R Markdown

```r
library(pacman)
p_load(tidyverse,knitr)


for(i in 2004:2019){
  datth<-read.csv(paste('Data/dathh',i,'.csv',sep = ""))
  datth$idmen<-as.character(datth$idmen)
  datth$mstatus<-as.character(datth$mstatus)
  assign(paste('datth_',i,sep = ""),datth)

  datind<-read.csv(paste('Data/datind',i,'.csv',sep = ""))
  datind$idind<-as.character(datind$idind)
  datind$idmen<-as.character(datind$idmen)
  assign(paste('datind_',i,sep = ""),datind)

}
```

Exercise 1 Basic Statistics

```r
#### 1. Number of households surveyed in 2007
datth_2007 %>% select(idmen) %>% summarise(number=n())
```

```
##   number
## 1  10498
```

```r
#### 2. Number of households with marital status "Couple with kids" in 2005
datth_2005 %>% filter(mstatus=='Couple, with Kids') %>% summarise(number=n())
```

```
##   number
## 1   3374
```

```r
#### 3. Number of individuals surveyed in 2008.
datind_2008 %>% select(idind) %>% summarise(number=n())
```

```
##   number
## 1  25510
```

```r
#### 4. Number of individuals aged between 25 and 35 in 2016.
datind_2016 %>% filter(age>=25,age<=35) %>% summarise(number=n())
```

```
##   number
## 1   2765
```

```r
#### 5. Cross-table gender/profession in 2009.
table(datind_2009$gender,datind_2009$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
```

6. Distribution of wages in 2005 and 2019. Report the mean, the standard deviation, the inter-decile ratio D9/D1 and the Gini coefficient They are discrete distribution.

```
#mean 2005
mean(datind_2005$wage,na.rm = TRUE)
```

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

```
#mean 2019
mean(datind_2019$wage,na.rm = TRUE)
```

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

```
#sd 2005
sd(datind_2005$wage,na.rm = TRUE)
```

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

```
#sd 2019
sd(datind_2019$wage,na.rm = TRUE)
```

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

```
#D9/D1
quantile(datind_2019$wage,na.rm = TRUE,0.9,names=F)/quantile(datind_2005$wage,na.rm = TRUE,0.9,names=F)
```

```
## [1] 1.245099
```

```
#the Gini coefficient 2005
getGini<-function(v){
  v<-na.omit(v)
  n <- length(v)
  s_v <- sort(v)
  gini <- 1 - ((2/(n+1)) * sum(cumsum(s_v))*(sum(s_v))^(-1))
  return(gini)
}
getGini(datind_2005$wage)
```

```
## [1] 0.6671299
```

