Homework 2

Zihao Zhang/zz247

```
# Data
# My understanding of the problem is that the data should be
# datind2009 instead of datind2009-2019
# and I think that someone's wage is zero is reasonable,
# so i didn't delete this kind of data
datind2009 <- fread("/Users/zhangzihao/Desktop/duke/613/HW/a1/Data/datind2009.csv",
                    colClasses=c(idind="character",idmen="character"), header=T)
datind2009 <- datind2009[,9:10] %>% na.omit()
```

```
# Exercise 1
# 1
x <- as.matrix(cbind(1, datind2009$age))
y <- matrix(datind2009$wage)
x0 <- x[,2]
m <- 0
dx <- 0
dy <- 0
for (i in 1:length(y)){
  m = m + ((x0[i] - mean(x0)) * (y[i] - mean(y)))
  dx = dx + (x0[i] - mean(x0))^2
  dy = dy + (y[i] - mean(y))^2
}
cor <- m/(dx^0.5*dy^0.5)
cor
# -0.1788512
cor(datind2009$wage,datind2009$age, use = "complete.obs")
# -0.1788512
# using the cor() function getting the same answer
```

```
> cor
[1] -0.1788512
```

```
# 2
beta.hat <- solve(t(x) %*% x) %*% t(x) %*% y
as.numeric(beta.hat)
# -180.1765

# 3
# standard formulas
# e = y-y_hat, sigma^2_hat = et*e/(n-p), var(beta_hat) = sigma^2_hat(xt*x)^(-1)
y_hat <- x %*% beta.hat
e <- y - y_hat
e <- as.matrix(e)
sigma <- t(e) %*%e / (length(y)-2)
sigma <- as.numeric(sigma)
var <- sigma * (t(x) %*% x)^(-1)
var
# 6.500973
```

```
> # 2
> beta.hat <- solve(t(x) %*% x) %*% t(x) %*% y
> as.numeric(beta.hat)
[1] 22075.1066  -180.1765
```

```
> var
          [,1]       [,2]
[1,] 17140.6890 358.683653
[2,]   358.6837   6.500973
```

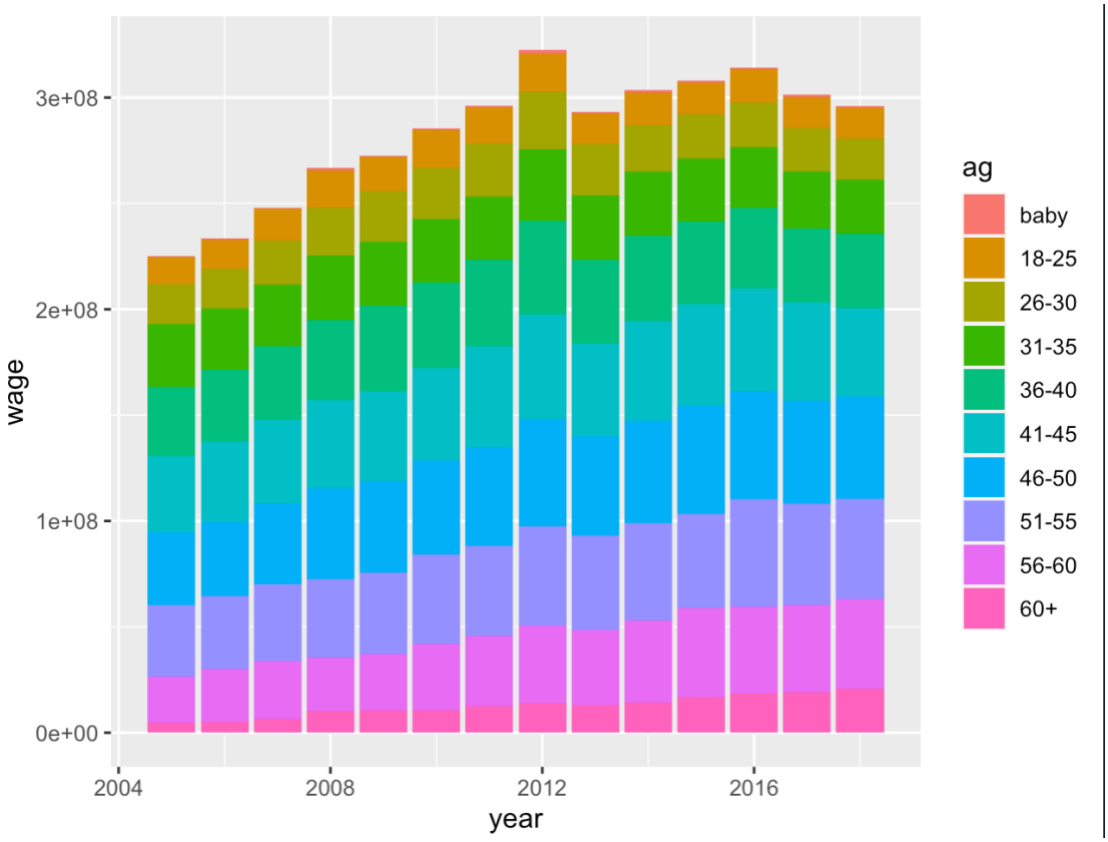
```r
# bootstrap
bootstrap = function(a,b,n){
  betas = c()
  for (i in 1:n) {
    sample <- sample(nrow(x),nrow(x),replace = T)
    boot1 = a[sample,]
    boot2 = b[sample]
    boot_beta = solve(t(boot1) %*% boot1) %*% t(boot1) %*% boot2
    betas = cbind(betas,boot_beta)
  }
  return(betas)
}

apply(bootstrap(x,y,49), MARGIN = 1, sd)
# 4.904961
apply(bootstrap(x,y,499), MARGIN = 1, sd)
# 5.240771
# doing 499 times is more closer to the result of standard formula
```

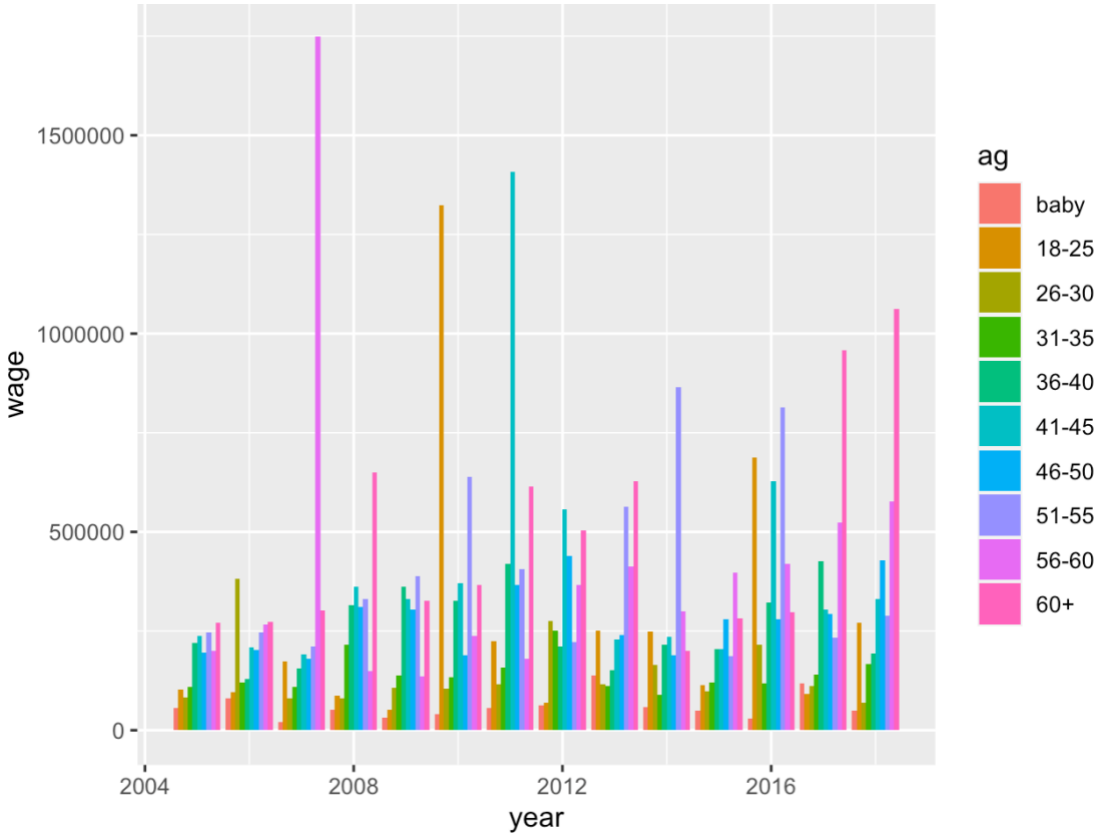
```
> apply(bootstrap(x,y,49), MARGIN = 1, sd)
[1] 281.054761   4.904961
> # 4.91395
> apply(bootstrap(x,y,499), MARGIN = 1, sd)
[1] 303.669074   5.240771
```

```r
# Exercise 2
# Data
path <- "/Users/zhangzihao/Desktop/duke/613/HW/a1/Data"
filenames <- dir(path)
filepath <- sapply(filenames, function(x){
  paste(path,x,sep='/')})
data <- lapply(filepath, function(x){
  fread(x, colClasses=c(idind="character",idmen="character"), header=T)})

datind <- data.frame()
for(i in 17:32){
  datind <- rbind(datind,data[[i]])
}
data2 <- datind %>% filter(year != "2019") %>%  filter(year != "2004")
```

```r
