#ECON613 A2 Mengzhi Chen

#==========

```
#Set working directory
setwd("/Users/halcyonchan/Desktop/Econ613/A2/Data")
install.packages("data.table")
library(data.table)
```

#Exercise1==========

#1.1==========

```
#Calculate the correlation between Y and X
#Read the file
datind2009 = fread("datind2009.csv")
install.packages("tidyverse")
library(tidyverse)
dat = datind2009 %>% drop_na(wage)
x = matrix(dat$age)
#Add one column of intercept
intercept = matrix(rep(1,20232))
X = cbind(intercept,x)
Y = matrix(dat$wage)
#Calculate the correlation
cor(x,Y,method=c("pearson","kendall","spearman"),use="complete.obs")
#The correlation coefficient is -0.1788512
```

```
> cor(x,Y,method=c("pearson","kendall","spearman"),use="complete.obs")
           [,1]
[1,] -0.1788512
```

#1.2==========

```
#Calculate the coefficients on this regression
beta = solve(t(X)%*%X)%*%t(X)%*%Y
#the coefficient of X: beta1 is -180.1765, and the intercept beta0 is 22075.1066
```

```
> beta = solve(t(X)%*%X)%*%t(X)%*%Y
> beta
           [,1]
[1,] 22075.1066
[2,]  -180.1765
```

#1.3==========

```
#Calculate the standard errors of beta
#Using the standard formulas of the OLS
Y_hat = X%*%beta
e = (Y-Y_hat)
s2 = t(e)%*%e/(length(Y)-2)
XX = solve(t(X)%*%X)
SE1 = sqrt(s2*XX[1,1]) #The standard error of coefficient of intercept is 357.8275
SE2 = sqrt(s2*XX[2,2]) #The standard error of coefficient of age is 6.968652
```

```
> SE1
         [,1]
[1,] 357.8275
> SE2
         [,1]
[1,] 6.968652
```

```r
#Using bootstrap with 49 replications
SE1 = matrix(1:49)
SE2 = matrix(1:49)
for (i in 1:49) {
  deter1 = sample(1:length(x),length(x),replace=TRUE) #sample the number of data in original data
  dat_boot = dat[deter1,] #Choose the corresponding number of data
  independ = matrix(dat_boot$age) #Choose the corresponding independent variable
  intercept = matrix(rep(1,length(independ)))
  independ1 = cbind(intercept,independ)
  depend = matrix(dat_boot$wage) #Choose the corresponding dependent variable
  coef = solve(t(independ1)%*%independ1)%*%t(independ1)%*%depend #Use formula to compute the coefficients
  #Calculate the standard error of intercept and coefficient
  depend_hat = independ1%*%coef
  e = (depend-depend_hat)
  s2 = t(e)%*%e/(length(Y)-2)
  XX = solve(t(X)%*%X)
  SE1[i] = sqrt(s2*XX[1,1])
  SE2[i] = sqrt(s2*XX[2,2])
}
SE1_boot49 = mean(SE1)
SE1_boot49 #The standard error of coefficient of intercept is 356.6284
SE2_boot49 = mean(SE2)
SE2_boot49 #The standard error of coefficient of age is 6.9453
```

```
> SE1_boot49 = mean(SE1)
> SE1_boot49 #The standard error of coefficient of intercept is 356.6284
[1] 356.6284
> SE2_boot49 = mean(SE2)
> SE2_boot49 #The standard error of coefficient of age is 6.9453
[1] 6.9453
```

```r
#Using bootstrap with 499 replications. The model is the same as above.
SE1 = matrix(1:499)
SE2 = matrix(1:499)
for (i in 1:499) {
  deter1 = sample(1:length(x),length(x),replace=TRUE)
  dat_boot = dat[deter1,]
  independ = matrix(dat_boot$age)
  intercept = matrix(rep(1,length(independ)))
  independ1 = cbind(intercept,independ)
  depend = matrix(dat_boot$wage)
  coef = solve(t(independ1)%*%independ1)%*%t(independ1)%*%depend
  depend_hat = independ1%*%coef
  e = (depend-depend_hat)
  s2 = t(e)%*%e/(length(Y)-2)
  XX = solve(t(X)%*%X)
  SE1[i] = sqrt(s2*XX[1,1])
  SE2[i] = sqrt(s2*XX[2,2])
}
SE1_boot49 = mean(SE1)
SE1_boot49 #The standard error of coefficient of intercept is 357.8577
SE2_boot49 = mean(SE2)
SE2_boot49 #The standard error of coefficient of age is 6.96924
```

```
> SE1_boot49 = mean(SE1)
> SE1_boot49 #The standard error of coefficient of intercept is 357.8577
[1] 357.8577
> SE2_boot49 = mean(SE2)
> SE2_boot49 #The standard error of coefficient of age is 6.96924
[1] 6.96924
```

Comment on the difference between the two strategies:

The standard error generated by bootstrap with 499 replications is more precise than that with 49 replications, because by repeating more times, the result will be more and more closer to the standard error calculated by OLS formulas.

#Exercise2==========

```
#read the data from 2005 to 2018
datind2005 = fread("datind2005.csv",colClasses=c(idmen = "character",idind = "character"))
datind2006 = fread("datind2006.csv",colClasses=c(idmen = "character",idind = "character"))
datind2007 = fread("datind2007.csv",colClasses=c(idmen = "character",idind = "character"))
datind2008 = fread("datind2008.csv",colClasses=c(idmen = "character",idind = "character"))
datind2009 = fread("datind2009.csv",colClasses=c(idmen = "character",idind = "character"))
datind2010 = fread("datind2010.csv",colClasses=c(idmen = "character",idind = "character"))
datind2011 = fread("datind2011.csv",colClasses=c(idmen = "character",idind = "character"))
datind2012 = fread("datind2012.csv",colClasses=c(idmen = "character",idind = "character"))
datind2013 = fread("datind2013.csv",colClasses=c(idmen = "character",idind = "character"))
datind2014 = fread("datind2014.csv",colClasses=c(idmen = "character",idind = "character"))
datind2015 = fread("datind2015.csv",colClasses=c(idmen = "character",idind = "character"))
datind2016 = fread("datind2016.csv",colClasses=c(idmen = "character",idind = "character"))
datind2017 = fread("datind2017.csv",colClasses=c(idmen = "character",idind = "character"))
datind2018 = fread("datind2018.csv",colClasses=c(idmen = "character",idind = "character"))
# Use rbind() to append all these datasets
datind=rbind(datind2005,datind2006,datind2007,datind2008,datind2009,datind2010,datind2011,
             datind2012,datind2013,datind2014,datind2015,datind2016,datind2017,datind2018)
```