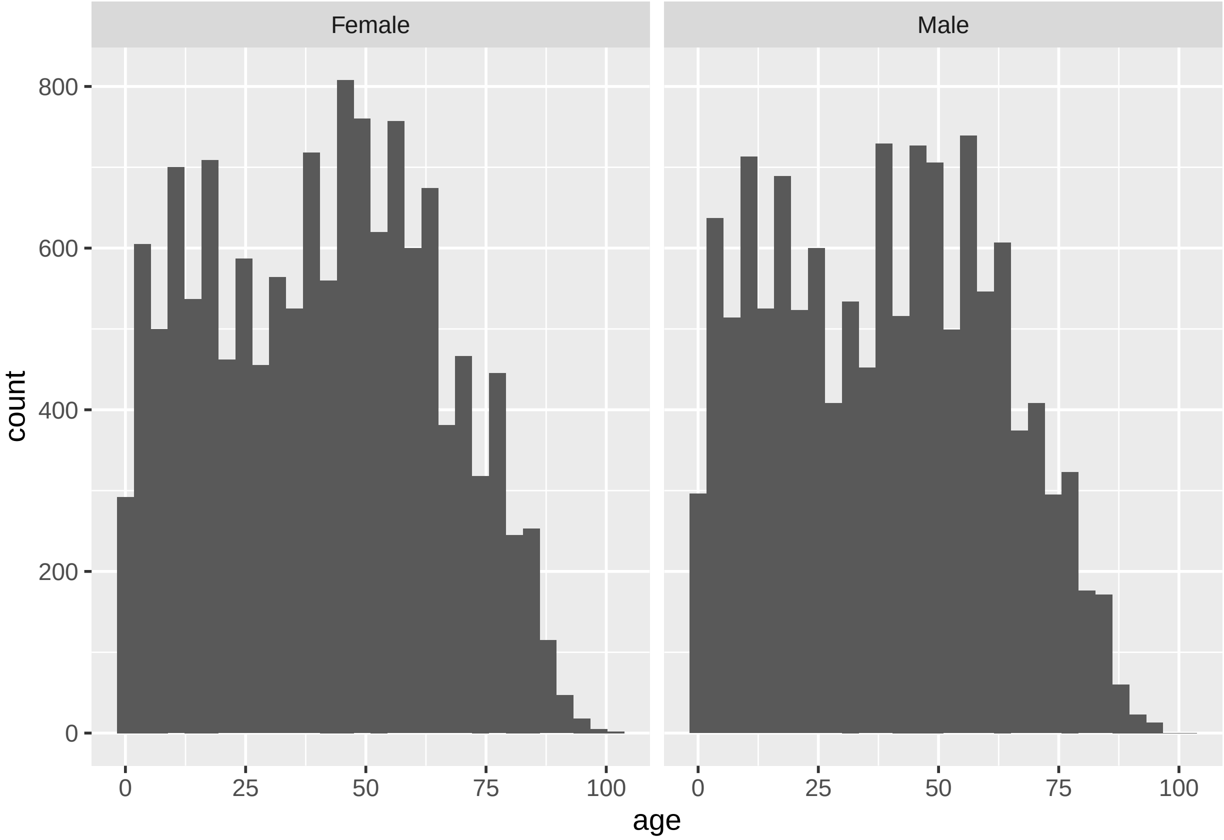
```
#the Gini coefficient 2019
getGini(datind_2019$wage)
```

```
## [1] 0.665499
```

7. Distribution of age in 2010. Plot an histogram. Is there any difference between men and women? It is a discrete distribution.From the histogram,we can see the difference between men and women is that the count number of women bigger than men about age at 50.

```
datind_2010 %>% group_by(gender,age) %>% ggplot(aes(x=age))+geom_histogram()+facet_grid(~gender)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



```
#### 8.  Number of individuals in Paris in 2011.
datind_2011 %>% inner_join(datth_2011,by='idmen')%>%
  filter(location=='Paris') %>% summarise(number=n())
```

```
##   number
## 1   3514
```

Exercise 2 Merge Datasets

```
#Read all individual datasets from 2004 to 2019. Append all these datasets.
datindall<-rbind(datind_2004,datind_2005,datind_2006,
                 datind_2007,datind_2008,datind_2009,
                 datind_2010,datind_2011,datind_2012,
                 datind_2013,datind_2014,datind_2015,
                 datind_2016,datind_2017,datind_2018,
                 datind_2019
                 )
```

```
#Read all household datasets from 2004 to 2019. Append all these datasets.
datthall<-rbind(datth_2004,datth_2005,datth_2006,
                 datth_2007,datth_2008,datth_2009,
                 datth_2010,datth_2011,datth_2012,
                 datth_2013,datth_2014,datth_2015,
                 datth_2016,datth_2017,datth_2018,
                 datth_2019
                 )
```

```
#List the variables that are simultaneously present in the individual and household datasets
common_variables<-c()
for(i in 1:length(names(datindall))){
  tmp<-names(datindall)[i]
  for(j in 1:length(names(datthall))){
    if(tmp==names(datthall)[j]){
      common_variables<-c(common_variables,tmp)
    }
  }
}
print(common_variables)
```

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

```
#Merge the appended individual and household datasets
merge_all<-datindall %>% inner_join(datthall,by=c('idmen','year'))
```

```
#Number of households in which there are more than four family members
bigger_four<- merge_all %>% group_by(idmen,idind) %>%
  summarise(number=n()) %>% filter(number>4)
```

```
## `summarise()` has grouped output by 'idmen'. You can override using the `.groups` argument.
```

