

# HW1 613

Mike Sun

1/16/2022

## *Excercise1*

Number of households surveyed in 2007:

```
length(dathh2007$idmen)
```

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

Number of households with marital status “Couple with kids” in 2005:

```
nrow(filter(dathh2005, mstatus == "Couple, No kids"))
```

```
## [1] 2656
```

Number of individuals surveyed in 2008:

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

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

Number of individuals aged between 25 and 35 in 2016:

```
length(filter(datind2016, age >= 25 & age <=35))
```

```
## [1] 10
```

Cross-table gender/profession in 2009:

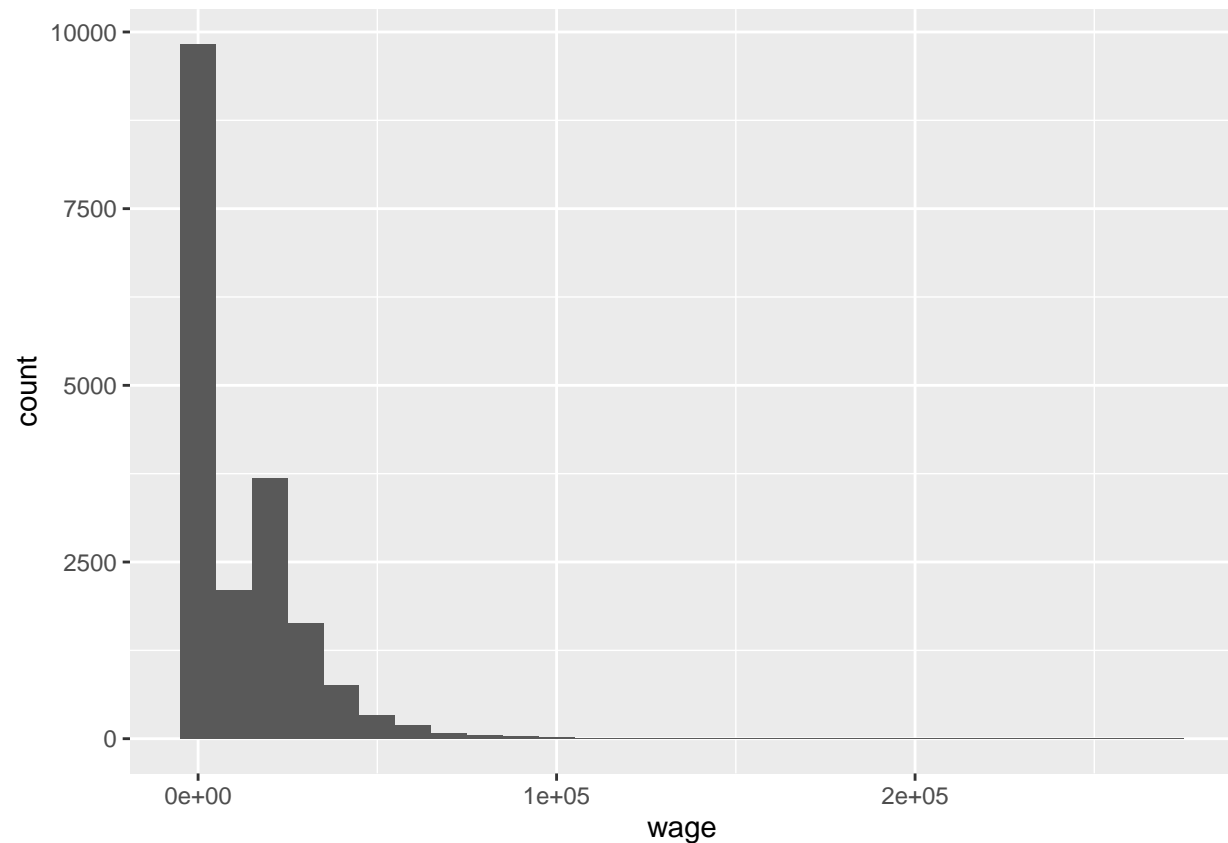
```
gender_prof_table_09 = table(datind2009$profession, datind2009$gender)
gender_prof_table_09
```

```
##
##      Female Male
##  0         11   19
## 11         30   57
## 12          8   19
## 13         29   78
## 21         63  213
## 22         65  114
## 23          8   48
## 31         68   98
## 33         85  107
## 34        184  142
## 35         50   59
## 37        179  260
## 38         78  368
## 42        258  110
```