#2.1==========

```
#Create a categorical variable ag, which bins the age variables into the following groups:
#“18-25”, “26- 30”, “31-35”, “36-40”,“41-45”, “46-50”,“51-55”, “56-60”, and “60+”.
datind[which(datind[,9]>=18 & datind[,9]<=25),"ag"] = "18-25"
datind[which(datind[,9]>=26 & datind[,9]<=30),"ag"] = "26-30"
datind[which(datind[,9]>=31 & datind[,9]<=35),"ag"] = "31-35"
datind[which(datind[,9]>=36 & datind[,9]<=40),"ag"] = "36-40"
datind[which(datind[,9]>=41 & datind[,9]<=45),"ag"] = "41-45"
datind[which(datind[,9]>=46 & datind[,9]<=50),"ag"] = "46-50"
datind[which(datind[,9]>=51 & datind[,9]<=55),"ag"] = "51-55"
datind[which(datind[,9]>=56 & datind[,9]<=60),"ag"] = "56-60"
datind[which(datind[,9]>60),"ag"] = "60+"
datind
```

```
> datind
           V1               idind            idmen year  empstat respondent profession gender age  wage    ag
     1:     1 1120001004058010001 1200010040580100 2005 Inactive          1            Female  31 12334 31-35
     2:     3 1120001006663010001 1200010066630100 2005 Employed          1         38   Male  32 50659 31-35
     3:     4 1120001006663010002 1200010066630100 2005 Employed          0         45 Female  28 19231 26-30
     4:     5 1120001008245010001 1200010082450100 2005  Retired          1            Female  90     0   60+
     5:     6 1120001008644010001 1200010086440100 2005 Employed          1         34   Male  37 31511 36-40
    ---
281854: 24693 1321065801707010001 3210658017070100 2018 Employed          1         22   Male  60 19467 56-60
281855: 24694 1321065801707010002 3210658017070100 2018  Retired          0       <NA> Female  64     0   60+
281856: 24695 1321065810196010001 3210658101960100 2018 Employed          1         35   Male  65     0   60+
281857: 24696 1321065812667010001 3210658126670100 2018 Inactive          1       <NA> Female  90     0   60+
281858: 24697 1321065812667010002 3210658126670100 2018 Employed          0         54 Female  55  9343 51-55
```

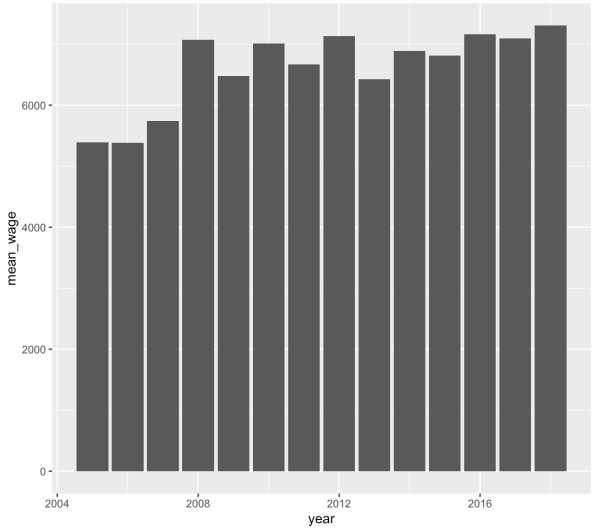
#2.2==========

```
#Plot the wage of each age group across years. Is there a trend?
datind = datind %>% drop_na(ag)
datind = datind %>% drop_na(wage)
#Group the data according to year and age groups,and calculate the mean wage of each age group across years
datind_ag = aggregate(x=datind$wage,by=list(datind$year,datind$ag),FUN=mean)
#rename the columns
colnames(datind_ag)[1]="year"
colnames(datind_ag)[2]="age"
colnames(datind_ag)[3]="mean_wage"
```

```
> datind_ag
    year   age  mean_wage
1   2005 18-25  5388.2935
2   2006 18-25  5383.7597
3   2007 18-25  5743.0889
4   2008 18-25  7071.3218
5   2009 18-25  6480.7255
6   2010 18-25  7010.6360
7   2011 18-25  6665.5854
8   2012 18-25  7133.4034
9   2013 18-25  6422.2698
10  2014 18-25  6889.6798
```

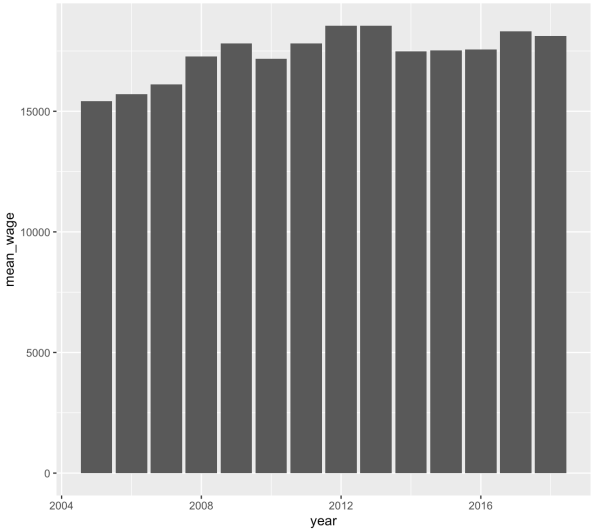
#Plot the the wage of age group 18-25 across years

```
dat_ag1 = datind_ag[which(datind_ag[2]=="18-25"),]
ggplot(select(dat_ag1,year,mean_wage),aes(x=year,y=mean_wage))+geom_bar(stat = "identity")
```



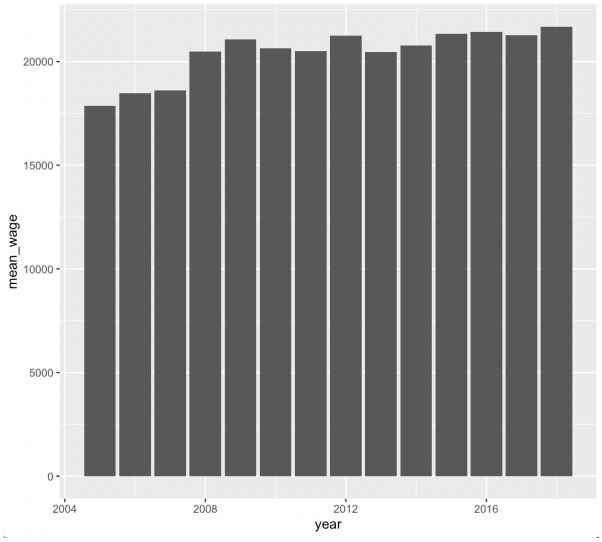
#Plot the the wage of age group 26-30 across years

```
dat_ag2 = datind_ag[which(datind_ag[2]=="26-30"),]
ggplot(select(dat_ag2,year,mean_wage),aes(x=year,y=mean_wage))+geom_bar(stat = "identity")
```