```
nrow(bigger_four)
```

```
## [1] 27604
```

```
#Number of households in which at least one member is unemployed
at_leat_one_unemployed<- merge_all %>% group_by(idmen,empstat) %>%
  filter(empstat=='Unemployed') %>% summarise(number=n()) %>% filter(number>=1)
```

```
## `summarise()` has grouped output by 'idmen'. You can override using the `.groups` argument.
```

```
nrow(at_leat_one_unemployed)
```

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

```
#Number of households in which at least two members are of the same profession
at_leat_two_profession<- merge_all %>% filter(profession!='') %>%
  group_by(idmen,profession) %>% summarise(number=n()) %>% filter(number>=2)
```

```
## `summarise()` has grouped output by 'idmen'. You can override using the `.groups` argument.
```

```
nrow(at_leat_two_profession)
```

```
## [1] 35307
```

```
#Number of individuals in the panel that are from household-Couple with kids
household_Couple <-merge_all %>% group_by(idmen,idind,mstatus) %>%
  filter(mstatus=='Couple, with Kids') %>% summarise(number=n())
```

```
## `summarise()` has grouped output by 'idmen', 'idind'. You can override using the `.groups` argument.
```

```
nrow(household_Couple)
```

```
## [1] 15992
```

```
#Number of individuals in the panel that are from Paris.
merge_all %>% filter(location=='Paris') %>% summarise(number=n())
```

```
##   number
## 1  51904
```

```
#Find the household with the most number of family members. Report its idmen
most_number<-merge_all %>% group_by(idmen,idind) %>% summarise(number=n()) %>% arrange(desc(number)) %>%
```

```
## `summarise()` has grouped output by 'idmen'. You can override using the `.groups` argument.
```

```
most_number
```

```
## # A tibble: 1 x 3
## # Groups:   idmen [1]
##   idmen            idind                number
##   <chr>            <chr>                 <int>
## 1 2202243098040100 1220224309804009984     81
```

```
most_number$idmen
```

```
## [1] "2202243098040100"
```

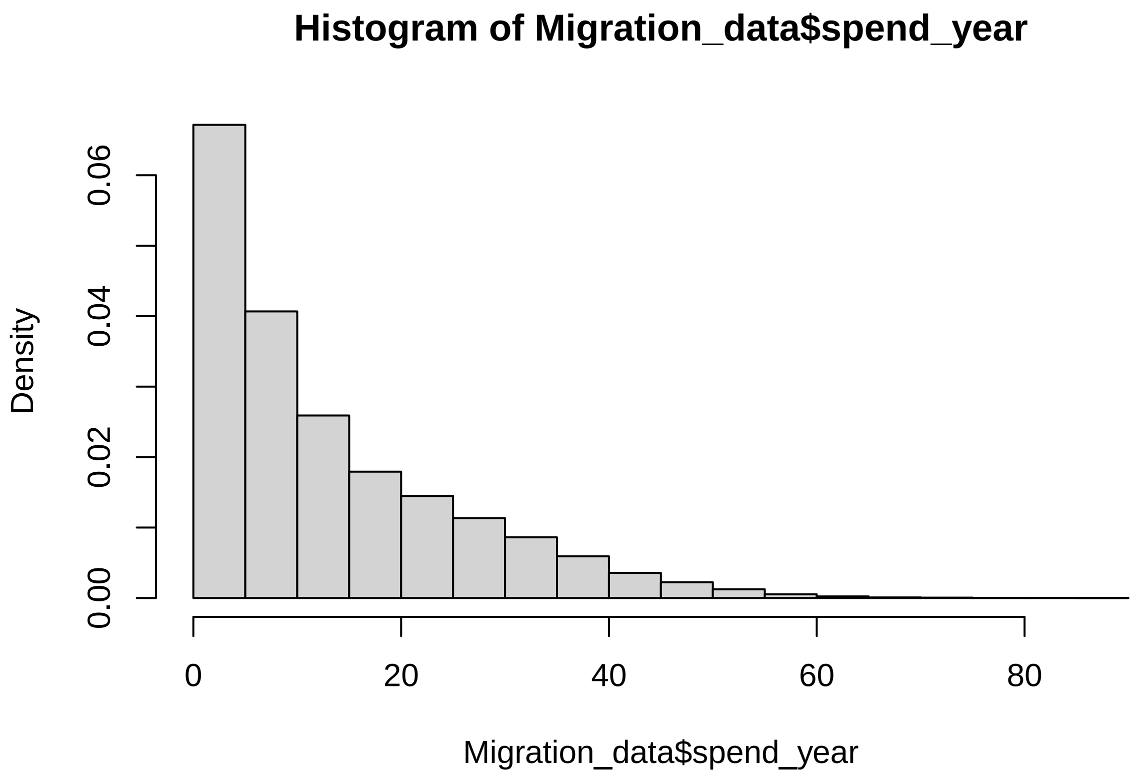
The most number of family member household's idmen is 2202243098040100.