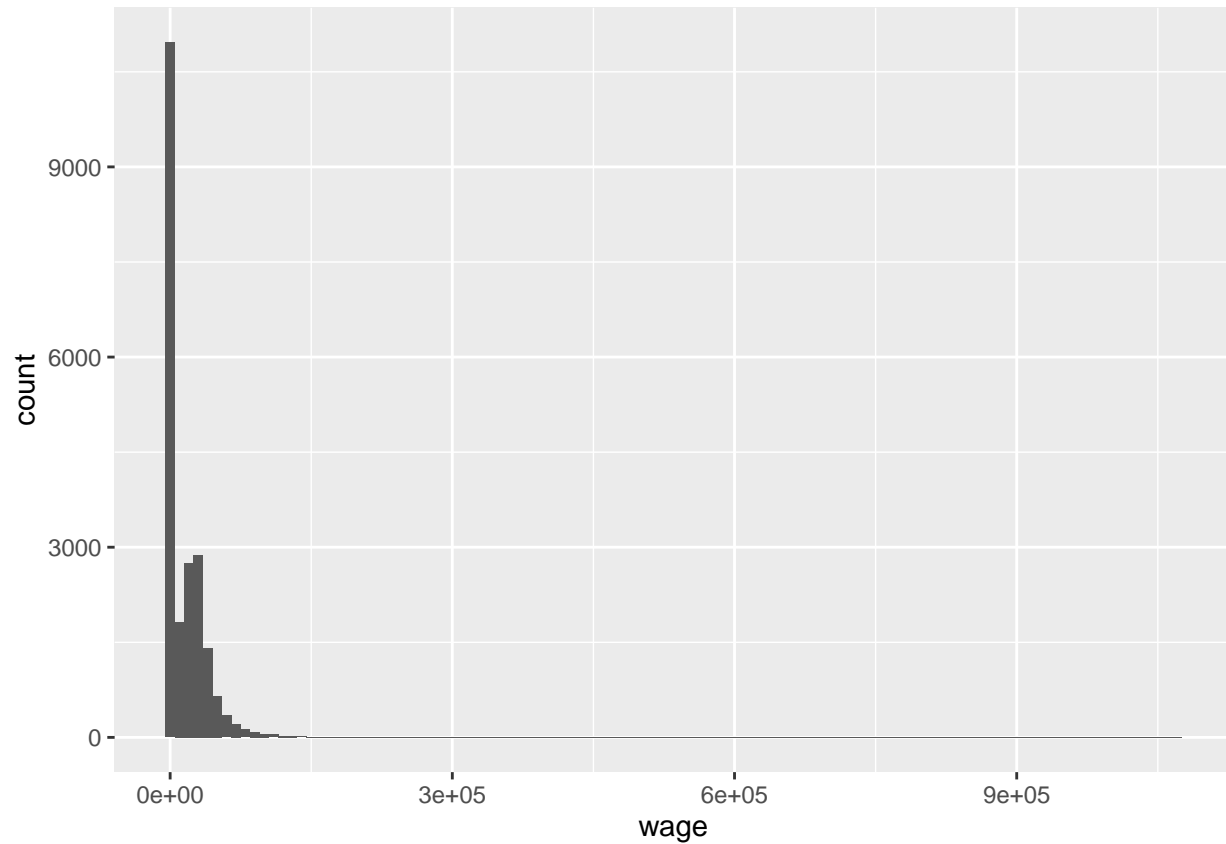
```
## 43 437 117
## 44 1 2
## 45 153 95
## 46 410 340
## 47 82 429
## 48 22 215
## 52 782 169
## 53 27 182
## 54 584 98
## 55 353 101
## 56 696 74
## 62 64 443
## 63 35 520
## 64 29 246
## 65 19 159
## 67 147 237
## 68 120 177
## 69 40 82
```

Distribution of wages in 2005 and 2019. Report the mean, the standard deviation, the inter-decile ratio D9/D1 and the Gini coefficient:

```
datind2005_wageclean = datind2005 %>% filter(!is.na(wage))
datind2019_wageclean = datind2019 %>% filter(!is.na(wage))
ggplot(data=datind2005_wageclean, aes(x=wage))+geom_histogram(binwidth=10000)
```



```
ggplot(data=datind2019_wageclean, aes(x=wage))+geom_histogram(binwidth=10000)
```



```
mean(datind2005_wageclean$wage)
```

```
## [1] 11992.26
```

```
mean(datind2019_wageclean$wage)
```

```
## [1] 15350.47
```

```
sd(datind2005_wageclean$wage)
```

```
## [1] 17318.56
```

```
sd(datind2019_wageclean$wage)
```

```
## [1] 23207.18
```

```
D9_1_2005 = quantile(datind2005_wageclean$wage, 0.9)/quantile(datind2005_wageclean$wage, 0.1)
```

```
D9_1_2019 = quantile(datind2019_wageclean$wage, 0.9)/quantile(datind2019_wageclean$wage, 0.1)
```

```
Gini(datind2005_wageclean$wage)
```

```
## [1] 0.6671654
```

```
Gini(datind2019_wageclean$wage)
```

