ECON613 HW1

Zhilin Tang, zt53

2022-01-19

```
library(ggplot2)
library(gridExtra)
library(dplyr)
library(data.table)
library(tinytex)
```

Exercise 1 Basic Statistics

1.1 Number of households surveyed in 2007

```
dim(dathh2007)[1]
```

```
## [1] 10498
```

1.2 Number of households with a marital status "Couple with kids" in 2005

```
length(which(dathh2005$mstatus=='Couple, with Kids'))
```

```
## [1] 3374
```

1.3 Number of individuals surveyed in 2008

```
length(unique(datind2008$idind))
```

```
## [1] 25510
```

1.4 Number of individuals aged between 25 and 35 in 2016

```
length(which(datind2016$age>=25 & datind2016$age<=35))
```

```
## [1] 2765
```

1.5 Cross-table gender/profession in 2009

```
table(datind2009$gender,datind2009$profession)
```

```
##
##              0  11  12  13  21  22  23  31  33  34  35  37  38  42  43  44  45
##   Female  11  30   8  29  63  65   8  68  85 184  50 179  78 258 437   1 153
##   Male    19  57  19  78 213 114  48  98 107 142  59 260 368 110 117   2  95
##
##             46  47  48  52  53  54  55  56  62  63  64  65  67  68  69
##   Female 410  82  22 782  27 584 353 696  64  35  29  19 147 120  40
##   Male   340 429 215 169 182  98 101  74 443 520 246 159 237 177  82
```

1.6 Distribution of wages in 2005 and 2019. Report the mean, the standard deviation, the inter-decile ratio D9/D1 and the Gini coefficient The formula for the Gini coefficient is

$$GINI = \frac{\sum_{i=1}^{n}\sum_{j=1}^{n}|x_i - x_j|}{2n^2\bar{x}}$$

```
gini = function(x){
  n = length(x)
  numerator = 0
  for (i in 1:n) {
    for (j in 1:n) {
      numerator = numerator + abs(x[i]-x[j])
    }
  }
  return(numerator/(2*n^2*mean(x)))
}
```

```
dsummary = function(x){
  x = x[which(!is.na(x) & x!=0)]
  D1 = quantile(x,0.1,na.rm=TRUE)[[1]]
  D9 = quantile(x,0.9,na.rm=TRUE)[[1]]
  out = c(mean=mean(x,na.rm=TRUE),
          sd=sd(x,na.rm=TRUE),
          D1=D1,
          D9=D9,
          inter_decile_ratio=D9/D1,
          Gini_coefficient=gini(x))
  return(out)
}

wage_dist2005 = dsummary(as.numeric(datind2005$wage))
wage_dist2019 = dsummary(as.numeric(datind2019$wage))
```

Table 1: Distribution of wage in 2005

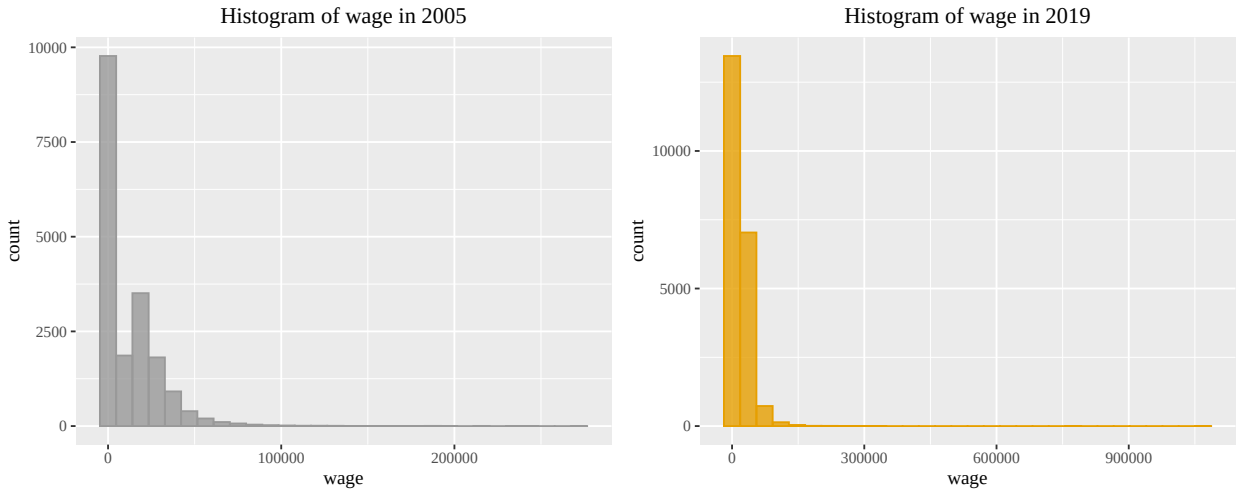
mean	sd	D1	D9	inter_decile_ratio	Gini_coefficient
22,443.03000	18,076.71000	4,547	40,452.50000	8.89653	0.37711

```
p1 = ggplot(datind2005,aes(x=wage)) +
  geom_histogram(color='#999999', fill='#999999',alpha=0.8) +
  ggtitle('Histogram of wage in 2005') +
```

Table 2: Distribution of wage in 2019

mean	sd	D1	D9	inter_decile_ratio	Gini_coefficient
27,578.84000	25,107.19000	3,634	50,375.60000	13.86230	0.39909