```
#Number of households present in 2010 and 2011.
nrow(merge_all %>% group_by(idmen) %>% filter(year>=2010,year<=2021) %>% summarise(number=n()))
```

```
## [1] 30891
```

Exercise 3 Migration

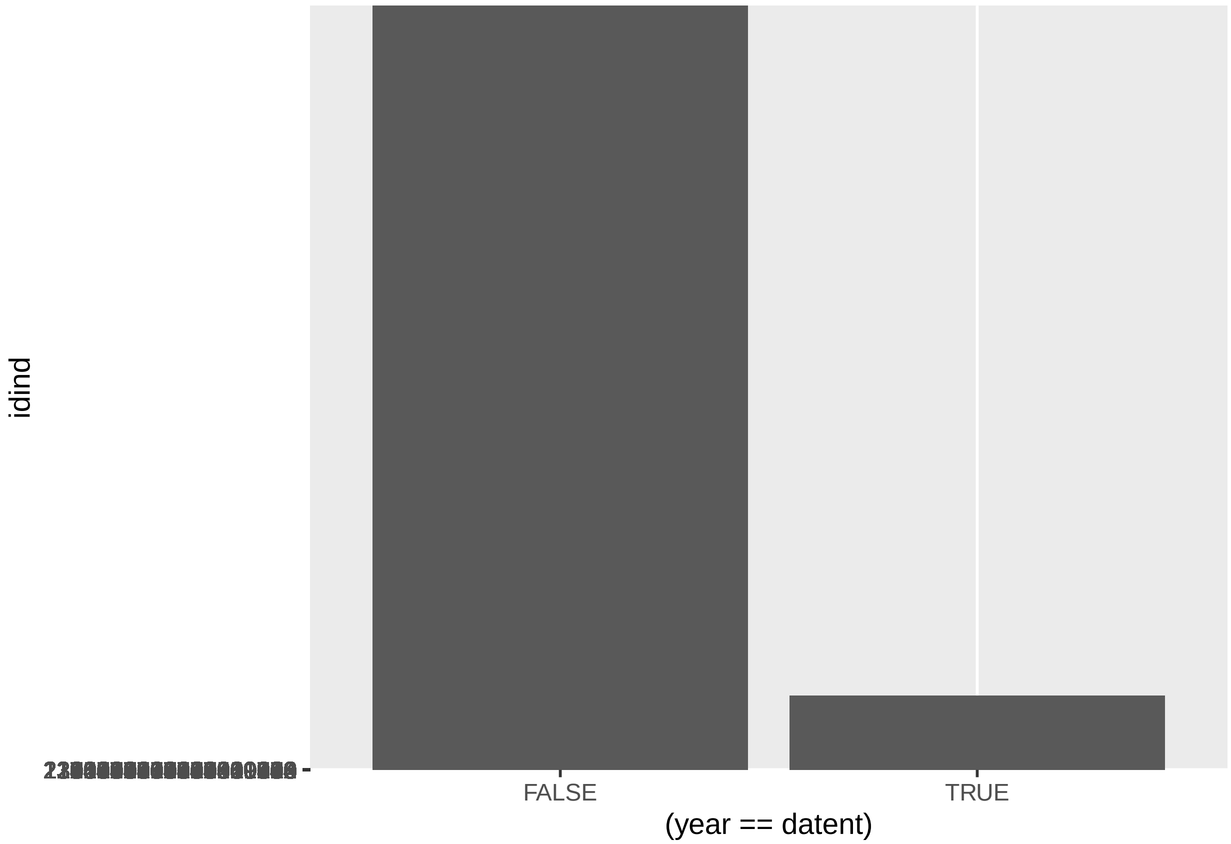
```
# Find out the year each household enters and exit the panel. Report the distribution of the time spent
#in the survey for each household.
Migration_data<-merge_all %>% filter(!is.na(myear))
Migration_data <- Migration_data %>% mutate(spend_year=year-myear)
hist(Migration_data$spend_year,freq = F)
```