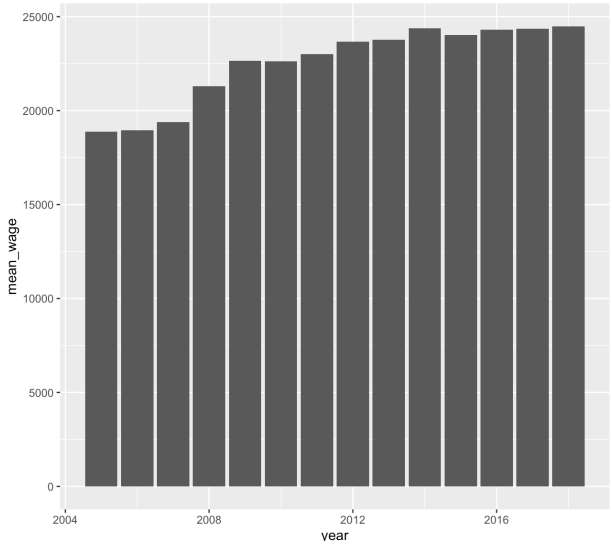
#Plot the the wage of age group 31-35 across years

```
dat_ag3 = datind_ag[which(datind_ag[2]=="31-35"),]
ggplot(select(dat_ag3,year,mean_wage),aes(x=year,y=mean_wage))+geom_bar(stat = "identity")
```



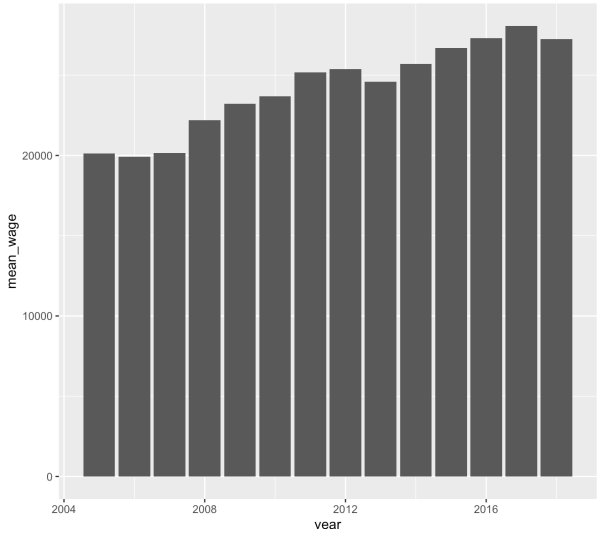
#Plot the the wage of age group 36-40 across years

```
dat_ag3 = datind_ag[which(datind_ag[2]=="36-40"),]
ggplot(select(dat_ag3,year,mean_wage),aes(x=year,y=mean_wage))+geom_bar(stat = "identity")
```



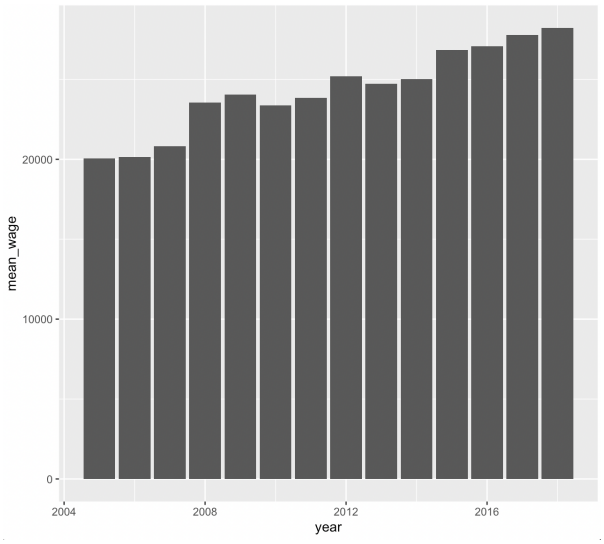
#Plot the the wage of age group 41-45 across years

```
dat_ag4 = datind_ag[which(datind_ag[2]=="41-45"),]
ggplot(select(dat_ag4,year,mean_wage),aes(x=year,y=mean_wage))+geom_bar(stat = "identity")
```



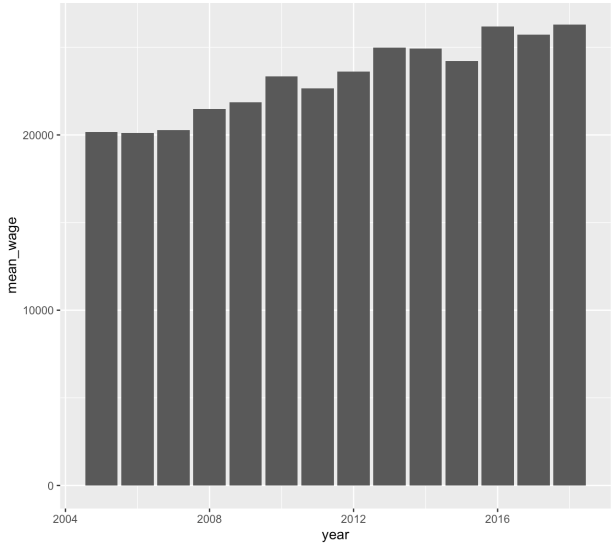
#Plot the the wage of age group 46-50 across years

```
dat_ag5 = datind_ag[which(datind_ag[2]=="46-50"),]
ggplot(select(dat_ag5,year,mean_wage),aes(x=year,y=mean_wage))+geom_bar(stat = "identity")
```



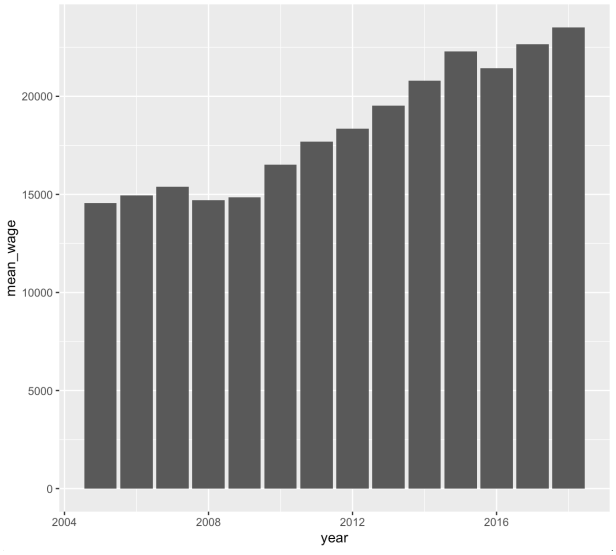
#Plot the the wage of age group 51-55 across years

```
dat_ag6 = datind_ag[which(datind_ag[2]=="51-55"),]
ggplot(select(dat_ag6,year,mean_wage),aes(x=year,y=mean_wage))+geom_bar(stat = "identity")
```



#Plot the the wage of age group 56-60 across years

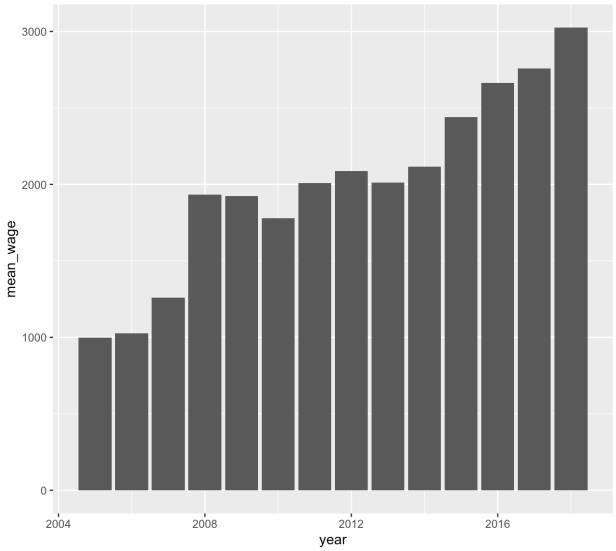
```
dat_ag7 = datind_ag[which(datind_ag[2]=="56-60"),]
ggplot(select(dat_ag7,year,mean_wage),aes(x=year,y=mean_wage))+geom_bar(stat = "identity")\
```



#Plot the the wage of age group 60+ across years

```
dat_ag8 = datind_ag[which(datind_ag[2]=="60+"),]
```

ggplot(select(dat_ag8,year,mean_wage),aes(x=year,y=mean_wage))+geom_bar(stat = "identity")



Trend: The salary of each age group increases through years, and the salary increases first and then decreases with age.

