CS544: Data Visualization Project - Adult Income Analysis

1. Introduction

Adult Income Analysis project is a data visualization project built on the USA 1994 census dataset.

1.1 Project Objective

The objective of this project is to analyse the dataset using data visualization techniques.

1.2 Libraries Used

For this project some additional libraries are used:

- **ggplot2**
- **tidyverse**
- **plotly**
- **kableExtra**

Purpose of these libraries is to generate advanced visual graphs and data formatting.

1.3 Dataset Overview

Dataset for Adult Income Analysis is data extracted from the 1994 census bureau database. Population in dataset is classified into two income categories, 1). >50K and 2). <= 50K. Along with information about income categories, the dataset consists of import information like age, workclass, work per hour, education etc.

1.4 Dataset structure information

The dataset consists of 15 attributes and the structure is as below

```
#Loading Data from CSV file
Original_Ds <- read.csv("/Users/jyotivashishth/Desktop/adult.csv" , header = TR
  UE)
glimpse(Original_Ds)
```

```
## Rows: 32,561
## Columns: 15
## $ age            <int> 90, 82, 66, 54, 41, 34, 38, 74, 68, 41, 45, 38, 52, 3
2…
## $ workclass      <fct> ?, Private, ?, Private, Private, Private, Private, St
a…
## $ fnlwgt         <int> 77053, 132870, 186061, 140359, 264663, 216864, 15060
1,…
## $ education      <fct> HS-grad, HS-grad, Some-college, 7th-8th, Some-colleg
e,…
## $ education.num  <int> 9, 9, 10, 4, 10, 9, 6, 16, 9, 10, 16, 15, 13, 14, 16,
…
## $ marital.status <fct> Widowed, Widowed, Widowed, Divorced, Separated, Divor
c…
## $ occupation     <fct> ?, Exec-managerial, ?, Machine-op-inspct, Prof-specia
l…
## $ relationship   <fct> Not-in-family, Not-in-family, Unmarried, Unmarried, O
w…
## $ race           <fct> White, White, Black, White, White, White, White, Whit
e…
## $ sex            <fct> Female, Female, Female, Female, Female, Female, Male,
…
## $ capital.gain   <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
…
## $ capital.loss   <int> 4356, 4356, 4356, 3900, 3900, 3770, 3770, 3683, 3683,
…
## $ hours.per.week <int> 40, 18, 40, 40, 40, 45, 40, 20, 40, 60, 35, 45, 20, 5
5…
## $ native.country <fct> United-States, United-States, United-States, United-S
t…
## $ income         <fct> <=50K, <=50K, <=50K, <=50K, <=50K, <=50K, <=50K, >50
K,…
```

2 Initial Pre-processing

As a part of pre-processing the dataset was explored for followings:

- Duplicate rows
- Missing data

2.1 Duplicate rows

Check for duplicate rows was carried out in two steps

Step 1: Identifying number of duplicate rows.

Step 2: Remove the duplicate rows from the original dataset.

```
##Initialise the vector
record_info <- vector()
record_info[1] <- nrow(Original_Ds)
##check for Duplicate  rows
Ds_data <- Original_Ds[duplicated(Original_Ds),]
cat("The Data set has " , nrow(Original_Ds) , " Rows")
```

```
## The Data set has  32561  Rows
```

```
#Display number of duplicate row
record_info[2] <- nrow(Ds_data)
##show the duplicate Data
cat("The Data set has " , nrow(Ds_data) , "Duplicate Rows")
```

```
## The Data set has  24 Duplicate Rows
```

```
##unique Records
DataCensus <- Original_Ds[!duplicated(Original_Ds),]
cat("After Removing the Duplicate Rows Data set has ",nrow(DataCensus),"Unique
   Rows")
```

```
## After Removing the Duplicate Rows Data set has  32537 Unique Rows
```

Overview of Data:

```
## Warning in kableExtra::kable_styling(., full_width = FALSE, position = "lef
t", :
## Please specify format in kable. kableExtra can customize either HTML or LaTe
X
## outputs. See https://haozhu233.github.io/kableExtra/ for details.
```