# 1
breaks <- c(-Inf,18,25,30,35,40,45,50,55,60,Inf)
labels <- c('baby','18-25', '26-30', '31-35', '36-40',
            '41-45','46-50','51-55', '56-60', '60+');
data2[,'ag'] <- cut(data2$age, breaks = breaks, labels = labels)

# 2
wage_age <- data2[,c(4,10,11)] %>% na.omit()

p <- ggplot(data = wage_age,
            mapping = aes(
              x = year,
              y = wage,
              fill = ag))
p + geom_col(position = "dodge")
# To show the absolute value of wages
p + geom_col()
# To show the proportion of each age group
# The overall wage level shows a trend of increasing year by year
# In general, the wages of the elderly(especially 56-60) have increased
# compared with the young
```

	idmen	year	empstat	respondent	profession	gender	age	wage	ag
0001	1200010040580100	2005	Inactive	1		Female	31	12334	31-35
0002	1200010040580100	2005	Inactive	0		Female	10	*NA*	baby
0001	1200010066630100	2005	Employed	1	38	Male	32	50659	31-35
0002	1200010066630100	2005	Employed	0	45	Female	28	19231	26-30
0001	1200010082450100	2005	Retired	1		Female	90	0	60+
0001	1200010086440100	2005	Employed	1	34	Male	37	31511	36-40
0002	1200010086440100	2005	Employed	0	42	Female	35	24873	31-35
0001	1200010102990100	2005	Employed	1	55	Female	41	30080	41-45
0002	1200010102990100	2005	Inactive	0		Female	16	0	baby
0001	1200010118450100	2005	Employed	1	37	Male	55	43296	51-55
0002	1200010118450100	2005	Employed	0	54	Female	55	20426	51-55
0001	1200020012930100	2005	Employed	1	11	Male	57	0	56-60
0002	1200020012930100	2005	Employed	0	11	Female	52	0	51-55


3e+08
2e+08
1e+08
0e+00
wage
2004
2008
2012
2016
year
ag
baby
18-25
26-30
31-35
36-40
41-45
46-50
51-55
56-60
60+


1500000
1000000
500000
0
wage
2004
2008
2012
2016
year
ag
baby
18-25
26-30
31-35
36-40
41-45
46-50
51-55
56-60
60+

```r
# 3
data2_1 <- datind %>% select(age,year,wage) %>% filter(year != "2019") %>%
  filter(year != "2004") %>% na.omit()
x_year =  data2_1 %>% select(age,year) %>% mutate(int = 1) %>% as.matrix()
y_year = data2_1$wage %>% as.matrix()
colnames(y) = c('wage')
beta_hat_2 = solve(t(x_year) %*% x_year) %*% t(x_year) %*% y_year
beta_hat_2
# -186.879
# because we only need to observe the coefficient of 'age',
# i don't add dummy variables for each year
# and i will conduct this process in exercise 4

reg <- lm(data2_1$wage ~ data2_1$age + as.factor(data2_1$year))
summary(reg)
# by using the function 'reg', we get the same coefficients
# |beta| is bigger, after adding the fixed effect. 'age' can explain more change of 'wag
```

```
> beta_hat_2
            [,1]
age     -186.8827
year     290.9967
int  -562591.9400
```

```
Coefficients:
                          Estimate Std. Error t value Pr(>|t|)