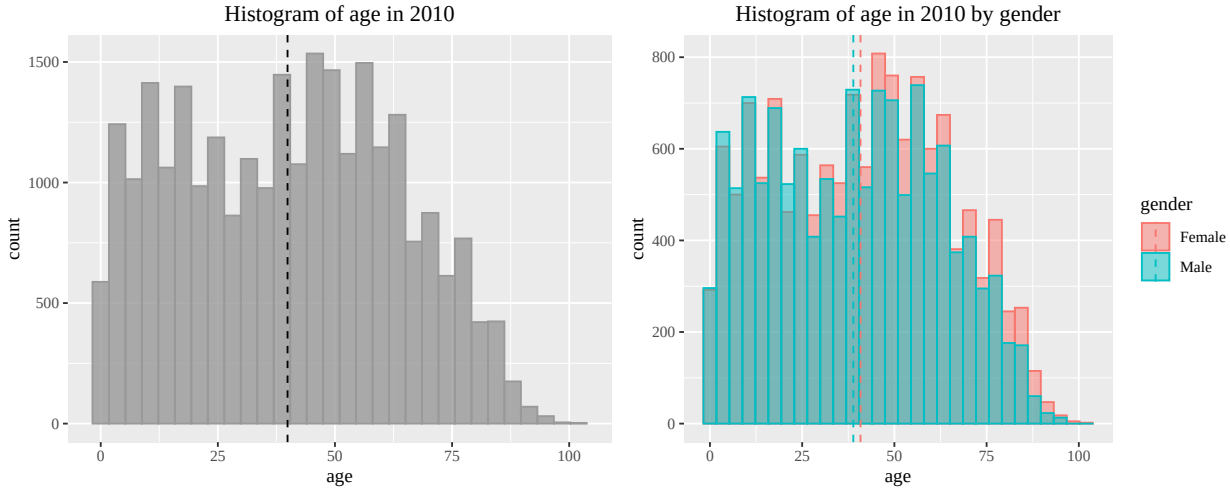
```
  theme(plot.title = element_text(hjust = 0.5),text=element_text(family='Times'))
p2 = ggplot(datind2019,aes(x=wage)) +
  geom_histogram(color='#E69F00', fill='#E69F00',alpha=0.8) +
  ggtitle('Histogram of wage in 2019') +
  theme(plot.title = element_text(hjust = 0.5),text=element_text(family='Times'))
grid.arrange(p1,p2,ncol=2)
```



1.7 Distribution of age in 2010. Plot an histogram. Is there any difference between men and women?

```
age_gender = data.frame(gender=datind2010$gender,age=datind2010$age)
mean_age_gender = age_gender %>%
  group_by(gender) %>%
  summarise(mean_age=mean(age,na.rm=TRUE))
```

```
p3 = ggplot(age_gender,aes(x=age)) +
  geom_histogram(color='#999999', fill='#999999',alpha=0.8) +
  geom_vline(aes(xintercept=mean(age)),color='black', linetype='dashed') +
  ggtitle('Histogram of age in 2010') +
  theme(plot.title = element_text(hjust = 0.5),text=element_text(family='Times'))
p4 = ggplot(age_gender,aes(x=age,fill=gender,color=gender)) +
  geom_histogram(position='identity',alpha=0.5) +
  geom_vline(data=mean_age_gender,aes(xintercept=mean_age,color=gender),
             linetype='dashed') +
  ggtitle('Histogram of age in 2010 by gender') +
  theme(plot.title = element_text(hjust = 0.5),text=element_text(family='Times'))
grid.arrange(p3,p4,ncol=2)
```



Yes, there is some difference between men and women. Females on average have larger age than males, and females are more older skewed.

1.8 Number of individuals in Paris in 2011

```
df2011 = left_join(datind2011[,-1], dathh2011[,-1], by = c('idmen','year'))
df2011_paris = df2011 %>%
  group_by(idind) %>%
  summarise(paris=as.integer(location=='Paris')) %>%
  filter(paris==1)
length(df2011_paris$idind)
```

```
## [1] 3514
```

Exercise 2 Merge Datasets

2.1 Read all individual datasets from 2004 to 2019. Append all these datasets
2.2 Read all household datasets from 2004 to 2019. Append all these datasets

```
for (year in 2004:2019){
  # Read all individual datasets from 2004 to 2019
  dathh_file = data.frame(fread(paste('dathh',year,'.csv',sep=''),header=TRUE))
  assign(paste('dathh',year,sep=''),dathh_file)
  datind_file = data.frame(fread(paste('datind',year,'.csv',sep=''),header=TRUE))
  assign(paste('datind',year,sep=''),datind_file)

  # Append all datasets
  if (year==2004){
    dathh = dathh_file
    datind = datind_file
  }else{
    dathh = rbind(dathh,dathh_file)
    datind = rbind(datind,datind_file)
  }
}
```

2.3 List the variables that are simultaneously present in the individual and household datasets

```
intersect(names(dathh),names(datind))
```

```
## [1] "V1"    "idmen" "year"
```

2.4 Merge the appended individual and household datasets

```
df = left_join(datind[,-1], dathh[,-1], by = c('idmen','year'))
```