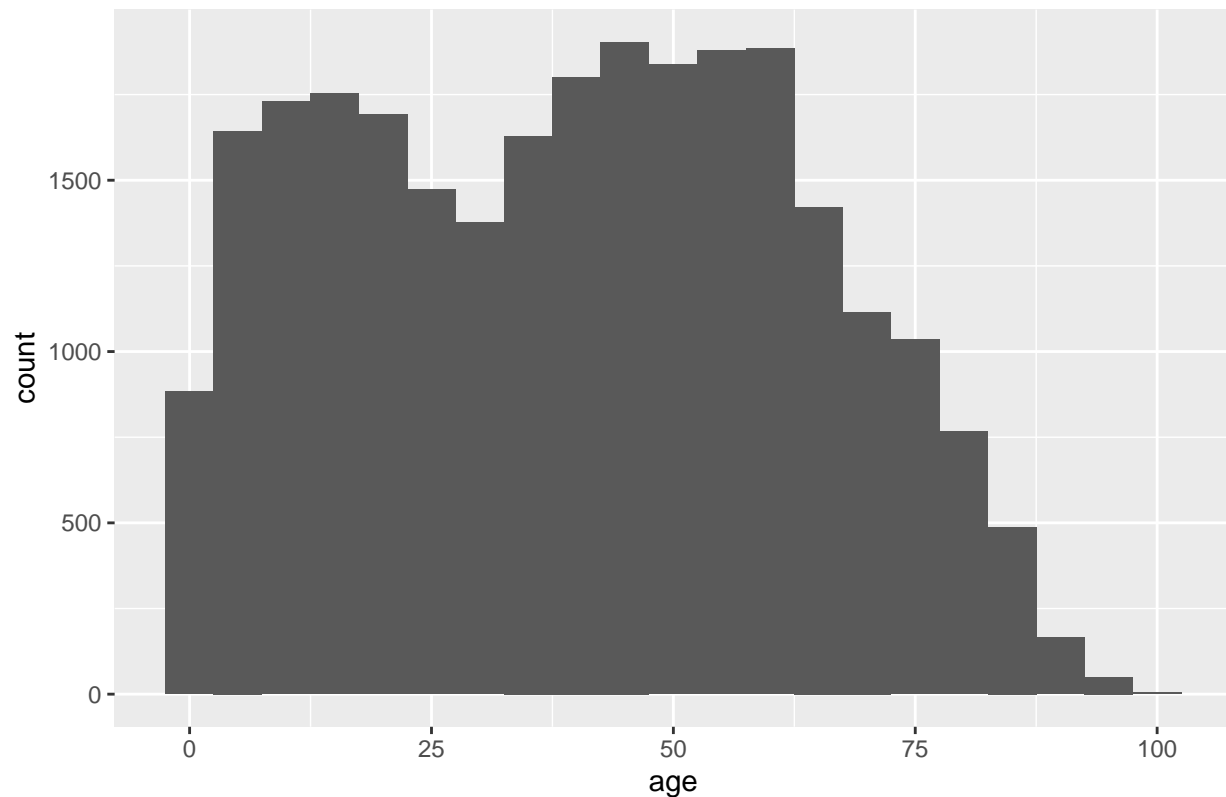
```
## [1] 0.6655301
```

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

```
datind2010_ageclean = datind2010 %>% filter(!is.na(age))

ggplot(datind2010_ageclean) + aes(age) + geom_histogram(binwidth=5) +
  ggtitle("2010 Population Count by Age with Bin size = 4")
```

2010 Population Count by Age with Bin size = 4

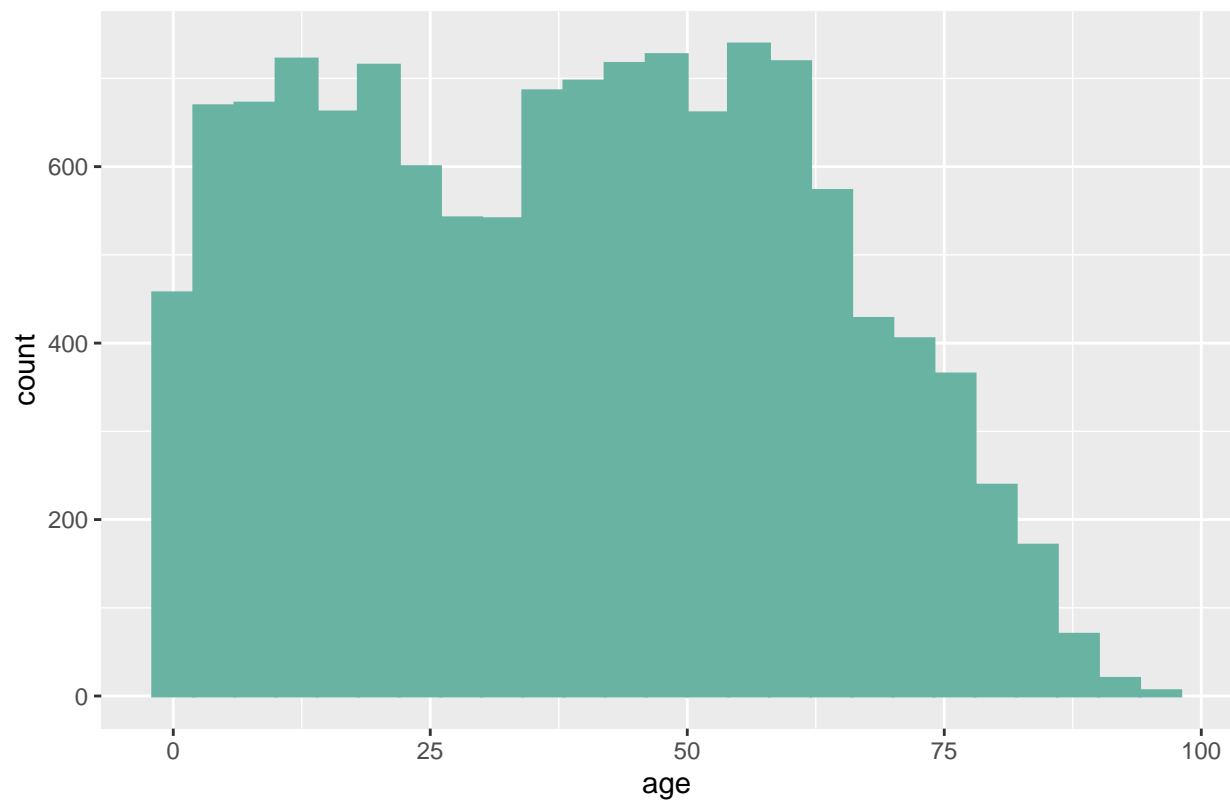


```
male_age_2010 <- datind2010_ageclean %>% filter( gender == "Male" ) %>% ggplot( aes(x=age)) +
  geom_histogram( binwidth=4, fill="#69b3a2", color="#69b3a2") +
  ggtitle("2010 Male Count by Age with Bin size = 4")

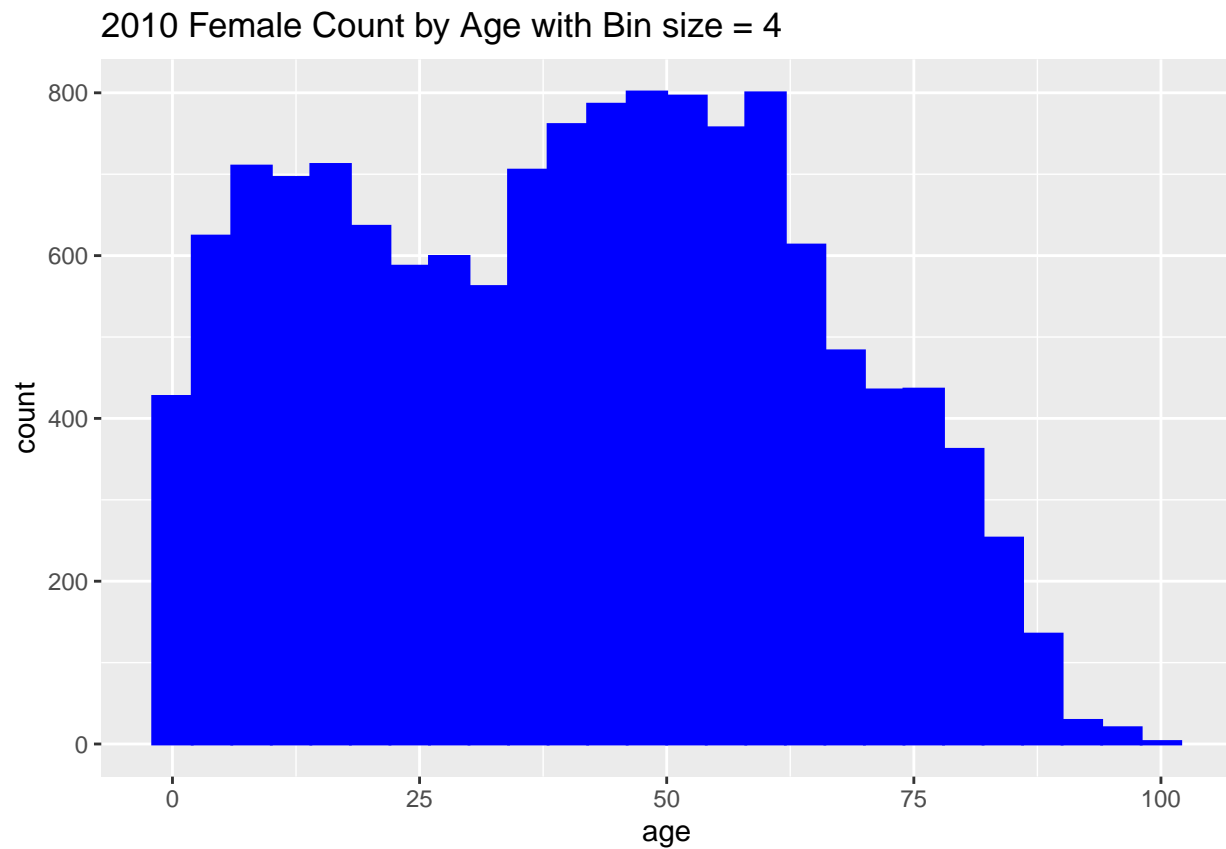
female_age_2010 <- datind2010_ageclean %>% filter( gender == "Female" ) %>% ggplot( aes(x=age)) +
  geom_histogram( binwidth=4, fill = "blue", color="blue") +
  ggtitle("2010 Female Count by Age with Bin size = 4")

male_age_2010
```