```
#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 ac
merge_all %>% filter(year==datent) %>% head(10)
```

```
##    X.x               idind             idmen year    empstat respondent
## 1   92 1120049301027010048 1200493010270100 2004 Unemployed          1
## 2   93 1120049301027010048 1200493010270100 2004   Employed          0
## 3   94 1120049301027010048 1200493010270100 2004   Inactive          0
## 4   95 1120049301027010048 1200493010270100 2004   Inactive          0
## 5  193 1120074202054009984 1200742020540100 2004   Employed          1
## 6  194 1120074202054009984 1200742020540100 2004   Employed          0
## 7  275 1120089601262009984 1200896012620100 2004   Employed          1
## 8  313 1120089808968009984 1200898089680100 2004    Retired          1
## 9  314 1120089808968009984 1200898089680100 2004    Retired          0
## 10 390 1120138606786009984 1201386067860100 2004   Employed          1
##    profession gender age  wage X.y datent myear           mstatus move
## 1             Female  36     0  43   2004  2004 Couple, with Kids   NA
## 2           68   Male  31     0  43   2004  2004 Couple, with Kids   NA
## 3             Female   8    NA  43   2004  2004 Couple, with Kids   NA
## 4             Female   8    NA  43   2004  2004 Couple, with Kids   NA
## 5           67   Male  29 16106  85   2004  2004   Couple, No kids   NA
## 6           56 Female  23 15180  85   2004  2004   Couple, No kids   NA
## 7           55   Male  36 31783 115   2004  2004            Single   NA
## 8              Female  55 24258 129   2004  1977   Couple, No kids   NA
## 9                Male  56  7453 129   2004  1977   Couple, No kids   NA
## 10          43 Female  44 27051 164   2004  2004            Single   NA
##                location
## 1                 Rural
## 2                 Rural
## 3                 Rural
## 4                 Rural
## 5  Urban 10000 to 19999
## 6  Urban 10000 to 19999
## 7                 Paris
## 8                 Rural
## 9                 Rural
## 10                Paris
```

```
merge_all %>% filter(!is.na(year),!is.na(datent),!is.na(idind))%>%
  ggplot(aes(x=(year==datent),y=idind))+geom_histogram(stat = "identity")
```



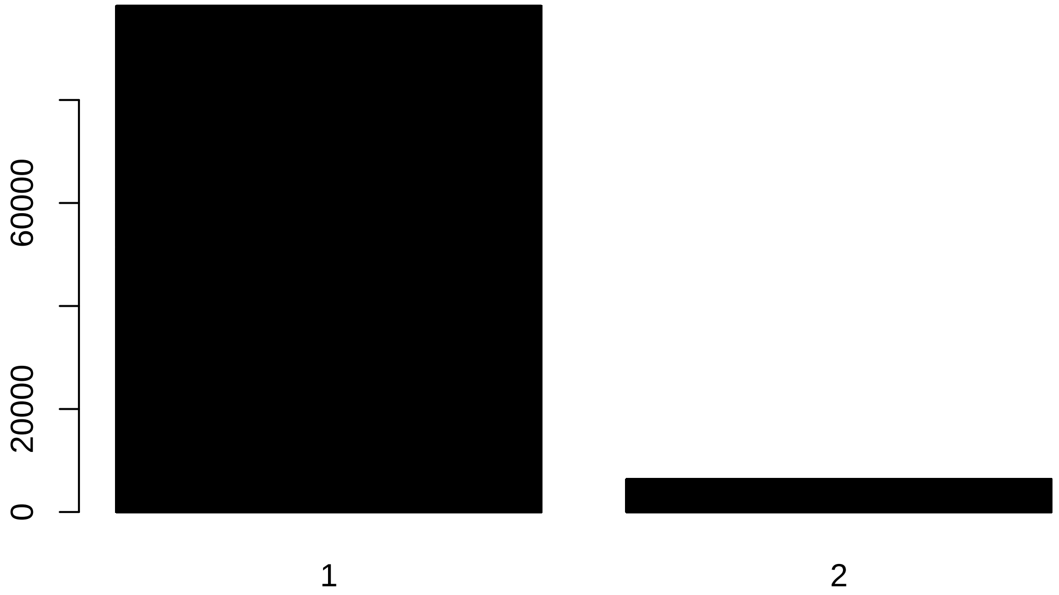
```
#Based on myear and move, identify whether or not household migrated at the year of survey. Report
#the first 10 rows of your result and plot the share of individuals in that situation across years.
# move
merge_all %>% filter(!is.na(move)) %>% head(10)
```