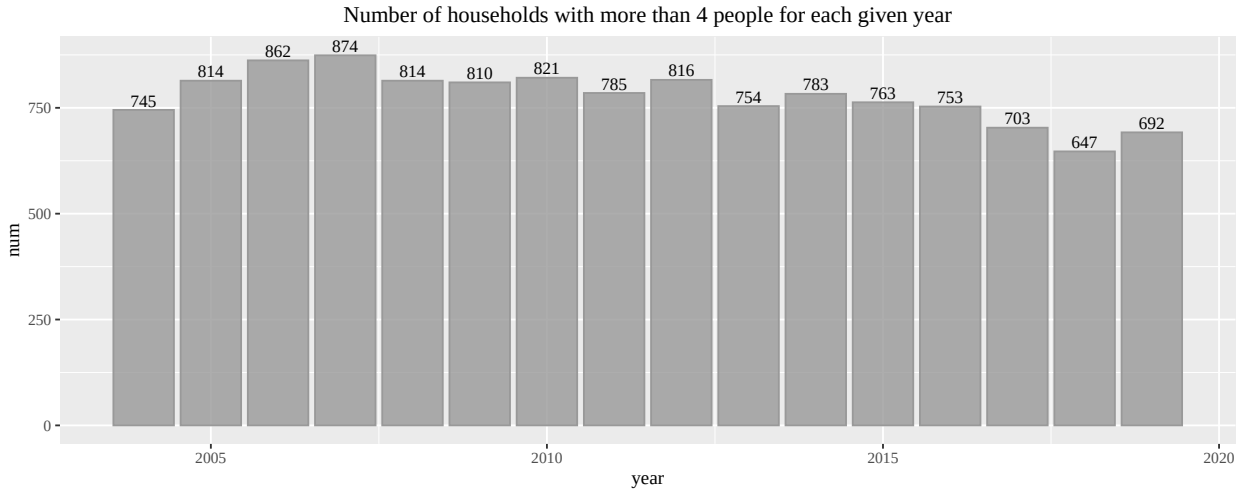
2.5 Number of households in which there are more than four family members

```
df_family_4 = df %>%
  group_by(idmen,year) %>%
  summarise(n=n()) %>%
  filter(n>4) %>%
  select(idmen,year,n)
```

If a household had more than 4 people in 2 different years, the answer can be 1 or 2 (once for each year). Here we think it would be 2.

```
num_by_year = function(data,id){
  num_by_year = c()
  for (year in 2004:2019) {
    if (id=='idmen'){
      num_by_year = c(num_by_year,length(unique(data$idmen[which(data$year==year)])))
    }else if (id=='idind'){
      num_by_year = c(num_by_year,length(unique(data$idind[which(data$year==year)])))
    }
  }
  return(num_by_year)
}
```

```
num_by_year_family_4 = num_by_year(df_family_4,'idmen')
data = data.frame(year=2004:2019,num=num_by_year_family_4)
ggplot(data, aes(x=year, y=num)) +
  geom_bar(stat='identity',color='#999999',fill='#999999',alpha=0.8) +
  geom_text(aes(label=num_by_year_family_4), vjust=-0.3, size=3.5,family='Times') +
  ggtitle('Number of households with more than 4 people for each given year') +
  theme(plot.title = element_text(hjust = 0.5),text=element_text(family='Times'))
```



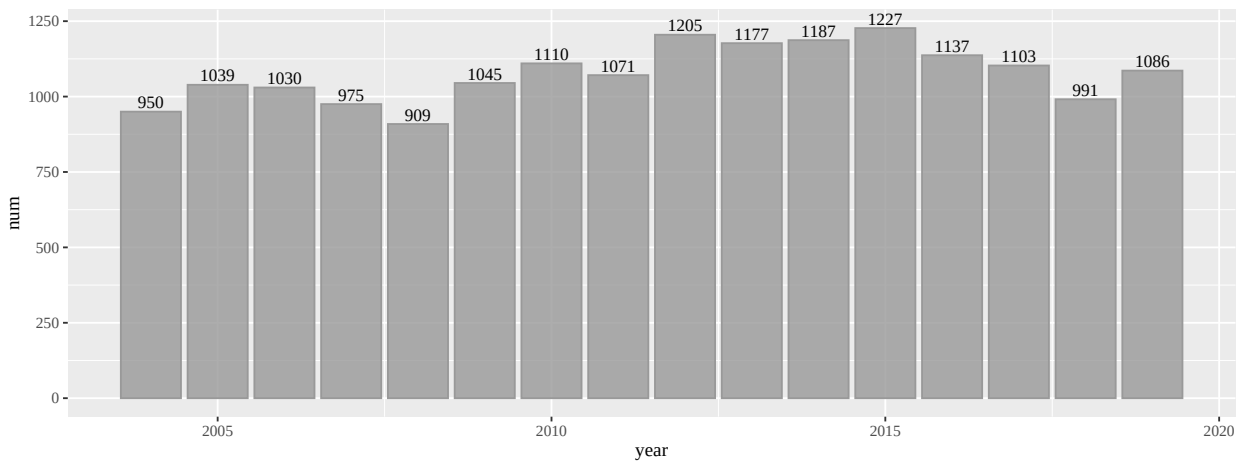

```
sum(num_by_year_family_4)
```

```
## [1] 12436
```

2.6 Number of households in which at least one member is unemployed

```
df_unemployed = df %>%
  group_by(idmen,year) %>%
  summarise(n=sum(empstat=='Unemployed')) %>%
  filter(n>=1) %>%
  select(idmen,year,n)
```

```
num_by_year_unemployed  = num_by_year(df_unemployed ,'idmen')
data = data.frame(year=2004:2019,num=num_by_year_unemployed)
ggplot(data, aes(x=year, y=num)) +
  geom_bar(stat='identity',color='#999999',fill='#999999',alpha=0.8) +
  geom_text(aes(label=num_by_year_unemployed), vjust=-0.3, size=3.5,family='Times') +
  ggtitle('Number of households with >=1 member being unemployed for each given year') +
  theme(plot.title = element_text(hjust = 0.5),text=element_text(family='Times'))
```