(Intercept)              20675.058    174.536 118.458  < 2e-16 ***
data2_1$age               -186.879      2.002 -93.366  < 2e-16 ***
as.factor(data2_1$year)2006   21.937    206.900   0.106    0.916
as.factor(data2_1$year)2007  294.803    204.759   1.440    0.150
as.factor(data2_1$year)2008 1425.191    205.328   6.941  3.9e-12 ***
as.factor(data2_1$year)2009 1720.360    205.075   8.389  < 2e-16 ***
as.factor(data2_1$year)2010 1869.525    203.142   9.203  < 2e-16 ***
as.factor(data2_1$year)2011 2116.018    202.051  10.473  < 2e-16 ***
as.factor(data2_1$year)2012 2601.227    199.589  13.033  < 2e-16 ***
as.factor(data2_1$year)2013 2478.843    203.357  12.190  < 2e-16 ***
as.factor(data2_1$year)2014 2749.675    202.408  13.585  < 2e-16 ***
as.factor(data2_1$year)2015 3120.969    202.710  15.396  < 2e-16 ***
as.factor(data2_1$year)2016 3410.113    202.643  16.828  < 2e-16 ***
```

```r
# Exercise 3
# 1
datind2007 <- data[["datind2007.csv"]][,-1]
datind2007 <- datind %>% filter(empstat != "Inactive") %>% filter(empstat != 'Retired')
datind2007 <- na.omit(datind2007)
# 2
probitfunc = 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
  likelihood  = y*log(pr) + (1-y)*log(1-pr)
  return(-sum(likelihood))
}
```

```r
# 3
set.seed(12345)
x_3 = datind2007$age
y_3 = datind2007$empstat
for (i in 1:length(y_3)){
  if(y_3[i] == "Employed"){y_3[i] <- 1
  }
  else{
    y_3[i] <- 0
  }
}
ntry = 100
y_3 <- as.numeric(y_3)
```

```r
out_3 = mat.or.vec(ntry,3)
for (i in 1:ntry){
  start = runif(2,-5,5)
  res = optim(start,fn=probitfunc,
              method="BFGS",
              control=list(trace=6,maxit=1000),
              x=x_3,
              y=y_3)
  out_3[i,c(1,2)] = res$par
  out_3[i,3] = res$value
}
out_3 = data.frame(out_3)
colnames(out_3) = c('int', 'beta^', '-like')
filter(out_3,out_3$'-like' == min(out_3$'-like'))
```

```
> filter(out_3,out_3$'-like' == min(out_3$'-like'))
       int      beta^    -like
1 1.586911 0.00948263 18717.01
```

```r
# 4
# write a function
probitfunc2 = 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
  likelihood = y*log(pr) + (1-y)*log(1-pr)
  return(-sum(likelihood))
}
```

```r
# optimize the model
set.seed(12345)
x_3_1 = datind2007$age
x_3_2 = datind2007$wage
ntry = 1000
out_3_2 = mat.or.vec(ntry,4)
for (i in 1:ntry){
  start = runif(3,-5,5)
  res = optim(start,
              fn=probitfunc2,
              method="BFGS",
              control=list(trace=6,maxit=1000),
              x1=x_3_1,
              x2=x_3_2,
              y=y_3)
  out_3_2[i,c(1,2,3)] = res$par
  out_3_2[i,4] = res$value
}
out_3_2 = data.frame(out_3_2)