#2.3==========

```
> #Consider Yit = βXit + γt + eit. After including a time fixed effect, how do the estimated coefficients change?
> #install.packages("plm")
> #library(plm)
> datind=rbind(datind2005,datind2006,datind2007,datind2008,datind2009,datind2010,datind2011,
+              datind2012,datind2013,datind2014,datind2015,datind2016,datind2017,datind2018)
> #The regression without fixed effect
> reg = lm(datind$wage~datind$age,datind)
> reg$coefficients[2]
datind$age
 -182.4896
> #Use plm and include a time fixed effect
> reg1 = plm(datind$wage~datind$age,datind,index=c("year"),model="within")
> reg1$coefficients[1]
datind$age
 -186.8793
> #The estimated coefficient is -186.8793, which decreases from -182.4896
```

#Exercise3==========

#We are interested in the effect of age on labor market participation.

#We consider this problem using the data from 2007. Consider a probit model.

#3.1==========

```
#Exclude all individuals who are inactive
datind2007 = fread("datind2007.csv")
#Use which to exclude "inactive" individuals
dat2 = datind2007[which(datind2007$empstat!="Inactive"),]
```

```
> dat2
          V1               idind            idmen year    empstat respondent profession gender age  wage
    1:     1 1140001000124010001 1400010001240100 2007 Unemployed          1         NA   Male  49     0
    2:     2 1140001000124010002 1400010001240100 2007   Employed          0         52 Female  49 22744
    3:     4 1140001001167010001 1400010011670100 2007   Employed          1         21   Male  40  1243
    4:     8 1140001002054010001 1400010020540100 2007   Employed          1         22   Male  57     0
    5:     9 1140001002054010002 1400010020540100 2007 Unemployed          0         NA Female  54     0
   ---
16635: 25901 2210955706315010002 2109557063150100 2007   Employed          0         43   Male  31 20088
16636: 25902 2210955711575010101 2109557115750101 2007   Employed          1         67 Female  48 24594
16637: 25903 2210955711575010102 2109557115750101 2007   Employed          0         47   Male  49 30390
16638: 25905 2211018011864010101 2110180118640101 2007   Employed          1         62   Male  47 15048
16639: 25906 2211030812404010002 2110308124040100 2007   Employed          0         67   Male  24 22726
```

#3.2==========

```
> #Write a function that returns the likelihood of the probit of being employed
> #You might want to write Xβ first. Then, calculate F(Xβ) and the log likelihood
> #Remember, for the probit model, F(x) is the standard normal distribution function
> probit = function(beta,x,y) {
+   xbeta = beta[1] + beta[2]*x
+   pr = pnorm(xbeta)
+   pr[pr>0.999999] = 0.999999
+   pr[pr<0.000001] = 0.000001
+   like = y*log(pr) + (1-y)*log(1-pr)
+   return(-sum(like,log=TRUE))
+ }
> #Set the dependent and independent variables
> dat2 = dat2 %>% mutate(employed = ifelse(dat2$empstat == "Employed",1,0))
> #Use probit regression in R
> reg2 = glm(dat2$employed~dat2$age,family = binomial(link = "probit"))
> summary(reg2)

Call:
glm(formula = dat2$employed ~ dat2$age, family = binomial(link = "probit"))

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.4654  -0.5284   0.2938   0.7033   2.5822

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  3.8291873  0.0505722   75.72   <2e-16 ***
dat2$age    -0.0678642  0.0009246  -73.40   <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: 21944  on 16638  degrees of freedom
Residual deviance: 13164  on 16637  degrees of freedom
AIC: 13168

Number of Fisher Scoring iterations: 6

> #Test whether the function is correct
> test_beta = reg2$coefficients
> test_beta #3.8291873(intercept)  -0.0678642(a)
(Intercept)    dat2$age
  3.8291873  -0.0678642
> #Get the same result
> probit(test_beta,dat2$age,dat2$employed) #6581.155
[1] 6581.155
> logLik(reg2) #-6582.155 (df=2)
'log Lik.' -6582.155 (df=2)
```

#3.3==========

```
> #Optimize the model and interpret the coefficients. You can use pre-programmed optimization packages
> start1 = runif(2,min=-0.07,max=3.9)
> probit(start1,dat2$age,dat2$employed)
[1] 10029.71
> opt1 = optim(start1,fn=probit,method="BFGS",
+              control=list(trace=6,REPORT=1,maxit=1000),x=dat2$age,y=dat2$employed,hessian=TRUE)
initial  value 10029.709505
iter   2 value 8759.925215
iter   3 value 7241.731424
iter   4 value 6615.238297
iter   5 value 6582.066130
iter   6 value 6581.202876
iter   7 value 6581.202719
iter   8 value 6581.155461
iter   8 value 6581.155434
iter   8 value 6581.155434
final  value 6581.155434
converged
> opt1$par
[1]  3.83103630 -0.06790002
> opt1$value
[1] 6581.155
> #Interpret the coefficient: Everything else is equal, if the age of the individual increases,
> #the probability of his/her labor market participation will decrease.
```

#3.4==========

```
> #Can you estimate the same model including wages as a deterxminant of labor market participation? Explain.
> dat3 = dat2 %>% drop_na(wage)
> probit2 = function(beta,x1,x2,y) {
+   xbeta = beta[1] + beta[2]*x1+beta[3]*x2
+   pr = pnorm(xbeta)
+   pr[pr>0.999999] = 0.999999
+   pr[pr<0.000001] = 0.000001
+   like = y*log(pr) + (1-y)*log(1-pr)
+   return(-sum(like,log=TRUE))
+ }
> #Set the dependent and independent variables
> a1 = dat3$age
> a2 = dat3$wage
> y1 = dat3$employed
> #Use probit regression in R
> reg3 = glm(y1~a1+a2,family = binomial(link = "probit"))
Warning messages:
1: glm.fit: algorithm did not converge
2: glm.fit: fitted probabilities numerically 0 or 1 occurred
> summary(reg3)

Call:
glm(formula = y1 ~ a1 + a2, family = binomial(link = "probit"))

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-8.4904  -0.3414   0.1534   0.4369   2.8676

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  2.440e+00  5.918e-02   41.22   <2e-16 ***
a1          -5.511e-02  1.036e-03  -53.18   <2e-16 ***
a2           6.351e-05  1.384e-06   45.88   <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: 21923.3  on 16620  degrees of freedom
Residual deviance:  9533.3  on 16618  degrees of freedom
AIC: 9539.3

Number of Fisher Scoring iterations: 25
```

```
> #Test whether the function is correct
> test_beta = reg3$coefficients
> test_beta
  (Intercept)            a1            a2
 2.439543e+00 -5.511476e-02  6.350692e-05
> probit2(test_beta,a1,a2,y1) #4636.991
[1] 4635.991
> logLik(reg3) #-4766.646
'log Lik.' -4766.646 (df=3)
> #I think I can't use the same model because the result from the function is different from the result by regression.
> #The algorithm did not converge and fitted probabilities numerically 0 or 1 occurred.
```

#Exercise4==========

#We are interested in the effect of age on labor market participation.