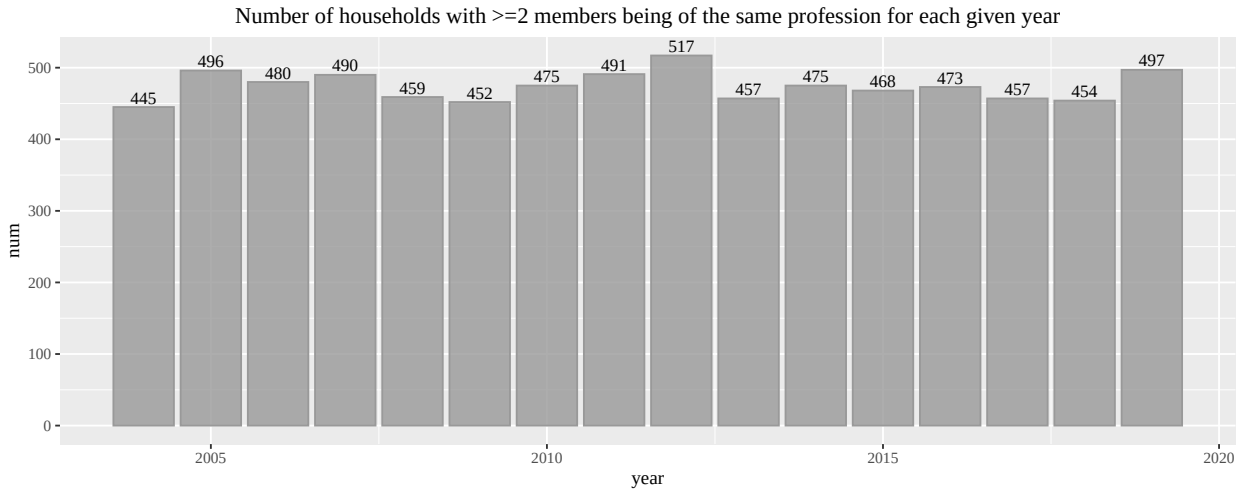
```
sum(num_by_year_unemployed)
```

```
## [1] 17242
```

2.7 Number of households in which at least two members are of the same profession

```
df_profession_completed = df[which(df$profession!='' & df$profession!='NA'),]
df_same_prof = df_profession_completed %>%
  group_by(idmen,year,profession) %>%
  summarise(n=n()) %>%
  filter(n>=2)  %>%
  select(idmen,year,profession,n)
```

```
num_by_year_same_prof = num_by_year(df_same_prof,'idmen')
data = data.frame(year=2004:2019,num=num_by_year_same_prof)
ggplot(data, aes(x=year, y=num)) +
  geom_bar(stat='identity',color='#999999',fill='#999999',alpha=0.8) +
  geom_text(aes(label=num_by_year_same_prof), vjust=-0.3, size=3.5,family='Times') +
  ggtitle(paste('Number of households with >=2 members being of',
                'the same profession for each given year')) +
  theme(plot.title = element_text(hjust = 0.5),text=element_text(family='Times'))
```



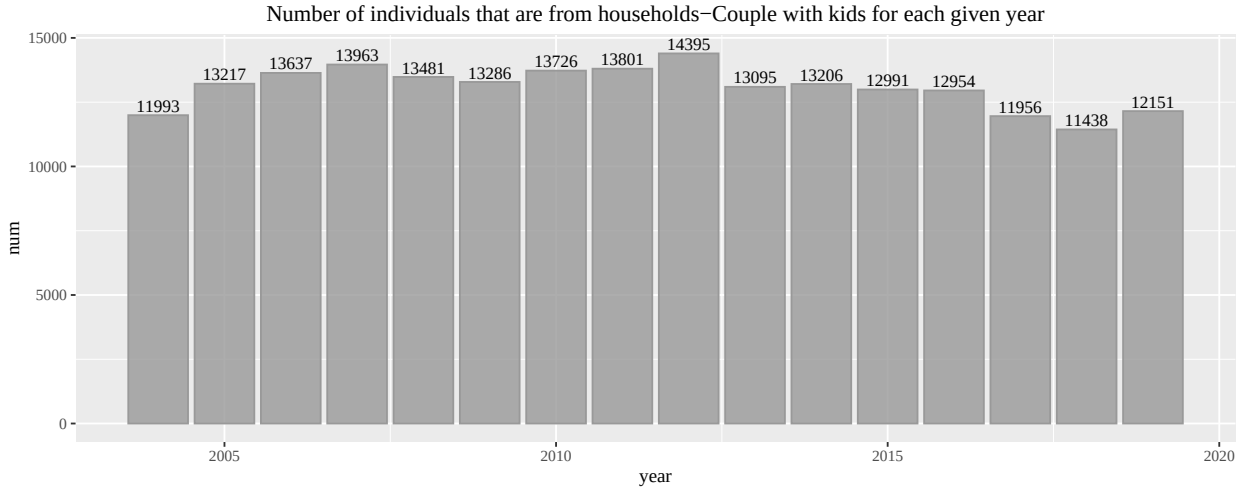

```
sum(num_by_year_same_prof)
```

```
## [1] 7586
```

2.8 Number of individuals in the panel that are from household-Couple with kids

```
df_kids = df %>%
  group_by(idmen,idind,year) %>%
  summarise(kids=as.integer(mstatus=='Couple, with Kids'))%>%
  filter(kids==1)
```

```
num_by_year_kids = num_by_year(df_kids,'idind')
data = data.frame(year=2004:2019,num=num_by_year_kids)
ggplot(data, aes(x=year, y=num)) +
  geom_bar(stat='identity',color='#999999',fill='#999999',alpha=0.8) +
  geom_text(aes(label=num_by_year_kids), vjust=-0.3, size=3.5,family='Times') +
  ggtitle(paste('Number of individuals that are from households-Couple with kids',
                'for each given year')) +
  theme(plot.title = element_text(hjust = 0.5),text=element_text(family='Times'))
```