colnames(out_3_2) = c('int', 'beta1^','beta2^', '-like')
filter(out_3_2,out_3_2$'-like' == min(out_3_2$'-like'))
```

```
> filter(out_3_2,out_3_2$'-like' == min(out_3_2$'-like'))
       int    beta1^       beta2^    -like
1 0.3229021 0.0131285 0.0002800045 23409.06
```

```r
# Exercise 4
# 1
data4 <- datind %>% filter(year != "2019") %>%  filter(year != "2004") %>%
  filter(year != "2018") %>% filter(year != "2017") %>% filter(year != "2016")
data4 <- data4 %>% filter(empstat != "Inactive") %>% filter(empstat != 'Retired')

# 2
# defining x1,x2 and y. creating dummy variables
x_4_1 = data4$age
y_4 = data4$empstat

# add year dummy variables
data4 <- data4 %>% mutate(dum = 1) %>%
  pivot_wider(names_from = year, values_from = dum, values_fill = 0)
```

```r
# defining the variables of time fixed effect
year06 <- data4$`2006`
year07 <- data4$`2007`
year08 <- data4$`2008`
year09 <- data4$`2009`
year10 <- data4$`2010`
year11 <- data4$`2011`
year12 <- data4$`2012`
year13 <- data4$`2013`
year14 <- data4$`2014`
year15 <- data4$`2015`
```

```r
set.seed(12345)
for (i in 1:length(y_4)){
  if(y_4[i] == "Employed"){y_4[i] <- 1
  }
  else{
    y_4[i] <- 0
  }
}
ntry = 100
y_4 <- as.numeric(y_4) %>% as.matrix()
colnames(y_4) = c('empstat')
```

```r
# probit
probitfunc3 = function(beta,x1,y,
                       year06, year07,year08, year09,
                       year10, year11, year12, year13,
                       year14, year15)
{
  xbeta  = beta[1] + beta[2]*x1 + beta[3]*year06 + beta[4]*year07 +
    beta[5]*year08 + beta[6]*year09 + beta[7]*year10 + beta[8]*year11+
    beta[9]*year12 + beta[10]*year13 + beta[11]*year14 + beta[12]*year15
  pr = pnorm(xbeta)
  pr[pr>0.999999] = 0.999999
  pr[pr<0.000001] = 0.000001
  likelihood  = y*log(pr) + (1-y)*log(1-pr)
  return(-sum(likelihood))
}
```

```r
out_4 = mat.or.vec(ntry,13)
for (i in 1:ntry){
  start = runif(12,-5,5)
  res = optim(start,fn=probitfunc3,
              method="BFGS",
              control=list(trace=6,maxit=1000),
              x1=x_4_1,
              year06=year06,
              year07=year07,
              year08=year08,
              year09=year09,
              year10=year10,
              year11=year11,
              year12=year12,
              year13=year13,
              year14=year14,
              year15=year15,
              y=y_4)
  out_4[i,c(1:12)] = res$par
  out_4[i,13] = res$value
}
```

```r
out_4 = data.frame(out_4)
colnames(out_4) = c('int', 'beta1^','year2006','year2007','year2008','year2009','year201
                    'year2011','year2012','year2013','year2014','year2015',
                    '-like')
probit_out_4 <- filter(out_4,out_4$'-like' == min(out_4$`-like`))