2010 Male Count by Age with Bin size = 4



female\_age\_2010



No noticable difference based on the graphs.

Number of individuals in Paris in 2011:

```
Pop_ind = left_join(dathh2011, datind2011, by="idmen")
Paris_ind = Pop_ind %>% filter(location == "Paris")
nrow(Paris_ind)
```

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

## Exercise2

Read all individual datasets from 2004 to 2019. Append all these datasets:

```
datind_full = unique(rbind(datind2004,datind2005,datind2006,datind2007,datind2008,datind2009,
                           datind2010,datind2011,datind2012,datind2013,datind2014,datind2015,
                           datind2016,datind2017,datind2018,datind2019))
```

Read all household datasets from 2004 to 2019. Append all these datasets:

```
dathh_full = unique(rbind(dathh2004,dathh2005,dathh2006,dathh2007,dathh2008,
                           dathh2009,dathh2010,dathh2011,dathh2012,dathh2013,
                           dathh2014,dathh2015,dathh2016,dathh2017,dathh2018,dathh2019))
```

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

```
col_check = colnames(dathh_full) %in% colnames(datind_full)
```

```
i=1
while (i <= length(col_check)){
  if (col_check[i] == "TRUE"){
    print(colnames(dathh_full)[i])
  }
  i= i+1
}
```

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

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

```
Full_Data = unique(left_join(dathh_full,datind_full,by=c("idmen","year")))
```

```
idmen_count <- Full_Data %>% group_by(idmen,year) %>% mutate(count_4 = n()) %>% filter(count_4 > 4)
```

```
length(idmen_count$idmen)
```

```
## [1] 67069
```

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

```
Unemployed = Full_Data %>% group_by(idmen, year, empstat) %>% filter(empstat == "Unemployed")
```

```
length(unique(Unemployed$idmen))
```

```
## [1] 8161
```

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

```
Same_Profession = Full_Data %>% group_by(idmen, year, profession) %>%
  mutate(n_member = n()) %>% filter(n_member > 2)
```

```
length(unique(Same_Profession$idmen))
```

```
## [1] 7032
```

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

```
Couple_kids = Full_Data %>% group_by(idmen,year,mstatus) %>% filter(mstatus == "Couple, with Kids")
length(unique(Couple_kids$idind))
```

```
## [1] 15567
```

Number of individuals in the panel that are from Paris:

```
Paris = Full_Data %>% group_by(idind,year,location) %>% filter(location == "Paris")
length(unique(Paris$idind))
```

```
## [1] 6177
```

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

```
max_idmen = idmen_count %>% filter(count_4 == max(idmen_count$count_4))
unique(max_idmen$idmen)
```

```
## [1] 2.207811e+15 2.510263e+15
```

Number of households present in 2010 and 2011:

```
dathh_1011 = intersect(dathh2010$idmen,dathh2011$idmen)
length(dathh_1011)
```