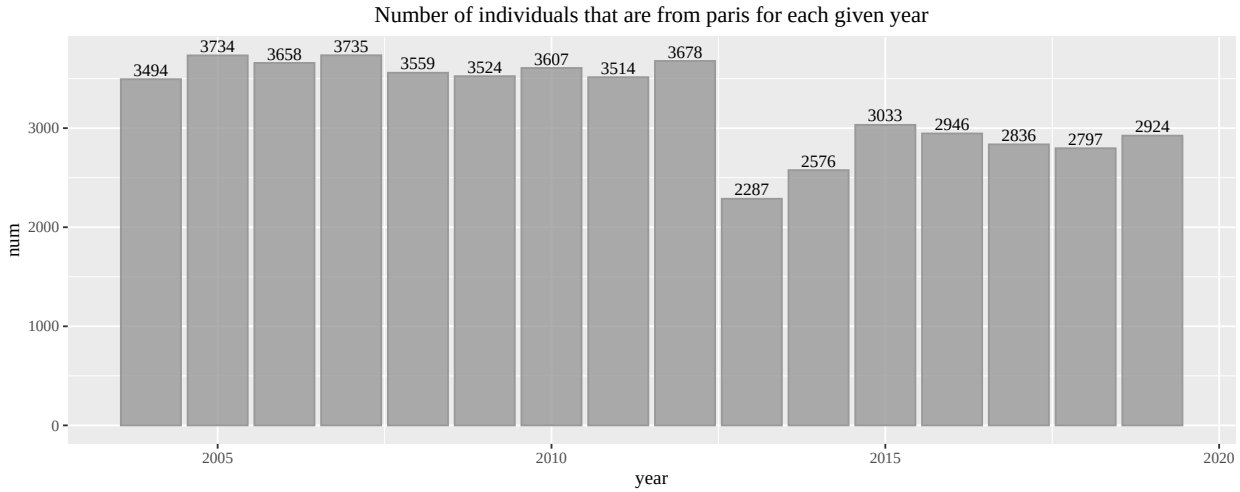

```
sum(num_by_year_kids)
```

```
## [1] 209290
```

2.9 Number of individuals in the panel that are from Paris

```
df_paris = df %>%
  group_by(idind,year) %>%
  summarise(paris=as.integer(location=='Paris')) %>%
  filter(paris==1)
```

```
num_by_year_paris = num_by_year(df_paris,'idind')
data = data.frame(year=2004:2019,num=num_by_year_paris)
ggplot(data, aes(x=year, y=num)) +
  geom_bar(stat='identity',color='#999999',fill='#999999',alpha=0.8) +
  geom_text(aes(label=num_by_year_paris), vjust=-0.3, size=3.5,family='Times')+
  ggtitle('Number of individuals that are from paris for each given year') +
  theme(plot.title = element_text(hjust = 0.5),text=element_text(family='Times'))
```




```
sum(num_by_year_paris)
```

```
## [1] 51902
```

2.10 Find the household with the most number of family members. Report its idmen

```
df_family = df %>%
  group_by(idmen,year) %>%
  summarise(n=n()) %>%
  select(idmen,year,n)
```

```
print(paste('The most number of family members is ',max(df_family$n),'.',sep=''))
```

```
## [1] "The most number of family members is 14."
```

```
as.character(df_family$idmen[which(df_family$n==max(df_family$n))])
```

```
## [1] "2207811124040100" "2510263102990100"
```

2.11 Number of households present in 2010 and 2011

```
df_year = df %>%
  filter(year==2010 | year==2011) %>%
  select(idind,idmen,year)
length(intersect(unique(df_year$idmen[df_year$year==2010]),
                 unique(df_year$idmen[df_year$year==2011])))
```