probit_out_4
```

```
> probit_out_4
       int    beta1^   year2006   year2007  year2008   year2009   year2010   year2011
1 0.7487368 0.0123165 0.01742118 0.08102626 0.1103683 0.02728214 0.02224208 0.05644727
    year2012    year2013    year2014    year2015    -like
1 0.01061494 -0.03906386 -0.03288038 -0.05256966 42243.66
```

```
# logit
logitfunc = function(beta,x1,y,
                     year06, year07, year08, year09,
                     year10, year11, year12, year13,
                     year14, year15)
{
  xbeta  = beta[1] + beta[2]*x1 + beta[3]*year06 + beta[4]*year07 +
    beta[5]*year08 + beta[6]*year09 + beta[7]*year10 +
    beta[8]*year11+ beta[9]*year12 + beta[10]*year13 + beta[11]*year14 + beta[12]*year15
  pr = 1/(1+exp(-xbeta))
  pr[pr>0.999999] = 0.999999
  pr[pr<0.000001] = 0.000001
  likelihood = y*log(pr) + (1-y)*log(1-pr)
  return(-sum(likelihood))
}
```

```
ntry = 100
out = mat.or.vec(ntry,13)
for (i in 1:ntry){
  start = runif(12,-5,5)
  res = optim(start,
              fn=logitfunc,
              method="BFGS",
              control=list(trace=6,maxit=1000),
              x1=x_4_1,
              year06=year06,
              year07=year07,
              year08=year08,
              year09=year09,
              year10=year10,
              year11=year11,
              year12=year12,
              year13=year13,
              year14=year14,
              year15=year15,
              y=y_4)
  out[i,c(1:12)] = res$par
  out[i,13] = res$value
}
```

```
out = data.frame(out)
colnames(out) = c('int', 'beta1^','year2006','year2007','year2008','year2009','year2010'
                  'year2011','year2012','year2013','year2014','year2015',
                  '-like')
logit_out_4 <- filter(out,out$'-like' == min(out$`-like`))
logit_out_4
```

```
> logit_out_4
       int     beta1^   year2006  year2007  year2008   year2009   year2010  year2011
1 1.121051 0.02529351 0.03212162 0.1570561 0.2121874 0.04483998 0.03670819 0.1016634
    year2012    year2013    year2014   year2015    -like
1 0.01174351 -0.08479274 -0.07154277 -0.1110621 42213.76
```

```r
# linear
x_l = cbind(rep(1,length(x_4_1)),x_4_1,
          year06,year07,
          year08,year09,
          year10,year11,
          year12,year13,
          year14,year15)
colnames(x_l)[1] = c('int')
colnames(x_l)[2] = c('age')

linear_out_4 = c(solve(t(x_l)%*%x_l)%*%t(x_l)%*%y_4)
linear_out_4 <- data.frame(linear_out_4)
linear_out_4 = t(linear_out_4)
colnames(linear_out_4) = c('int', 'beta1^','year2006','year2007','year2008','year2009','y
                  'year2011','year2012','year2013','year2014','year2015')
linear_out_4