#Use the pooled version of the data from 2005 to 2015. Additional controls include time-fixed effects.

#4.1==========

```
#Use the pooled version of the data from 2005 to 2015
datind2005 = fread("datind2005.csv",colClasses=c(idmen = "character",idind = "character"))
datind2006 = fread("datind2006.csv",colClasses=c(idmen = "character",idind = "character"))
datind2007 = fread("datind2007.csv",colClasses=c(idmen = "character",idind = "character"))
datind2008 = fread("datind2008.csv",colClasses=c(idmen = "character",idind = "character"))
datind2009 = fread("datind2009.csv",colClasses=c(idmen = "character",idind = "character"))
datind2010 = fread("datind2010.csv",colClasses=c(idmen = "character",idind = "character"))
datind2011 = fread("datind2011.csv",colClasses=c(idmen = "character",idind = "character"))
datind2012 = fread("datind2012.csv",colClasses=c(idmen = "character",idind = "character"))
datind2013 = fread("datind2013.csv",colClasses=c(idmen = "character",idind = "character"))
datind2014 = fread("datind2014.csv",colClasses=c(idmen = "character",idind = "character"))
datind2015 = fread("datind2015.csv",colClasses=c(idmen = "character",idind = "character"))
datind2=rbind(datind2005,datind2006,datind2007,datind2008,datind2009,datind2010,datind2011,
              datind2012,datind2013,datind2014,datind2015)
#Exclude all individuals who are inactive
dat4 = datind2[which(datind2$empstat!="Inactive"),]
```

```
> dat4
            V1               idind             idmen year   empstat respondent profession gender age  wage
     1:      3 1120001006663010001 1200010066630100 2005  Employed          1         38   Male  32 50659
     2:      4 1120001006663010002 1200010066630100 2005  Employed          0         45 Female  28 19231
     3:      5 1120001008245010001 1200010082450100 2005   Retired          1            Female  90     0
     4:      6 1120001008644010001 1200010086440100 2005  Employed          1         34   Male  37 31511
     5:      7 1120001008644010002 1200010086440100 2005  Employed          0         42 Female  35 24873
    ---
190292: 26640 1291065801575010001 2910658015750100 2015  Employed          1         22 Female  57     0
190293: 26641 1291065801575010002 2910658015750100 2015  Employed          0         34   Male  68 92729
190294: 26642 1291065805741010001 2910658057410100 2015   Retired          1       <NA>   Male  82     0
190295: 26643 1291065805741010002 2910658057410100 2015   Retired          0       <NA> Female  81     0
190296: 26644 1291065809437010001 2910658094370100 2015   Retired          1       <NA>   Male  63     0
```

#4.2==========

```
> #Write and optimize the probit, logit, and the linear probability models
> #Remember, for the logit model, F(x) is the logistic function exp(x)/(1+exp(x))
> dat4 = dat4 %>% mutate(employed = ifelse(dat4$empstat == "Employed",1,0),
+                        y05 = ifelse(dat4$year=="2005",1,0),
+                        y06 = ifelse(dat4$year=="2006",1,0),
+                        y07 = ifelse(dat4$year=="2007",1,0),
+                        y08 = ifelse(dat4$year=="2008",1,0),
+                        y09 = ifelse(dat4$year=="2009",1,0),
+                        y10 = ifelse(dat4$year=="2010",1,0),
+                        y11 = ifelse(dat4$year=="2011",1,0),
+                        y12 = ifelse(dat4$year=="2012",1,0),
+                        y13 = ifelse(dat4$year=="2013",1,0),
+                        y14 = ifelse(dat4$year=="2014",1,0))
> dat4
```

```
> dat4
           V1               idind            idmen year  empstat respondent profession gender age  wage employed y05 y06 y07 y08
     1:     3 1120001006663010001 1200010066630100 2005 Employed          1         38   Male  32 50659        1   1   0   0   0
     2:     4 1120001006663010002 1200010066630100 2005 Employed          0         45 Female  28 19231        1   1   0   0   0
     3:     5 1120001008245010001 1200010082450100 2005  Retired          1            Female  90     0        0   1   0   0   0
     4:     6 1120001008644010001 1200010086440100 2005 Employed          1         34   Male  37 31511        1   1   0   0   0
     5:     7 1120001008644010002 1200010086440100 2005 Employed          0         42 Female  35 24873        1   1   0   0   0
    ---
190292: 26640 1291065801575010001 2910658015750100 2015 Employed          1         22 Female  57     0        1   0   0   0   0
190293: 26641 1291065801575010002 2910658015750100 2015 Employed          0         34   Male  68 92729        1   0   0   0   0
190294: 26642 1291065805741010001 2910658057410100 2015  Retired          1       <NA>   Male  82     0        0   0   0   0   0
190295: 26643 1291065805741010002 2910658057410100 2015  Retired          0       <NA> Female  81     0        0   0   0   0   0
190296: 26644 1291065809437010001 2910658094370100 2015  Retired          1       <NA>   Male  63     0        0   0   0   0   0
        y09 y10 y11 y12 y13 y14
     1:   0   0   0   0   0   0
     2:   0   0   0   0   0   0
     3:   0   0   0   0   0   0
     4:   0   0   0   0   0   0
     5:   0   0   0   0   0   0
    ---
190292:   0   0   0   0   0   0
190293:   0   0   0   0   0   0
190294:   0   0   0   0   0   0
190295:   0   0   0   0   0   0
190296:   0   0   0   0   0   0
```

