 = CV_HIGHEST_DEGREE
_9900_1999,
             DEGREE_2000 = CV_HIGHEST_DEGREE_0001_2000, DEGREE_2001 = CV_HIGHEST_DEGREE_010
2_2001,
             DEGREE_2002 = CV_HIGHEST_DEGREE_0203_2002, DEGREE_2003 = CV_HIGHEST_DEGREE_030
4_2003,
             DEGREE_2004 = CV_HIGHEST_DEGREE_0405_2004, DEGREE_2005 = CV_HIGHEST_DEGREE_050
6_2005,
             DEGREE_2006 = CV_HIGHEST_DEGREE_0607_2006, DEGREE_2007 = CV_HIGHEST_DEGREE_070
8_2007,
             DEGREE_2008 = CV_HIGHEST_DEGREE_0809_2008, DEGREE_2009 = CV_HIGHEST_DEGREE_091
0_2009,
             DEGREE_2010 = CV_HIGHEST_DEGREE_1011_2010, DEGREE_2011 = CV_HIGHEST_DEGREE_111
2_2011,
             DEGREE_2013 = CV_HIGHEST_DEGREE_1314_2013)

library(panelr)
```

```
## Loading required package: lme4
```

```
## Loading required package: Matrix
```

```
##
## Attaching package: 'Matrix'
```

```
## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack
```

```
##
## Attaching package: 'panelr'
```

```
## The following object is masked from 'package:stats':
##
##     filter
```

```
str_which( colnames(p3), "DEGREE")
```

```
##  [1]  18  30  42  54  65  78  91 101 113 125 136 147 159 171 187 200 215 233
```

```
str_which( colnames(p3), "MAR")
```

```
##  [1]   7  19  31  43  55  66  79  92 102 114 126 137 148 160 172 188 201 216 234
```

```
var_list = c(1,2, 247:249, 269, 250:268, str_which( colnames(p3), "YINC"), str_which( colna
mes(p3), "MAR"),str_which( colnames(p3), "DEGREE"))

try <- p3[,var_list]
try <- as.data.frame(try)

longp <- long_panel(try, label_location = "end", prefix = "_", periods = c(1997:2011, 2013,
2015, 2017, 2019))

saveRDS(longp, file = "longp.rds")


#In mutate, NA - some value will produce NA. That is fine for this analysis.

longp <- longp %>%
  mutate(income_diff = `YINC-1700` - mincome, mar_diff = CV_MARSTAT_COLLAPSED - mmar,
         exp_dif = work_exp - mexp, sch_dif = DEGREE - mschool)
```

Within Estimator

```
modwithin <- longp %>%
  select(income_diff, mar_diff, exp_dif, sch_dif) %>%
  na.omit() %>%
  lm(income_diff ~ mar_diff + exp_dif + sch_dif -1, data =.)

#Note that I did not include intercept because the intercept is subtracted by construction

summary(modwithin)
```

```
##
## Call:
## lm(formula = income_diff ~ mar_diff + exp_dif + sch_dif - 1,
##     data = .)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -108233    3199   14973   29002  327057
##
## Coefficients:
##          Estimate Std. Error t value Pr(>|t|)
## mar_diff 12981.07     344.88   37.64   <2e-16 ***
## exp_dif   4274.28      39.14  109.19   <2e-16 ***
## sch_dif   2739.19      32.68   83.83   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 32250 on 81959 degrees of freedom
## Multiple R-squared:  0.3064, Adjusted R-squared:  0.3063
## F-statistic: 1.207e+04 on 3 and 81959 DF,  p-value: < 2.2e-16
```

(First) Difference

Use long data group by ID, lag value

Note that my first difference method is using yit minus its previous period. For a person i's data in 2009, I am taking the difference of 2009 and 2008.

```
longp <- longp %>%
  group_by(id) %>%
  mutate(lincome = dplyr::lag(`YINC-1700`, n = 1), lmar = dplyr::lag(CV_MARSTAT_COLLAPSED,
 n = 1),
         lexp = dplyr::lag(work_exp, n =1), lsch = dplyr::lag(DEGREE, n = 1)) %>%
  mutate(fdincome = `YINC-1700` - lincome, fdmar = CV_MARSTAT_COLLAPSED - lmar,
         fdexp = work_exp - lexp, fdsch = DEGREE - lsch)

modFD <- longp %>%
  select(fdincome, fdmar, fdexp, fdsch) %>%
  na.omit() %>%
  lm(fdincome ~ fdmar + fdexp + fdsch -1 , data =.)

#I also exclude the intercept since the intercept is subtracted with the first difference m
ethod
summary(modFD)
```

```
##
## Call:
## lm(formula = fdincome ~ fdmar + fdexp + fdsch - 1, data = .)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -206894   -2007    1448    7339  325680
##
## Coefficients:
##       Estimate Std. Error t value Pr(>|t|)
## fdmar  2909.31     266.83  10.903  < 2e-16 ***
## fdexp   972.07      34.70  28.016  < 2e-16 ***
## fdsch   187.16      28.25   6.624 3.52e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 16610 on 58533 degrees of freedom
## Multiple R-squared:  0.01657,    Adjusted R-squared:  0.01651
## F-statistic: 328.6 on 3 and 58533 DF,  p-value: < 2.2e-16
```

4.3

```
coef <- list( First_difference = modFD$coefficients, Within = modwithin$coefficients, Betwe
en = between$coefficients)
coef
```

```
## $First_difference
##     fdmar     fdexp     fdsch
## 2909.3115  972.0722  187.1596
##
## $Within
##  mar_diff   exp_dif   sch_dif
## 12981.075  4274.285  2739.193
##
## $Between
## (Intercept)        mmar        mexp     mschool
## -1060.57519   767.21591   481.70324    52.66101
```

It is evident that my three models produce very different results. However, the coefficients for three models are all positive and significant. At least we can be certain about the positive relationship between our independent variables and income.

The difference in coefficients might very well be how the models deal with NA. Because I did not drop “every” rows with NA value, the three approaches should have different observations going into the regressions. For example, the between estimator could have the most observations since I compute the mean regardless of the presence of NA in a row (I just skip that one with na.rm = T).

On the other hand, the first difference approach certainly won’t use the observations in the first year (1997) while the within estimator and between estimator (incorporate in the mean value) will use them.