```
##    X.x               idind            idmen year    empstat respondent
## 1    3 1240546407362010112 2405464073620100 2015    Retired          1
## 2    4 1240546407362010112 2405464073620100 2015    Retired          0
## 3    8 1240546403254010112 2405464032540100 2015   Employed          1
## 4    9 1240546403254010112 2405464032540101 2015   Employed          0
## 5   10 1260546410880009984 2605464108800100 2015   Employed          1
## 6   11 2260546410880009984 2605464108800100 2015 Unemployed          0
## 7   12 1260546410880009984 2605464108800100 2015   Inactive          0
## 8   13 1260546410880009984 2605464108800100 2015   Inactive          0
## 9   18 1280546401760009984 2805464017600100 2015 Unemployed          1
## 10  21 1260546401575010048 2605464015750100 2015   Employed          1
##    profession gender age  wage  X.y datent myear           mstatus move
## 1        <NA>   Male  72     0 1544   1982    NA   Couple, No kids    1
## 2        <NA> Female  67     0 1544   1982    NA   Couple, No kids    1
## 3          38   Male  27 51770 1545   1998    NA            Single    1
## 4          37 Female  34 62497 1546   2011    NA            Single    1
## 5          37 Female  29 40363 3439   2014    NA Couple, with Kids    2
## 6        <NA>   Male  30 20900 3439   2014    NA Couple, with Kids    2
## 7        <NA>   Male   1    NA 3439   2014    NA Couple, with Kids    2
## 8        <NA>   Male   0    NA 3439   2014    NA Couple, with Kids    2
## 9        <NA> Female  58     0 6250   2006    NA            Single    1
## 10         38 Female  36 46114 3440   2011    NA Couple, with Kids    1
##                    location
## 1                     Paris
## 2                     Paris
## 3                     Paris
## 4                     Paris
## 5  Urban 200000 to 1999999
## 6  Urban 200000 to 1999999
## 7  Urban 200000 to 1999999
## 8  Urban 200000 to 1999999
## 9                     Paris
## 10                    Paris
```

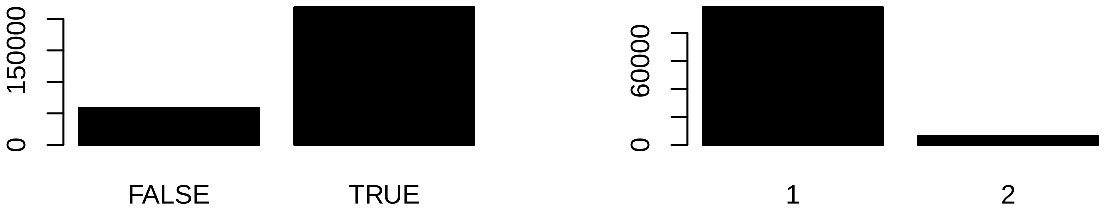
```
# not move
merge_all %>% filter(myear<year) %>% filter(is.na(move)) %>% head(10)
```