#Probit Model

```
> #Write the probit regression and fixed the effect of year by adding dummy variables for each year
> probit3 = function(beta,x,y05,y06,y07,y08,y09,y10,y11,y12,y13,y14,y) {
+   xbeta = beta[1]+beta[2]*x+beta[3]*y05+beta[4]*y06+beta[5]*y07+beta[6]*y08
+   +beta[7]*y09+beta[8]*y10+beta[9]*y11+beta[10]*y12+beta[11]*y13+beta[12]*y14
+   pr = pnorm(xbeta)
+   pr[pr>0.999999] = 0.999999
+   pr[pr<0.000001] = 0.000001
+   like = y*log(pr) + (1-y)*log(1-pr)
+   return(-sum(like))
+ }
>
> reg4 = glm(dat4$employed~dat4$age+dat4$y05+dat4$y06+dat4$y07+dat4$y08+dat4$y09+
+              dat4$y10+dat4$y11+dat4$y12+dat4$y13+dat4$y14,family = binomial(link = "probit"))
> summary(reg4)

Call:
glm(formula = dat4$employed ~ dat4$age + dat4$y05 + dat4$y06 +
    dat4$y07 + dat4$y08 + dat4$y09 + dat4$y10 + dat4$y11 + dat4$y12 +
    dat4$y13 + dat4$y14, family = binomial(link = "probit"))

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.3309  -0.6057   0.3056   0.7375   2.8148

Coefficients:
             Estimate Std. Error  z value Pr(>|z|)
(Intercept)  3.573877   0.018018  198.346   <2e-16 ***
dat4$age    -0.063590   0.000258 -246.455   <2e-16 ***
dat4$y05    -0.001498   0.016716   -0.090   0.9286
dat4$y06     0.007246   0.016552    0.438   0.6616
dat4$y07     0.034773   0.016339    2.128   0.0333 *
dat4$y08     0.040796   0.016372    2.492   0.0127 *
dat4$y09    -0.009823   0.016277   -0.603   0.5462
dat4$y10    -0.004281   0.016056   -0.267   0.7898
dat4$y11     0.022757   0.015938    1.428   0.1533
dat4$y12     0.009331   0.015690    0.595   0.5520
dat4$y13    -0.018758   0.015966   -1.175   0.2401
dat4$y14     0.011415   0.015867    0.719   0.4719
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 255165  on 190295  degrees of freedom
Residual deviance: 159784  on 190284  degrees of freedom
AIC: 159808

Number of Fisher Scoring iterations: 5
```

```
> #Output the coefficient
> beta = reg4$coefficients
> beta
  (Intercept)     dat4$age     dat4$y05     dat4$y06     dat4$y07     dat4$y08     dat4$y09     dat4$y10     dat4$y11
 3.573876816 -0.063589615 -0.001498005  0.007245616  0.034773071  0.040796017 -0.009822925 -0.004280617  0.022756942
    dat4$y12     dat4$y13     dat4$y14
 0.009331532 -0.018757919  0.011415279
> beta[2] #-0.06358962 (dat4$age)
   dat4$age
-0.06358962
> #Optimize the probit3 model
> start_probit=runif(12,min=-0.064,max=1)
> probit3(start_probit,dat4$age,dat4$y05,dat4$y06,dat4$y07,dat4$y08,dat4$y09,dat4$y10,
+         dat4$y11,dat4$y12,dat4$y13,dat4$y14,dat4$employed)
[1] 116045.3
> opt_probit = optim(start_probit,fn=probit3,method="BFGS",control=list(trace=6,REPORT=1,maxit=1000),
+              x=dat4$age,y05=dat4$y05,y06=dat4$y06,y07=dat4$y07,y08=dat4$y08,y09=dat4$y09,y10=dat4$y10,
+              y11=dat4$y11,y12=dat4$y12,y13=dat4$y13,y14=dat4$y14,y=dat4$employed,hessian=TRUE)
initial  value 116045.254602
iter   2 value 115661.850861
iter   3 value 86586.149413
iter   4 value 85914.128267
iter   5 value 85911.904403
iter   6 value 83165.521127
iter   7 value 80787.809758
iter   8 value 80778.798780
iter   9 value 80015.562131
iter  10 value 79896.977298
iter  11 value 79896.536069
iter  11 value 79896.536037
iter  11 value 79896.535217
final  value 79896.535217
converged
> opt_probit$par[2]
[1] -0.06359803
> opt_probit$value #The model is optimized with likelihood decreased
[1] 79896.54
```

#Logit Model

```
> #Write the logit model and fixed the effect of year
> logit = function(beta,x,y05,y06,y07,y08,y09,y10,y11,y12,y13,y14,y) {
+   xbeta = beta[1]+beta[2]*x+beta[3]*y05+beta[4]*y06+beta[5]*y07+beta[6]*y08
+   +beta[7]*y09+beta[8]*y10+beta[9]*y11+beta[10]*y12+beta[11]*y13+beta[12]*y14
+   pr = exp(xbeta)/(1+exp(xbeta))
+   pr[pr>0.999999] = 0.999999
+   pr[pr<0.000001] = 0.000001
+   like = y*log(pr) + (1-y)*log(1-pr)
+   return(-sum(like,log=TRUE))
+ }
>
> reg5 = glm(dat4$employed~dat4$age+dat4$y05+dat4$y06+dat4$y07+dat4$y08+dat4$y09+
+              dat4$y10+dat4$y11+dat4$y12+dat4$y13+dat4$y14,family = binomial(link = "logit"))
> summary(reg5)
```

```
Call:
glm(formula = dat4$employed ~ dat4$age + dat4$y05 + dat4$y06 +
    dat4$y07 + dat4$y08 + dat4$y09 + dat4$y10 + dat4$y11 + dat4$y12 +
    dat4$y13 + dat4$y14, family = binomial(link = "logit"))

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.2421  -0.5466   0.2595   0.6449   2.8174

Coefficients:
             Estimate Std. Error  z value Pr(>|z|)
(Intercept)  7.0860112  0.0370709  191.148   <2e-16 ***
dat4$age    -0.1241426  0.0005540 -224.098   <2e-16 ***
dat4$y05    -0.0553091  0.0305535   -1.810   0.0703 .
dat4$y06    -0.0348003  0.0302203   -1.152   0.2495
dat4$y07     0.0062686  0.0298080    0.210   0.8334
dat4$y08     0.0265171  0.0298754    0.888   0.3748
dat4$y09    -0.0498114  0.0296793   -1.678   0.0933 .
dat4$y10    -0.0410544  0.0292294   -1.405   0.1602
dat4$y11     0.0130749  0.0290068    0.451   0.6522
dat4$y12    -0.0000687  0.0285544   -0.002   0.9981
dat4$y13    -0.0426575  0.0290634   -1.468   0.1422
dat4$y14     0.0128977  0.0288479    0.447   0.6548
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 255165  on 190295  degrees of freedom
Residual deviance: 153801  on 190284  degrees of freedom
AIC: 153825

Number of Fisher Scoring iterations: 5

> #Output the coefficient
> beta = reg5$coefficients
> beta
  (Intercept)      dat4$age      dat4$y05      dat4$y06      dat4$y07      dat4$y08      dat4$y09      dat4$y10      dat4$y11
 7.086011e+00 -1.241425e-01 -5.530910e-02 -3.480031e-02  6.268598e-03  2.651706e-02 -4.981138e-02 -4.105440e-02  1.307491e-02
     dat4$y12      dat4$y13      dat4$y14
-6.869765e-05 -4.265746e-02  1.289771e-02
> beta[2] #-0.1241425 (dat4$age)
  dat4$age
-0.1241425
>
> #Optimize the logit model
> start_logit=runif(12,min=-0.13)
> logit(reg5$coefficients,dat4$age,dat4$y05,dat4$y06,dat4$y07,dat4$y08,dat4$y09,dat4$y10,
+       dat4$y11,dat4$y12,dat4$y13,dat4$y14,dat4$employed)
[1] 76906.43
> opt_logit = optim(reg5$coefficients,fn=logit,method="BFGS",control=list(trace=6,REPORT=1,maxit=1000),
+              x=dat4$age,y05=dat4$y05,y06=dat4$y06,y07=dat4$y07,y08=dat4$y08,y09=dat4$y09,y10=dat4$y10,
+              y11=dat4$y11,y12=dat4$y12,y13=dat4$y13,y14=dat4$y14,y=dat4$employed,hessian=TRUE)
initial  value 76906.432601
iter   2 value 76905.729356
iter   3 value 76905.352792
iter   4 value 76905.302139
iter   5 value 76905.299339
iter   6 value 76905.124132
iter   7 value 76904.711962
iter   7 value 76904.711519
iter   7 value 76904.710712
final  value 76904.710712
converged
> opt_logit$par[2]
  dat4$age
-0.1241426
> opt_logit$value
[1] 76904.71
```