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

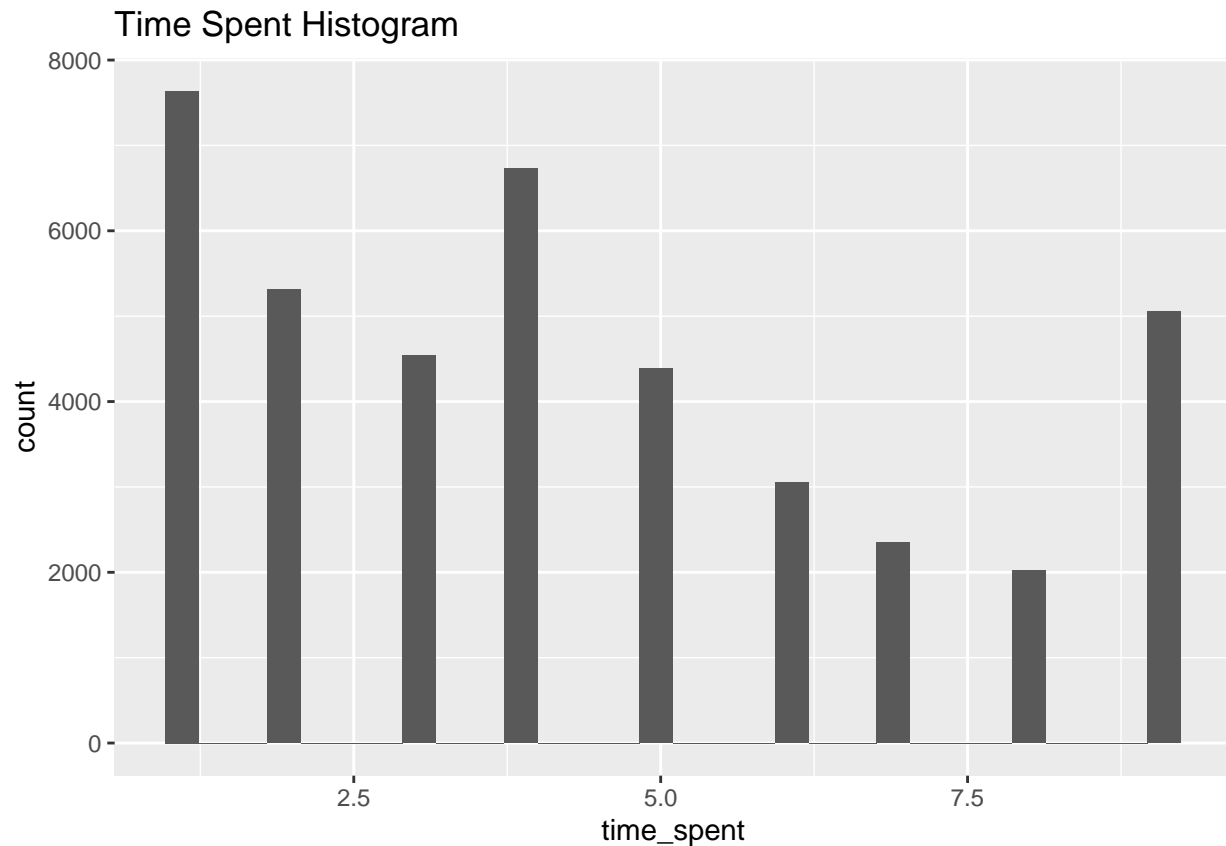


### Exercise3

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

```
entry_exit <- Full_Data %>% group_by(idmen) %>% summarize(entry = min(year), exit = max(year)) %>%
  mutate(time_spent = (exit - entry + 1))

ggplot(entry_exit, aes(x=time_spent)) + geom_histogram(bins = 30) + ggtitle("Time Spent Histogram")
```



Based 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:

```
hh_year = Full_Data %>% mutate(move_in_check = ifelse(datent == year, 1, 0))

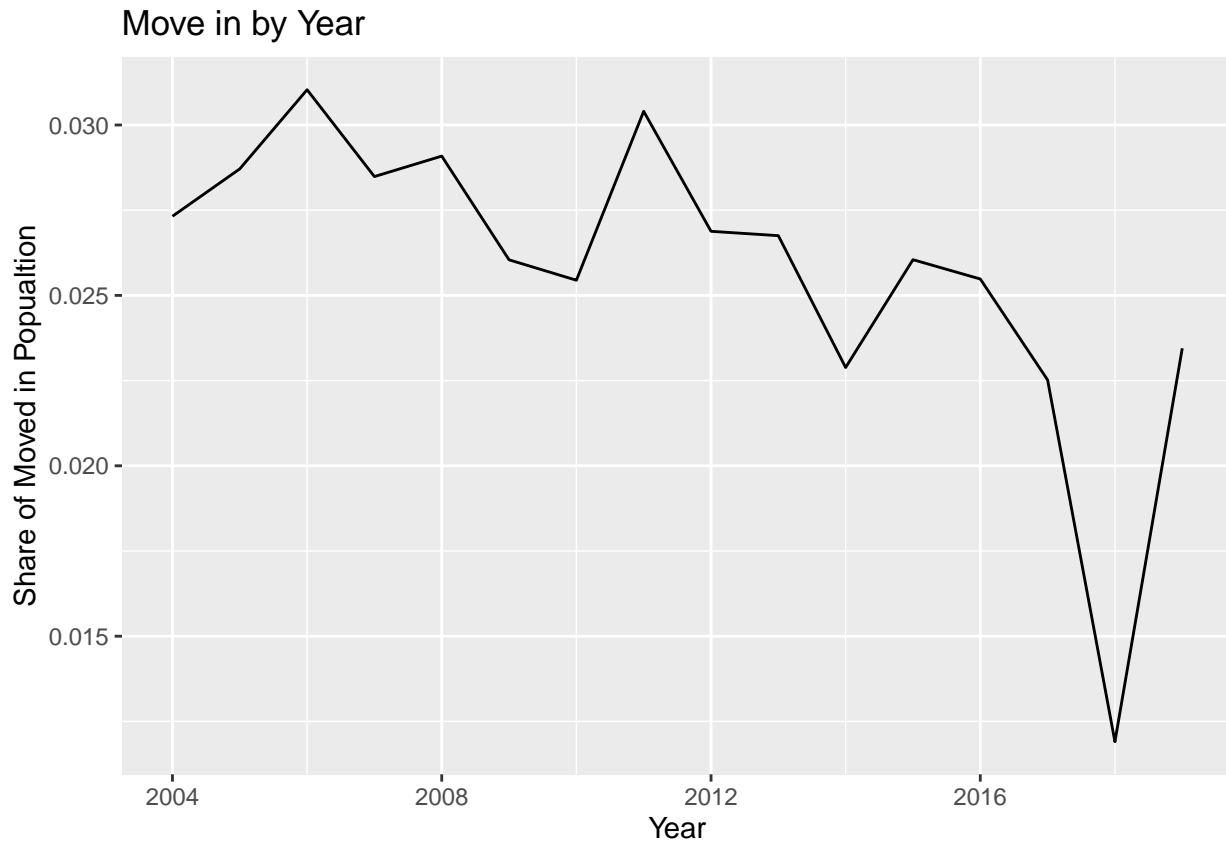
head(hh_year, 10)
```