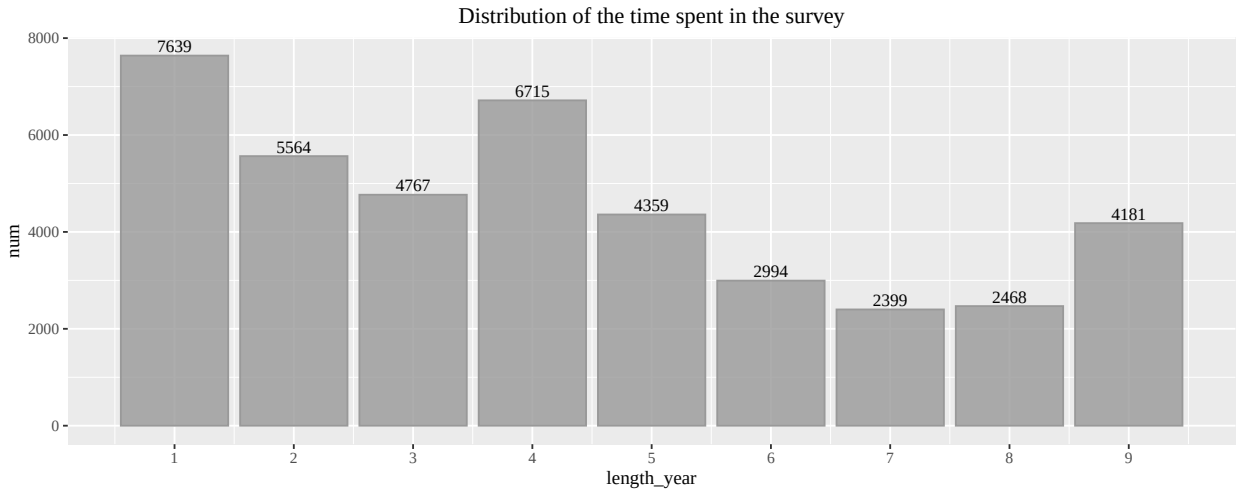
```
## [1] 8984
```

Exercise 3 Migration

3.1 Find out the year each household enters and exit the panel. Report the distribution of the time spent in the survey for each household

```
df_enter_exit = df %>%
  group_by(idmen) %>%
  arrange(year) %>%
  mutate(enter_year=first(year)) %>%
  mutate(exit_year=last(year)+1) %>%
  mutate(length_year=length(unique(year))) %>%
  filter(!is.na(length_year)) %>%
  select(idmen,length_year,enter_year,exit_year)
```

```
length_year_dist = c()
for (i in unique(df_enter_exit$length_year)) {
  length_year_i = length(unique(df_enter_exit$idmen[which(df_enter_exit$length_year==i)]))
  length_year_dist = c(length_year_dist,length_year_i)
}
data = data.frame(length_year=unique(df_enter_exit$length_year),num=length_year_dist)
ggplot(data, aes(x=length_year, y=num)) +
  geom_bar(stat='identity',color='#999999',fill='#999999',alpha=0.8) +
  geom_text(aes(label=length_year_dist),vjust=-0.3, size=3.5,family='Times') +
  ggtitle('Distribution of the time spent in the survey') +
  theme(plot.title = element_text(hjust = 0.5),text=element_text(family='Times')) +
  scale_x_continuous(breaks=unique(df_enter_exit$length_year))
```



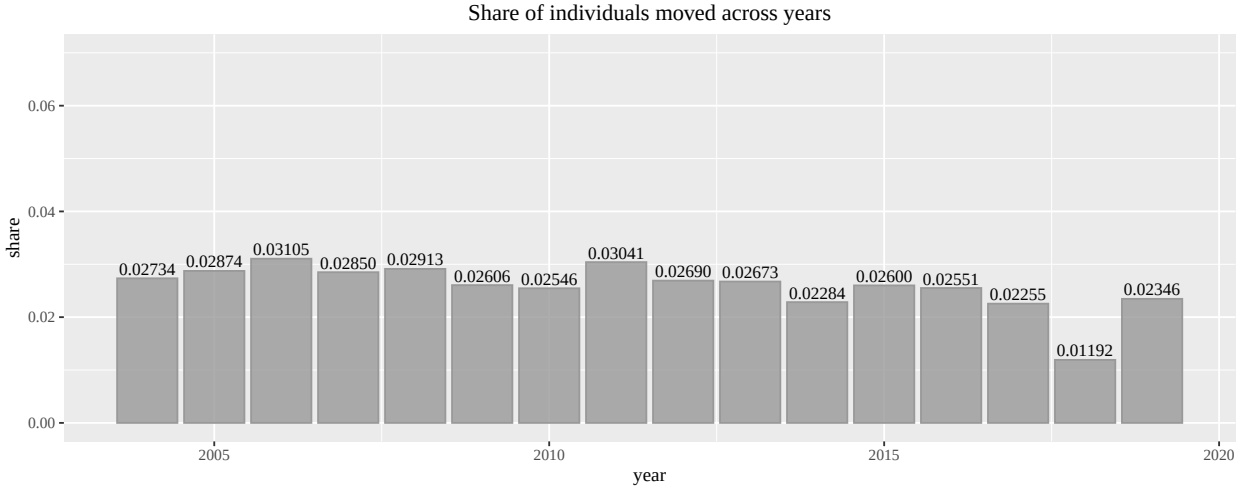
3.2 Base on `datent`, identify whether or not a household moved into its current dwelling at the year of survey. Report the first 10 rows of your result and plot the share of individuals in that situation across years

```
df_move1 = df[!is.na(df$datent),] %>%
  mutate(move_this_year=as.integer(year==datent)) %>%
  select(idind,idmen,year,datent,move_this_year)
df_move1[1:10,]
```

```
##                  idind            idmen year datent move_this_year
## 1  1120001001293010001 1200010012930100 2004   2000              0
## 2  1120001004058010001 1200010040580100 2004   2001              0
```

```
## 3  1120001004058010002 1200010040580100 2004   2001              0
## 4  1120001006663010001 1200010066630100 2004   2000              0
## 5  1120001006663010002 1200010066630100 2004   2000              0
## 6  1120001008245010001 1200010082450100 2004   1957              0
## 7  1120001008644010001 1200010086440100 2004   2001              0
## 8  1120001008644010002 1200010086440100 2004   2001              0
## 9  1120001010299010001 1200010102990100 2004   1990              0
## 10 1120001010299010002 1200010102990100 2004   1990              0
```

```
num_ind_move1 = num_by_year(df_move1[which(df_move1$move_this_year==1),],'idind')
num_ind_total1 = num_by_year(df_move1,'idind')
share_move1 = num_ind_move1/num_ind_total1
data1 = data.frame(year=2004:2019,share=share_move1)
ggplot(data1, aes(x=year, y=share)) +
  geom_bar(stat='identity',color='#999999',fill='#999999',alpha=0.8) +
  geom_text(aes(label=sprintf('%.5f',share_move1)), vjust=-0.3, size=3.5,family='Times') +
  ggtitle('Share of individuals moved across years') +
  theme(plot.title = element_text(hjust = 0.5),text=element_text(family='Times')) +
  scale_y_continuous(limits=c(0,0.07))
```