#Linear Probability Model

```
> #Write the linear probability model
> linear = function(beta,x,y05,y06,y07,y08,y09,y10,y11,y12,y13,y14,y) {
+   y_hat = beta[1]+beta[2]*x+beta[3]*y05+beta[4]*y06+beta[5]*y07+beta[6]*y08
+   +beta[7]*y09+beta[8]*y10+beta[9]*y11+beta[10]*y12+beta[11]*y13+beta[12]*y14
+   y_hat = as.numeric(y_hat)
+   e = (y - y_hat)
+   return(sum(e^2))
+ }
>
> reg6= glm(dat4$employed~dat4$age+dat4$y05+dat4$y06+dat4$y07+dat4$y08+dat4$y09+
+             dat4$y10+dat4$y11+dat4$y12+dat4$y13+dat4$y14)
> summary(reg6)

Call:
glm(formula = dat4$employed ~ dat4$age + dat4$y05 + dat4$y06 +
    dat4$y07 + dat4$y08 + dat4$y09 + dat4$y10 + dat4$y11 + dat4$y12 +
    dat4$y13 + dat4$y14)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-1.27519  -0.23173   0.02784   0.28419   1.09432

Coefficients:
             Estimate Std. Error  t value Pr(>|t|)
(Intercept)  1.5470015  0.0037357  414.111   <2e-16 ***
dat4$age    -0.0184661  0.0000484 -381.532   <2e-16 ***
dat4$y05    -0.0023945  0.0040345   -0.594    0.553
dat4$y06    -0.0002178  0.0039994   -0.054    0.957
dat4$y07     0.0041216  0.0039451    1.045    0.296
dat4$y08     0.0051784  0.0039459    1.312    0.189
dat4$y09    -0.0041762  0.0039341   -1.062    0.288
dat4$y10    -0.0032870  0.0038873   -0.846    0.398
dat4$y11     0.0029301  0.0038576    0.760    0.448
dat4$y12     0.0011928  0.0038018    0.314    0.754
dat4$y13    -0.0046855  0.0038750   -1.209    0.227
dat4$y14     0.0021611  0.0038557    0.560    0.575
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for gaussian family taken to be 0.13503)

    Null deviance: 45430  on 190295  degrees of freedom
Residual deviance: 25694  on 190284  degrees of freedom
AIC: 159029

Number of Fisher Scoring iterations: 2

> #Output the coefficient
> beta = reg6$coefficients
> beta
  (Intercept)      dat4$age      dat4$y05      dat4$y06      dat4$y07      dat4$y08      dat4$y09      dat4$y10      dat4$y11
 1.5470014501 -0.0184661429 -0.0023945445 -0.0002178286  0.0041215725  0.0051783753 -0.0041762078 -0.0032869638  0.0029301380
     dat4$y12      dat4$y13      dat4$y14
 0.0011927466 -0.0046855405  0.0021611125
> beta[2] #-0.01846614 (dat4$age)
   dat4$age
-0.01846614
```

```
> #Optimize the linear probability model
> start_linear=runif(12,min=-0.02,max=1)
> linear(beta,dat4$age,dat4$y05,dat4$y06,dat4$y07,dat4$y08,dat4$y09,dat4$y10,
+        dat4$y11,dat4$y12,dat4$y13,dat4$y14,dat4$employed)
[1] 25695.19
> opt_linear = optim(reg6$coefficients,fn=linear,method="BFGS",control=list(trace=6,REPORT=1,maxit=1000),
+              x=dat4$age,y05=dat4$y05,y06=dat4$y06,y07=dat4$y07,y08=dat4$y08,y09=dat4$y09,y10=dat4$y10,
+              y11=dat4$y11,y12=dat4$y12,y13=dat4$y13,y14=dat4$y14,y=dat4$employed,hessian=TRUE)
initial  value 25695.188370
iter   2 value 25695.145915
iter   3 value 25695.120404
iter   4 value 25695.119438
iter   4 value 25695.119438
iter   5 value 25695.119052
iter   6 value 25695.116607
iter   6 value 25695.116265
iter   6 value 25695.116260
final  value 25695.116260
converged
> opt_linear$par[2]
  dat4$age
-0.0184598
> opt_linear$value
[1] 25695.12
```

#4.3==========

```
> #Interpret and compare the estimated coefficients. How significant are they?
> opt_probit$par[2] #The estimated coefficients -0.06359803
[1] -0.06359803
> opt_logit$par[2] #The estimated coefficients in logit model is -0.1241426
  dat4$age
-0.1241426
> opt_linear$par[2] #The estimated coefficients in llinear probability model is -0.0184598
  dat4$age
-0.0184598
> #The estimated coefficients in each model are all negative, which indicates that when everything else is equal,
> #the individual is less likely to participate in job with the increase of age.


> #Compute the t statistics of estimator in probit model
> probit_hessian = opt_probit$hessian[1:6,1:6]
> t_probit = opt_probit$par[2]/sqrt(solve(probit_hessian)[2,2])
> t_probit
[1] -260.195
> #Test the significant
> qt(0.975,df=(nrow(dat4)-13)) #1.959976
[1] 1.959976
> #Since |t_probit| > 1.959976, the estimated coefficient is significant.


> #Compute the t statistics of estimator in logit model
> logit_hessian = opt_logit$hessian[1:6,1:6]
> t_logit = opt_logit$par[2]/sqrt(solve(logit_hessian)[2,2])
> t_logit
 dat4$age
-224.2683
> #Since |t_logit| > 1.959976, the estimated coefficient is significant.
```

```
> #Compute the t statistics of estimator in linear probability model
> y_hat = opt_linear$par[1]+opt_linear$par[2]*dat4$age+opt_linear$par[3]*dat4$y05+opt_linear$par[4]*dat4$y06+
+   opt_linear$par[5]*dat4$y07+opt_linear$par[6]*dat4$y08+opt_linear$par[7]*dat4$y09+opt_linear$par[8]*dat4$y10+
+   opt_linear$par[9]*dat4$y11+opt_linear$par[10]*dat4$y12+opt_linear$par[11]*dat4$y13+opt_linear$par[12]*dat4$y14
> e = (dat4$employed-y_hat)
> s2 = t(e)%*%e/(nrow(dat4)-13)
> X = cbind(rep(1,nrow(dat4)), dat4$age, dat4$y05,dat4$y06,dat4$y07,dat4$y08,dat4$y09,dat4$y10,
+           dat4$y11,dat4$y12,dat4$y13,dat4$y14)
> XX = solve(t(X)%*%X)
> SE2_linear = sqrt(s2*XX[2,2])
> SE2_linear
              [,1]
[1,] 4.840016e-05
> t_linear = opt_linear$par[2]/SE2_linear
> t_linear
          [,1]
[1,] -381.3996
> #Test the significant
> qt(0.975,df=(nrow(dat4)-13)) #1.959976
[1] 1.959976
> #Since |t_linear| > 1.959976, the estimated coefficient is significant.
```