```

```
> linear_out_4
                  int      beta1^    year2006   year2007   year2008    year2009
linear_out_4 0.7977484 0.002335862 0.002933288 0.01394793 0.01844256 0.004083413
               year2010    year2011     year2012     year2013     year2014    year2015
linear_out_4 0.003303572 0.008887393 0.0008988494 -0.008347668 -0.007049788 -0.01091764
```

```r
# 3
# probit
beta_probit = probit_out_4[,1:12]
error = y_4 - x_l %*% t(beta_probit)
sigma = (t(error) %*% error) / (length(x_l) - 12)
sigma = as.numeric(sigma)
se_beta = diag((sigma * (t(x_l) %*% x_l)^(-1))^(1/2))
t_probit = probit_out_4/se_beta
t_probit = t_probit[,-13]
significance <- list()
for (i in 1:12){
  if(t_probit[i]>1.96){
    significance[i] = 1
  }else{
    significance[i] = 0
  }
}
probit_sig <- cbind(t(probit_out_4[,1:12]),significance)
colnames(probit_sig) <- c('variables','significance_5%')
probit_sig
```

```
> probit_sig
        variables   significance_5%
int      0.7487368   1
beta1^   0.0123165   1
year2006 0.01742118  1
year2007 0.08102626  1
year2008 0.1103683   1
year2009 0.02728214  1
year2010 0.02224208  1
year2011 0.05644727  1
year2012 0.01061494  1
year2013 -0.03906386 0
year2014 -0.03288038 0
year2015 -0.05256966 0
```

```r
# logit
beta_logit = logit_out_4[,1:12]
error = y_4 - x_l %*% t(beta_logit)
sigma = (t(error) %*% error) / (length(x_l) - 12)
sigma = as.numeric(sigma)
se_beta = diag((sigma * (t(x_l) %*% x_l)^(-1))^(1/2))
t_logit = logit_out_4/se_beta
t_logit = t_logit[,-13]
significance <- list()
for (i in 1:12){
  if(t_logit[i]>1.96){
    significance[i] = 1
  }else{
    significance[i] = 0
  }
}
logit_sig <- cbind(t(logit_out_4[,1:12]),significance)
colnames(logit_sig) <- c('variables','significance_5%')
logit_sig
```

```
> logit_sig
        variables   significance_5%
int     1.121051    1
beta1^  0.02529351  1
year2006 0.03212162  1
year2007 0.1570561   1
year2008 0.2121874   1
year2009 0.04483998  1
year2010 0.03670819  1
year2011 0.1016634   1
year2012 0.01174351  1
year2013 -0.08479274 0
year2014 -0.07154277 0
year2015 -0.1110621  0
```

```r
# linear
beta_linear = linear_out_4
error = y_4 - x_l %*% t(beta_linear)
sigma = (t(error) %*% error) / (length(x_l) - 12)
sigma = as.numeric(sigma)
se_beta = diag((sigma * (t(x_l) %*% x_l)^(-1))^(1/2))
t_linear = linear_out_4/se_beta
significance <- list()
for (i in 1:12){
  if(t_linear[i]>1.96){
    significance[i] = 1
  }else{
    significance[i] = 0
  }
}
linear_sig <- cbind(t(linear_out_4),significance)
colnames(linear_sig) <- c('variables','significance_5%')
linear_sig
```

```
> linear_sig
         variables    significance_5%
int      0.7977484    1