##	X.x	idmen	year	datent	myear	mstatus	move	location	X.y
## 1	1	1.20001e+15	2004	2000	2000	Single	NA	Paris	1
## 2	2	1.20001e+15	2004	2001	2001	Single Parent	NA	Paris	2
## 3	2	1.20001e+15	2004	2001	2001	Single Parent	NA	Paris	3
## 4	3	1.20001e+15	2004	2000	2000	Couple, No kids	NA	Paris	4
## 5	3	1.20001e+15	2004	2000	2000	Couple, No kids	NA	Paris	5
## 6	4	1.20001e+15	2004	1957	1957	Single	NA	Paris	6
## 7	5	1.20001e+15	2004	2001	2001	Couple, No kids	NA	Paris	7
## 8	5	1.20001e+15	2004	2001	2001	Couple, No kids	NA	Paris	8
## 9	6	1.20001e+15	2004	1990	1990	Single Parent	NA	Paris	9
## 10	6	1.20001e+15	2004	1990	1990	Single Parent	NA	Paris	10

```
##           idind empstat respondent profession gender age  wage move_in_check
## 1  1.120001e+18 Employed          1         67  Male  31 19187             0
## 2  1.120001e+18 Employed          1         56 Female  30 11586             0
## 3  1.120001e+18 Inactive          0          9  Female   9   NA             0
## 4  1.120001e+18 Employed          1         38  Male  31 44656             0
## 5  1.120001e+18 Employed          0         45 Female  27 20413             0
## 6  1.120001e+18 Retired           1          89  Female  89    0             0
## 7  1.120001e+18 Employed          1         34  Male  36 30702             0
## 8  1.120001e+18 Employed          0         42 Female  34 24650             0
## 9  1.120001e+18 Employed          1         46 Female  40 29604             0
## 10 1.120001e+18 Inactive          0          15  Female  15   NA             0
```

```
hh_year_ratio = hh_year %>% group_by(year) %>%
  summarise(total_c = n(), move_c = sum(move_in_check, na.rm = T), m_ratio = move_c/total_c)

ggplot(hh_year_ratio, aes(y=m_ratio, x=year)) + geom_line() +
  ggtitle("Move in by Year") + xlab("Year") + ylab("Share of Moved in Popualtion")
```



Based on myear and move, identify whether or not 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:

```
summary(Full_Data)
```

```
##           X.x           idmen           year           datent
## Min.      :    1   Min.    :1.200e+15   Min.    :2004   Min.    :1912
## 1st Qu.: 2676   1st Qu.:2.003e+15   1st Qu.:2008   1st Qu.:1990
## Median : 5375   Median :2.311e+15   Median :2012   Median :2001
## Mean    : 5418   Mean    :2.349e+15   Mean    :2012   Mean    :1997
## 3rd Qu.: 8095   3rd Qu.:2.710e+15   3rd Qu.:2015   3rd Qu.:2007
```