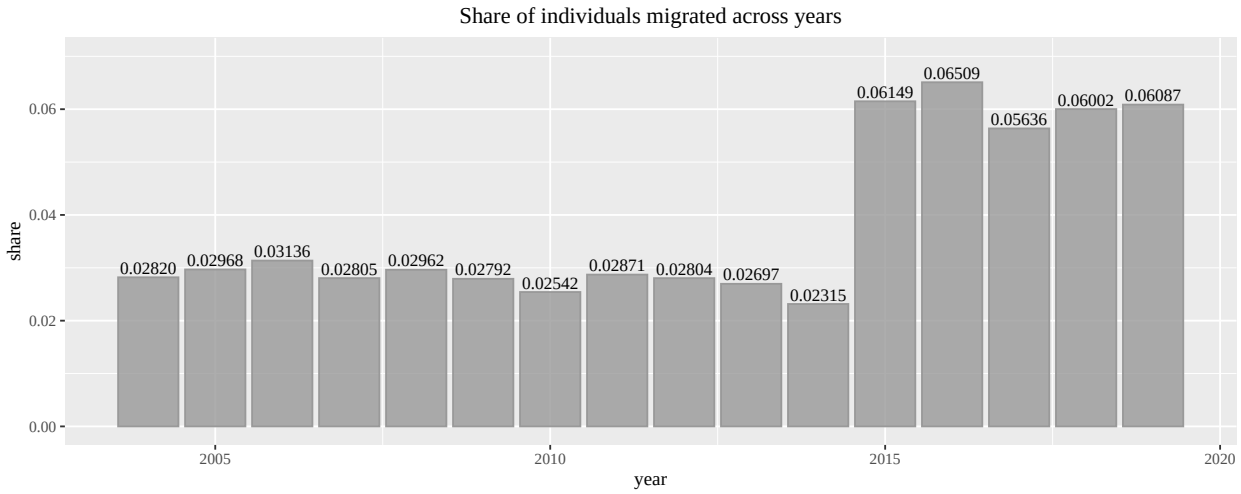


3.3 Base on myear and move, identify whether or not a household migrated at the year of survey. Report the first 10 rows of your result and plot the share of individuals in that situation across years

```
df_b2014 = df[which(df$year<=2014),]
df_a2014 = df[which(df$year>2014),]
df_move_b2014 = df_b2014[!is.na(df_b2014$myear),] %>%
  mutate(move_this_year=as.integer(year==myear)) %>%
  select(idind,idmen,year,myear,move,move_this_year)
df_move_a2014 = df_a2014[!is.na(df_a2014$move),] %>%
  mutate(move_this_year=as.integer(move==2)) %>%
  select(idind,idmen,year,myear,move,move_this_year)
df_move2 = rbind(df_move_b2014,df_move_a2014)
df_move2[1:10,]
```

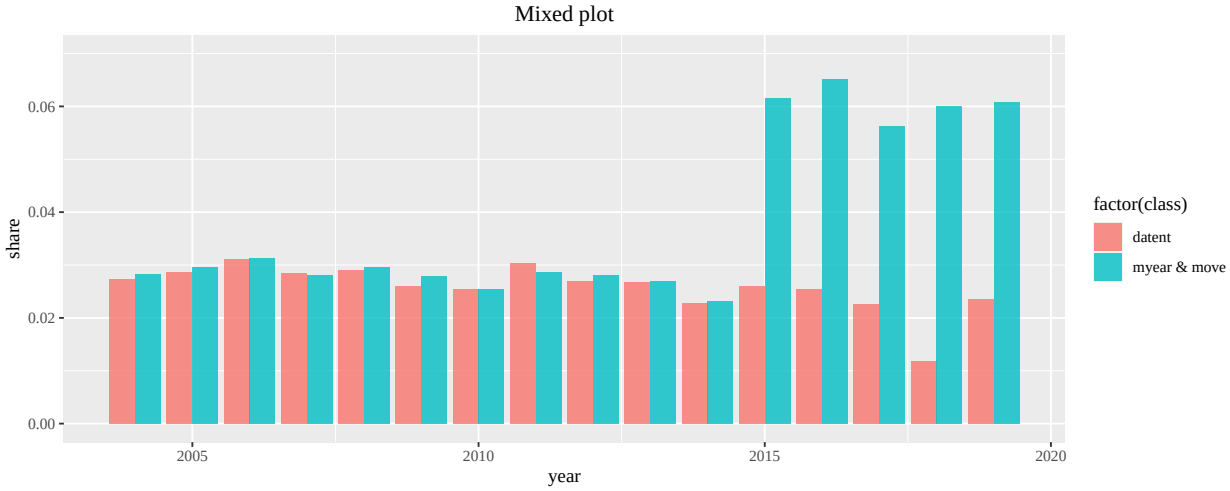
```
##                  idind            idmen year myear move move_this_year
## 1  1120001001293010001 1200010012930100 2004  2000   NA              0
## 2  1120001004058010001 1200010040580100 2004  2001   NA              0
## 3  1120001004058010002 1200010040580100 2004  2001   NA              0
## 4  1120001006663010001 1200010066630100 2004  2000   NA              0
## 5  1120001006663010002 1200010066630100 2004  2000   NA              0
## 6  1120001008245010001 1200010082450100 2004  1957   NA              0
## 7  1120001008644010001 1200010086440100 2004  2001   NA              0
## 8  1120001008644010002 1200010086440100 2004  2001   NA              0
## 9  1120001010299010001 1200010102990100 2004  1990   NA              0
## 10 1120001010299010002 1200010102990100 2004  1990   NA              0
```

```
num_ind_move2 = num_by_year(df_move2[which(df_move2$move_this_year==1),],'idind')
num_ind_total2 = num_by_year(df_move2,'idind')
share_move2 = num_ind_move2/num_ind_total2
data2 = data.frame(year=2004:2019,share=share_move2)
ggplot(data2, aes(x=year, y=share)) +
  geom_bar(stat='identity',color='#999999',fill='#999999',alpha=0.8) +
  geom_text(aes(label=sprintf('%.5f',share_move2)), vjust=-0.3, size=3.5,family='Times') +
  ggtitle('Share of individuals migrated across years') +
  theme(plot.title = element_text(hjust = 0.5),text=element_text(family='Times')) +
  scale_y_continuous(limits=c(0,0.07))
```