beta1^   0.002335862  1
year2006 0.002933288  1
year2007 0.01394793   1
year2008 0.01844256   1
year2009 0.004083413  1
year2010 0.003303572  1
year2011 0.008887393  1
year2012 0.0008988494 0
year2013 -0.008347668 0
year2014 -0.007049788 0
year2015 -0.01091764  0
```

```
# Exercise 5
# 1
# Marginal effect evaluated at the mean
x_age_mean = mean(data4$age)
year06_mean = mean(data4$'2006')
year07_mean = mean(data4$'2007')
year08_mean = mean(data4$'2008')
year09_mean = mean(data4$'2009')
year10_mean = mean(data4$'2010')
year11_mean = mean(data4$'2011')
year12_mean = mean(data4$'2012')
year13_mean = mean(data4$'2013')
year14_mean = mean(data4$'2014')
year15_mean = mean(data4$'2015')
x_mean <- cbind(1,x_age_mean,year06_mean,year07_mean,year08_mean,
                year09_mean,year10_mean,year11_mean,year12_mean,year13_mean,year14_mean,y
```

```
# probit
xB <- t(probit_out_4[,1:12]) %*% x_mean
probit_me = dnorm(xB) %*% as.numeric(probit_out_4[,-13])
probit_me = t(probit_me)
names(probit_me) = c('intercept_me', 'beta_me',
                     'year06_me','year07_me','year08_me',
                     'year09_me','year10_me','year11_me',
                     'year12_me','year13_me','year14_me',
                     'year15_me')
probit_me
```

```
> probit_me
          int    beta1^  year2006  year2007  year2008  year2009  year2010  year2011
[1,] 0.3056473 0.3831387 0.3825906 0.3778825 0.3770281 0.3813342 0.3819902 0.3786912
     year2012  year2013  year2014  year2015
[1,] 0.383291 0.3799498 0.3806331 0.3788947
attr(,"names")
 [1] "intercept_me" "beta_me"      "year06_me"    "year07_me"    "year08_me"
 [6] "year09_me"    "year10_me"    "year11_me"    "year12_me"    "year13_me"
[11] "year14_me"    "year15_me"
```

```r
# logit
xB <- t(logit_out_4[,1:12]) %*% x_mean
logit_me = (exp(-xB)/(1+exp(-xB))^2) %*% as.numeric(logit_out_4[,-13])
logit_me = t(logit_me)
names(logit_me) = c('intercept_me', 'beta_me',
                    'year06_me','year07_me','year08_me',
                    'year09_me','year10_me','year11_me',
                    'year12_me','year13_me','year14_me',
                    'year15_me')
logit_me

# 2
# it is a little bit difficult
```

```
> logit_me
          int    beta1^  year2006  year2007  year2008  year2009  year2010  year2011
[1,] 0.2898564 0.3673147 0.3666086 0.3608062 0.3593587 0.3653124 0.3661278 0.3621353
     year2012  year2013  year2014  year2015
[1,] 0.3684483 0.3627041 0.3633202 0.3618797
attr(,"names")
 [1] "intercept_me" "beta_me"      "year06_me"    "year07_me"    "year08_me"
 [6] "year09_me"    "year10_me"    "year11_me"    "year12_me"    "year13_me"
[11] "year14_me"    "year15_me"
```