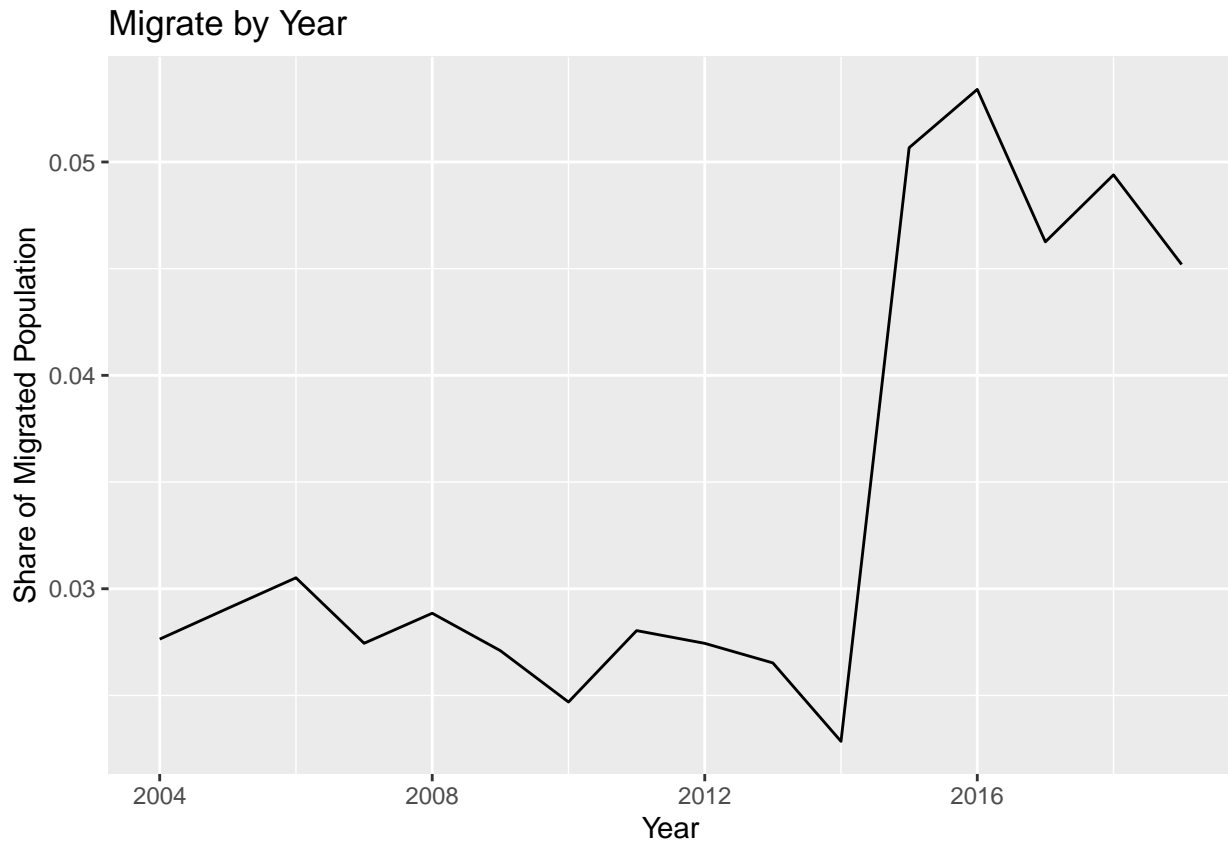
```
## Max. :11999 Max. :3.413e+15 Max. :2019 Max. :2019
## NA's :245
## myear mstatus move location
## Min. :1922 Length:413501 Min. :1.00 Length:413501
## 1st Qu.:1989 Class :character 1st Qu.:1.00 Class :character
## Median :2000 Mode :character Median :1.00 Mode :character
## Mean :1996 Mean :1.06
## 3rd Qu.:2005 3rd Qu.:1.00
## Max. :2014 Max. :2.00
## NA's :136321 NA's :308876
## X.y idind empstat respondent
## Min. : 1 Min. :1.200e+17 Length:413501 Min. :0.0000
## 1st Qu.: 6462 1st Qu.:1.200e+18 Class :character 1st Qu.:0.0000
## Median :12923 Median :1.240e+18 Mode :character Median :0.0000
## Mean :12960 Mean :1.272e+18 Mean :0.4063
## 3rd Qu.:19384 3rd Qu.:1.281e+18 3rd Qu.:1.0000
## Max. :28534 Max. :2.331e+18 Max. :1.0000
##
## profession gender age wage
## Length:413501 Length:413501 Min. : -1.00 Min. : 0
## Class :character Class :character 1st Qu.: 19.00 1st Qu.: 0
## Mode :character Mode :character Median : 41.00 Median : 3880
## Mean : 40.35 Mean : 13693
## 3rd Qu.: 59.00 3rd Qu.: 23357
## Max. :102.00 Max. :1747898
## NA's :2 NA's :85183
```

```
migrate_year = Full_Data %>%
  mutate(mi_check = ifelse(is.na(move) == F & move ==2, 1, ifelse(year == myear, 1 , 0)))
head(migrate_year, 10)
```

```
## X.x idmen year datent myear mstatus move location X.y
## 1 1 1.20001e+15 2004 2000 2000 Single NA Paris 1
## 2 2 1.20001e+15 2004 2001 2001 Single Parent NA Paris 2
## 3 2 1.20001e+15 2004 2001 2001 Single Parent NA Paris 3
## 4 3 1.20001e+15 2004 2000 2000 Couple, No kids NA Paris 4
## 5 3 1.20001e+15 2004 2000 2000 Couple, No kids NA Paris 5
## 6 4 1.20001e+15 2004 1957 1957 Single NA Paris 6
## 7 5 1.20001e+15 2004 2001 2001 Couple, No kids NA Paris 7
## 8 5 1.20001e+15 2004 2001 2001 Couple, No kids NA Paris 8
## 9 6 1.20001e+15 2004 1990 1990 Single Parent NA Paris 9
## 10 6 1.20001e+15 2004 1990 1990 Single Parent NA Paris 10
## idind empstat respondent profession gender age wage mi_check
## 1 1.120001e+18 Employed 1 67 Male 31 19187 0
## 2 1.120001e+18 Employed 1 56 Female 30 11586 0
## 3 1.120001e+18 Inactive 0 Female 9 NA 0
## 4 1.120001e+18 Employed 1 38 Male 31 44656 0
## 5 1.120001e+18 Employed 0 45 Female 27 20413 0
## 6 1.120001e+18 Retired 1 Female 89 0 0
## 7 1.120001e+18 Employed 1 34 Male 36 30702 0
## 8 1.120001e+18 Employed 0 42 Female 34 24650 0
## 9 1.120001e+18 Employed 1 46 Female 40 29604 0
## 10 1.120001e+18 Inactive 0 Female 15 NA 0
```