```
##    X.x               idind            idmen year  empstat respondent profession
## 1    1 1120001001293010048 1200010012930100 2004 Employed          1         67
## 2    2 1120001004058009984 1200010040580100 2004 Employed          1         56
## 3    3 1120001004058009984 1200010040580100 2004 Inactive          0
## 4    4 1120001006663010048 1200010066630100 2004 Employed          1         38
## 5    5 1120001006663010048 1200010066630100 2004 Employed          0         45
## 6    6 1120001008245010048 1200010082450100 2004  Retired          1
## 7    7 1120001008644009984 1200010086440100 2004 Employed          1         34
## 8    8 1120001008644009984 1200010086440100 2004 Employed          0         42
## 9    9 1120001010299010048 1200010102990100 2004 Employed          1         46
## 10  10 1120001010299010048 1200010102990100 2004 Inactive          0
##    gender age  wage X.y datent myear         mstatus move location
## 1    Male  31 19187   1   2000  2000          Single   NA    Paris
## 2  Female  30 11586   2   2001  2001   Single Parent   NA    Paris
## 3  Female   9    NA   2   2001  2001   Single Parent   NA    Paris
## 4    Male  31 44656   3   2000  2000 Couple, No kids   NA    Paris
## 5  Female  27 20413   3   2000  2000 Couple, No kids   NA    Paris
## 6  Female  89     0   4   1957  1957          Single   NA    Paris
## 7    Male  36 30702   5   2001  2001 Couple, No kids   NA    Paris
## 8  Female  34 24650   5   2001  2001 Couple, No kids   NA    Paris
## 9  Female  40 29604   6   1990  1990   Single Parent   NA    Paris
## 10 Female  15    NA   6   1990  1990   Single Parent   NA    Paris
```

```
barplot(table(merge_all$idind,merge_all$move))
```



```
# Mix the two plots you created above in one graph, clearly label the graph. Do you prefer one method
#over the other? Justify
par(mfrow=c(2,2))
barplot(table(merge_all$idind,(merge_all$datent==merge_all$myear)))
barplot(table(merge_all$idind,merge_all$move))
```



We prefer the last method,because the method can see the two plots in contrast.

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

nrow(merge_all %>% filter(!is.na(move),is.na(profession)) %>% group_by(idmen,idind)  %>% summarise(numbe
```

```
## `summarise()` has grouped output by 'idmen'. You can override using the `.groups` argument.
```

```
## [1] 14837
```

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.
#Hint: Construct a year of entry and exit for each individual.


attrition_f<-function(year){

  temp<-assign(paste('attribution_',year,sep=''),0)


  datind<-read.csv(paste('Data/datind',year-1,'.csv',sep = ""))
  datind$idind<-as.character(datind$idind)
  datind$idmen<-as.character(datind$idmen)
  last_year<-assign(paste('datind_',year-1,sep = ""),datind)

  datind<-read.csv(paste('Data/datind',year,'.csv',sep = ""))
  datind$idind<-as.character(datind$idind)
  datind$idmen<-as.character(datind$idmen)
  this_year<-assign(paste('datind_',year,sep = ""),datind)

  for(i in 1:nrow(last_year)){

    if(last_year$idind[i]  %in% this_year$idind){
      next
    }else{
      temp<-temp+1
    }


  }
  return(temp)
}
#2005
attrition_f(2005)
```

```
## [1] 2719
```

```
#2006
attrition_f(2006)
```

```
## [1] 4497
```

```
#2007
attrition_f(2007)
```

```
## [1] 4107
```

```
#2008
attrition_f(2008)
```

```
## [1] 5461
```

```
#2009
attrition_f(2009)
```

```
## [1] 4818
```

```
#2010
attrition_f(2010)
```

```
## [1] 4309
```

```
#2011
attrition_f(2011)
```

```
## [1] 4665
```

```
#2012
attrition_f(2012)
```

```
## [1] 4141
```

```
#2013
attrition_f(2013)
```

```
## [1] 6715
```

```
#2014
attrition_f(2014)
```

```
## [1] 5322
```

```
#2015
attrition_f(2015)
```

```
## [1] 5421
```

```
#2016
attrition_f(2016)
```

```
## [1] 5369
```

```
#2017
attrition_f(2017)
```

```
## [1] 6234
```

```
#2018
attrition_f(2018)
```

```
## [1] 5775
```

```
#2019
attrition_f(2019)
```

```
## [1] 5593
```