3.4 Mix the two plots you created above in one graph, clearly label the graph. Do you prefer one method over the other? justify

```
data1['class'] = 1
data2['class'] = 2
data_mix = rbind(data1,data2)
ggplot(data_mix,aes(x=year, y=share)) +
  geom_bar(stat='identity',aes(fill=factor(class)), alpha=0.8,position='dodge') +
  ggtitle('Mixed plot') +
  theme(plot.title = element_text(hjust = 0.5),text=element_text(family='Times')) +
  scale_y_continuous(limits=c(0,0.07)) +
  scale_fill_discrete(labels = c('datent', 'myear & move'))
```

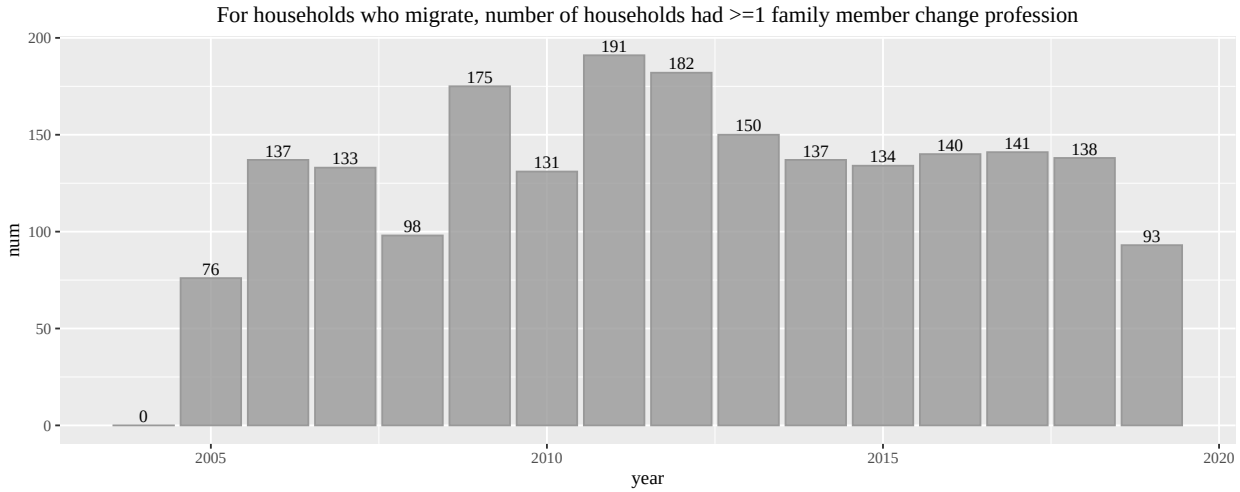


I would prefer to use variable `datent` because `datent` is available during the whole survey period. However, we cannot guarantee that **`myear=year`** and **`move=2`** representing exactly the same thing. From the mixed plot, we observe that before 2014, the differences between shares of individuals based on `datent` and `myear` are small; whereas after 2014, the differences become much larger. This also implies some inconsistent of measure, so using `datent` is a wiser choice.

3.5 For households who migrate, find out how many households had at least one family member change his/her profession or employment status.

```
households = unique(df_move1$idmen[which(df_move1$move_this_year==1)])
df_households = df[which(df$idmen %in% households),]
df_prof_change = df_households[which(df_households$profession!='' &
                                     df_households$profession!='NA'),] %>%
  group_by(idind) %>%
  mutate(num_profession=length(unique(profession))) %>%
  mutate(change_prof=as.integer(profession!=lag(profession))) %>%
  filter(num_profession>1 & change_prof==1)
```

```
num_by_year_prof_change = num_by_year(df_prof_change,'idmen')
data = data.frame(year=2004:2019,num=num_by_year_prof_change)
ggplot(data, aes(x=year, y=num)) +
  geom_bar(stat='identity',color='#999999',fill='#999999',alpha=0.8) +
  geom_text(aes(label=num_by_year_prof_change), vjust=-0.3, size=3.5,family='Times') +
  ggtitle(paste('For households who migrate,',
                'number of households had >=1 family member change profession')) +
  theme(plot.title = element_text(hjust = 0.5),text=element_text(family='Times'))
```




```
sum(num_by_year_prof_change)
```

```
## [1] 2056
```

Exercise 4 Attrition

Compute the attrition across each year, where attrition is defined as the reduction in the number of individuals staying in the data panel. Report your final result as a table in proportions.

```
for (year in 2004:2019){
  if (year==2004){
    idind_last_year = unique(df$idind[which(df$year==2004)])
    prop = c(0)
  }else{
    idind_this_year = unique(df$idind[which(df$year==year)])
    num_exit = sum(!(idind_last_year %in% idind_this_year))
    prop = c(prop,num_exit/length(idind_last_year))
    idind_last_year = idind_this_year
  }
}
```

Table 3: Table of Attrition in proportions

	year	prop
1	2004	0
2	2005	0.13530
3	2006	0.20007
4	2007	0.17871
5	2008	0.22670
6	2009	0.20561
7	2010	0.18379
8	2011	0.19362
9	2012	0.16989
10	2013	0.25462
11	2014	0.21982
12	2015	0.21918
13	2016	0.21723
14	2017	0.25070
15	2018	0.24420
16	2019	0.24313