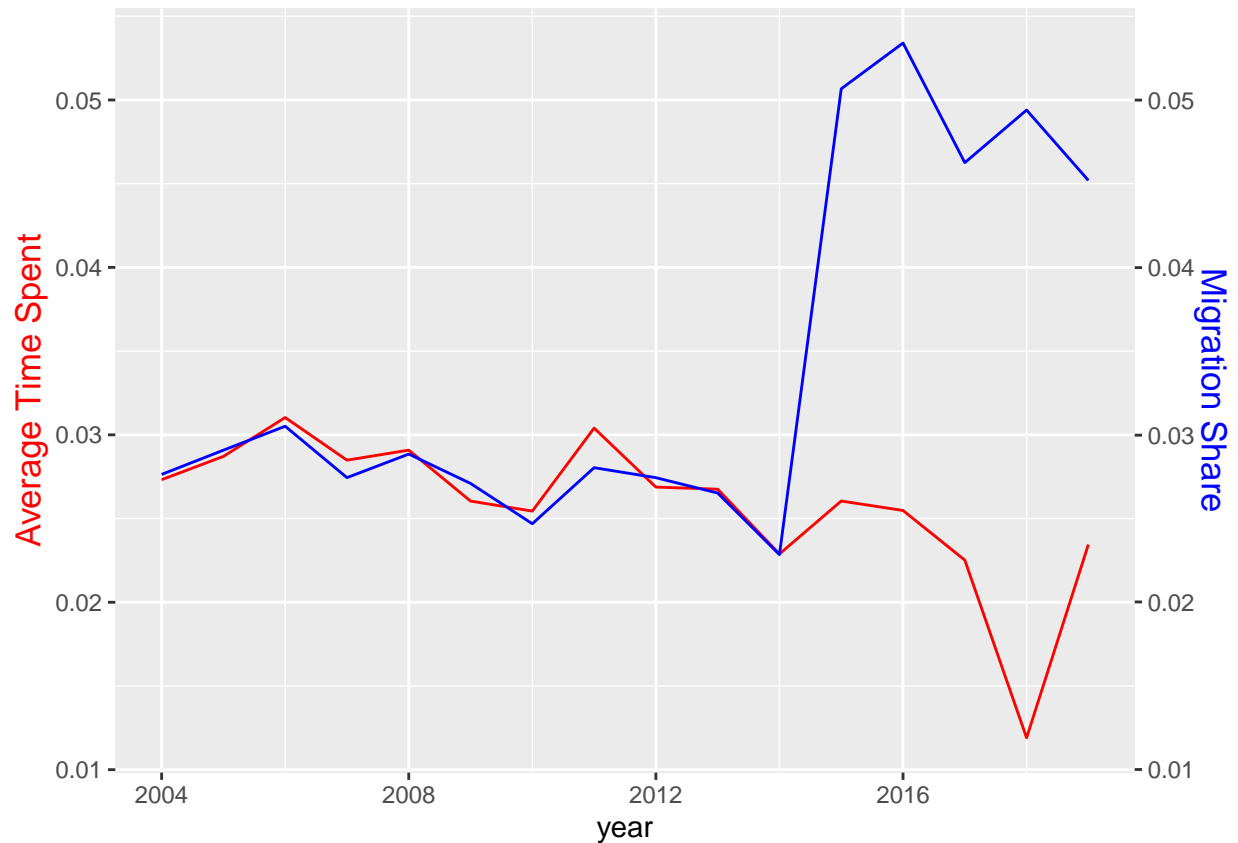
```
migrate_year_share = migrate_year %>% group_by(year) %>%
  summarise(year_count = n(), mi_count = sum(mi_check, na.rm = TRUE)) %>%
  mutate(mi_share = mi_count/year_count)
```

```
ggplot(migrate_year_share, aes(y=mi_share,x=year)) + geom_line() +
  ggtitle("Migrate by Year") + xlab("Year") + ylab("Share of Migrated Population")
```



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

```
ggplot(by_plot, aes(x=year)) +
  geom_line(aes(y=m_ratio), color="red") + geom_line(aes(y=mi_share), color="blue") +
  scale_y_continuous(name = "Average Time Spent", sec.axis = sec_axis(trans=~.*1, name="Migration Share"))
  theme(
    axis.title.y = element_text(color = "red", size=13),
    axis.title.y.right = element_text(color = "blue", size=13)
  )
```



I prefer the time spent graph since it is straightforward to understand. Moreover, migration share data has many missing values. In fact, after 2014, we do not have data on myear, and this inconsistency in measurement partially contributes to the jump after 2015.

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

```
prof_emp_c = migrate_year %>% filter(mi_check == 1) %>%
  group_by(idmen,profession,empstat) %>% mutate(check = n()) %>%
  mutate(ifelse(check != 1, 1,0)) %>% filter(check == 1)

length(prof_emp_c$idmen)
```

```
## [1] 7239
```

#### Exercise4

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.

```
years <- 2004:2018
ratio <- seq(2005, 2019, by = 1)
attrition <- seq(2005, 2019, by = 1)
ind_c <- seq(2005, 2019, by = 1)
n = 1
for (i in years) {
  j <- i + 1
  attrition[n] <- length(setdiff(Full_Data[Full_Data$year == i,]$idind,
                                Full_Data[Full_Data$year == j,]$idind))
  ind_c[n] <- length(Full_Data[Full_Data$year == i,]$idind)
  ratio[n] <- attrition[n]/ind_c[n]
  n = n + 1
}

attrition_ratio <- data.frame(year = 2005:2019, attrition_ratio = ratio)

attrition_ratio
```

##	year	attrition_ratio
## 1	2005	0.05640354
## 2	2006	0.08246359
## 3	2007	0.07213312
## 4	2008	0.09101787
## 5	2009	0.08428067
## 6	2010	0.07512397
## 7	2011	0.08104644
## 8	2012	0.07177422
## 9	2013	0.10745076
## 10	2014	0.09239935
## 11	2015	0.09448613
## 12	2016	0.09281639
## 13	2017	0.10586558
## 14	2018	0.10617274
## 15	2019	0.10664831