#Exercise5==========

#5.1==========

#Compute the marginal effect of the previous probit and logit models

```
> #Since |t_linear| > 1.959976, the estimated coefficient is significant.
> #Compute the marginal effect of the previous probit and logit models
> y_hat_probit = opt_probit$par[1]+opt_probit$par[2]*dat4$age+opt_probit$par[3]*dat4$y05+opt_probit$par[4]*dat4$y06+
+   opt_probit$par[5]*dat4$y07+opt_probit$par[6]*dat4$y08+opt_probit$par[7]*dat4$y09+opt_probit$par[8]*dat4$y10+
+   opt_probit$par[9]*dat4$y11+opt_probit$par[10]*dat4$y12+opt_probit$par[11]*dat4$y13+opt_probit$par[12]*dat4$y14
> mar1 = mean(dnorm(y_hat_probit))
> probit_margin = mar1*opt_probit$par
> probit_margin[2]
[1] -0.01537953
>
> y_hat_logit = opt_logit$par[1]+opt_logit$par[2]*dat4$age+opt_logit$par[3]*dat4$y05+opt_logit$par[4]*dat4$y06+
+   opt_logit$par[5]*dat4$y07+opt_logit$par[6]*dat4$y08+opt_logit$par[7]*dat4$y09+opt_logit$par[8]*dat4$y10+
+   opt_logit$par[9]*dat4$y11+opt_logit$par[10]*dat4$y12+opt_logit$par[11]*dat4$y13+opt_logit$par[12]*dat4$y14
> mar2 = mean(dlogis(y_hat_logit))
> logit_margin = mar2*opt_logit$par
> logit_margin[2]
  dat4$age
-0.0160158
```

#5.2==========

```r
#Construct the standard errors of the marginal effects. Hint: Boostrap may be the easiest way.
#Suppose we replicate 10 times for 12 coefficients
#Probit model
mat1 = matrix(data=0,nrow=10,ncol=12)
for (i in 1:10) {
  #Sample a new dataset from original data
  dat_boost1 = dat4[sample(1:nrow(dat4),nrow(dat4),replace=TRUE),]
  #Use the data sampled from orginal dataset, and optimize the probit model
  opt_sam_probit = optim(reg4$coefficients,fn=probit3,method="BFGS",control=list(trace=6,REPORT=1,maxit=1000),
        x=dat_boost1$age,y05=dat_boost1$y05,y06=dat_boost1$y06,y07=dat_boost1$y07,y08=dat_boost1$y08,
        y09=dat_boost1$y09,y10=dat_boost1$y10,y11=dat_boost1$y11,y12=dat_boost1$y12,y13=dat_boost1$y13,
        y14=dat_boost1$y14,y=dat_boost1$employed,hessian=TRUE)
  #The estimate value of probit model
  y_hat_probit2 = opt_sam_probit$par[1]+opt_sam_probit$par[2]*dat_boost1$age+opt_sam_probit$par[3]*dat_boost1$y05+
    opt_sam_probit$par[4]*dat_boost1$y06+opt_sam_probit$par[5]*dat_boost1$y07+opt_sam_probit$par[6]*dat_boost1$y08+
    opt_sam_probit$par[7]*dat_boost1$y09+opt_sam_probit$par[8]*dat_boost1$y10+opt_sam_probit$par[9]*dat_boost1$y11+
    opt_sam_probit$par[10]*dat_boost1$y12+opt_sam_probit$par[11]*dat_boost1$y13+opt_sam_probit$par[12]*dat_boost1$y14
  #Compute the marginal effect
  dist1 = mean(dnorm(y_hat_probit2))
  mat1[i,] = dist1*opt_sam_probit$par
}
#Standard error of marginal effect in probit model
se_probit = sd(mat1[,2])
se_probit
```

```
> #Standard error of marginal effect in probit model
> se_probit = sd(mat1[,2])
> se_probit
[1] 2.31937e-05
```

```r
#Logit model
mat2 = matrix(data=0,nrow=10,ncol=12)
for (i in 1:10) {
  #Sample a new dataset from original data
  dat_boost2 = dat4[sample(1:nrow(dat4),nrow(dat4),replace=TRUE),]
  #Use the data sampled from orginal dataset, and optimize the logit model
  opt_sam_logit = optim(reg5$coefficients,fn=logit,method="BFGS",control=list(trace=6,REPORT=1,maxit=1000),
                        x=dat_boost2$age,y05=dat_boost2$y05,y06=dat_boost2$y06,y07=dat_boost2$y07,y08=dat_boost2$y08,
                        y09=dat_boost2$y09,y10=dat_boost2$y10,y11=dat_boost2$y11,y12=dat_boost2$y12,y13=dat_boost2$y13,
                        y14=dat_boost2$y14,y=dat_boos21$employed,hessian=TRUE)
  #The estimate value of logit model
  y_hat_logit2 = opt_sam_logit$par[1]+opt_sam_logit$par[2]*dat_boost1$age+opt_sam_logit$par[3]*dat_boost1$y05+
    opt_sam_logit$par[4]*dat_boost1$y06+opt_sam_logit$par[5]*dat_boost1$y07+opt_sam_logit$par[6]*dat_boost1$y08+
    opt_sam_logit$par[7]*dat_boost1$y09+opt_sam_logit$par[8]*dat_boost1$y10+opt_sam_logit$par[9]*dat_boost1$y11+
    opt_sam_logit$par[10]*dat_boost1$y12+opt_sam_logit$par[11]*dat_boost1$y13+opt_sam_logit$par[12]*dat_boost1$y14
  #Compute the marginal effect
  dist2 = mean(dlogis(y_hat_logit2))
  mat2[i,] = dist2*opt_sam_logit$par
}
#Standard error of marginal effect in probit model
se_logit = sd(mat2[,2])
se_logit
```

```
> #Standard error of marginal effect in probit model
> se_logit = sd(mat2[,2])
> se_logit
[1] 2.803061e-05
```