Category	Count
Total Rows	32,561
Duplicate Rows	24
Unique Rows	32,537

2.2 Replace the missing data(? in our case)

Here is a glimpse of dataset

```
head(DataCensus)
```

```
##   age workclass fnlwgt    education education.num marital.status
## 1  90         ?  77053      HS-grad             9        Widowed
## 2  82   Private 132870      HS-grad             9        Widowed
## 3  66         ? 186061 Some-college            10        Widowed
## 4  54   Private 140359      7th-8th             4       Divorced
## 5  41   Private 264663 Some-college            10      Separated
## 6  34   Private 216864      HS-grad             9       Divorced
##          occupation  relationship  race    sex capital.gain capital.loss
## 1                 ? Not-in-family White Female            0         4356
## 2   Exec-managerial Not-in-family White Female            0         4356
## 3                 ?     Unmarried Black Female            0         4356
## 4 Machine-op-inspct     Unmarried White Female            0         3900
## 5    Prof-specialty     Own-child White Female            0         3900
## 6     Other-service     Unmarried White Female            0         3770
##   hours.per.week native.country income
## 1             40  United-States  <=50K
## 2             18  United-States  <=50K
## 3             40  United-States  <=50K
## 4             40  United-States  <=50K
## 5             40  United-States  <=50K
## 6             45  United-States  <=50K
```

In the above dataset, few of the columns have missing values populated with '?'. Replacing '?' values with 'NA' will be carried out in three steps:

Step 1: Figure out the columns having '?' values(missing values).

```
Col_name <- vector()
for ( i in 1:ncol(DataCensus))
{
  Missingcount<- 0
  Missingcount <- sum( as.character(DataCensus[ , i]) == "?")

  ##If there are missing values
  if(as.integer(Missingcount) > 0)
  {
    cat("Column " , as.character(colnames(DataCensus)[i] ), " has " , as.charac
  ter(Missingcount) , "missing Values \n" )
    Col_name <- c(Col_name, colnames(DataCensus)[i])
  }
}
```

```
## Column  workclass  has  1836 missing Values
## Column  occupation  has  1843 missing Values
## Column  native.country  has  582 missing Values
```

Step 2: Add 'NA' to the levels of columns containing '?' values.

Step 3: Replace all '?' with 'NA' and remove '?' from the column levels.

```
for ( x in 1:length(Col_name) )
{
  ## Add Values in Level
  levels(DataCensus[, Col_name[x]])[length(levels(DataCensus[, Col_name[x]])) +
  1 ] <- "NA"
  ##replace the Values
  Index<- which(as.character(DataCensus[ , Col_name[x]]) == "?" )
  DataCensus[Index , Col_name[x]] <- "NA"
  #remove the redundant level
  index <- which(levels(DataCensus[, Col_name[x]]) == "?")
  levels(DataCensus[, Col_name[x]])[index] <- "NA"

}
head(DataCensus)
```

```
##   age workclass fnlwgt    education education.num marital.status
## 1  90        NA  77053      HS-grad             9        Widowed
## 2  82   Private 132870      HS-grad             9        Widowed
## 3  66        NA 186061 Some-college            10        Widowed
## 4  54   Private 140359      7th-8th             4       Divorced
## 5  41   Private 264663 Some-college            10      Separated
## 6  34   Private 216864      HS-grad             9       Divorced
##          occupation  relationship  race    sex capital.gain capital.loss
## 1                NA Not-in-family White Female            0         4356
## 2   Exec-managerial Not-in-family White Female            0         4356
## 3                NA     Unmarried Black Female            0         4356
## 4 Machine-op-inspct     Unmarried White Female            0         3900
## 5    Prof-specialty     Own-child White Female            0         3900
## 6     Other-service     Unmarried White Female            0         3770
##   hours.per.week native.country income
## 1             40  United-States  <=50K
## 2             18  United-States  <=50K
## 3             40  United-States  <=50K
## 4             40  United-States  <=50K
## 5             40  United-States  <=50K
## 6             45  United-States  <=50K
```

3. Data Analysis

To understand data distribution, the dataset was explored for some of the columns.

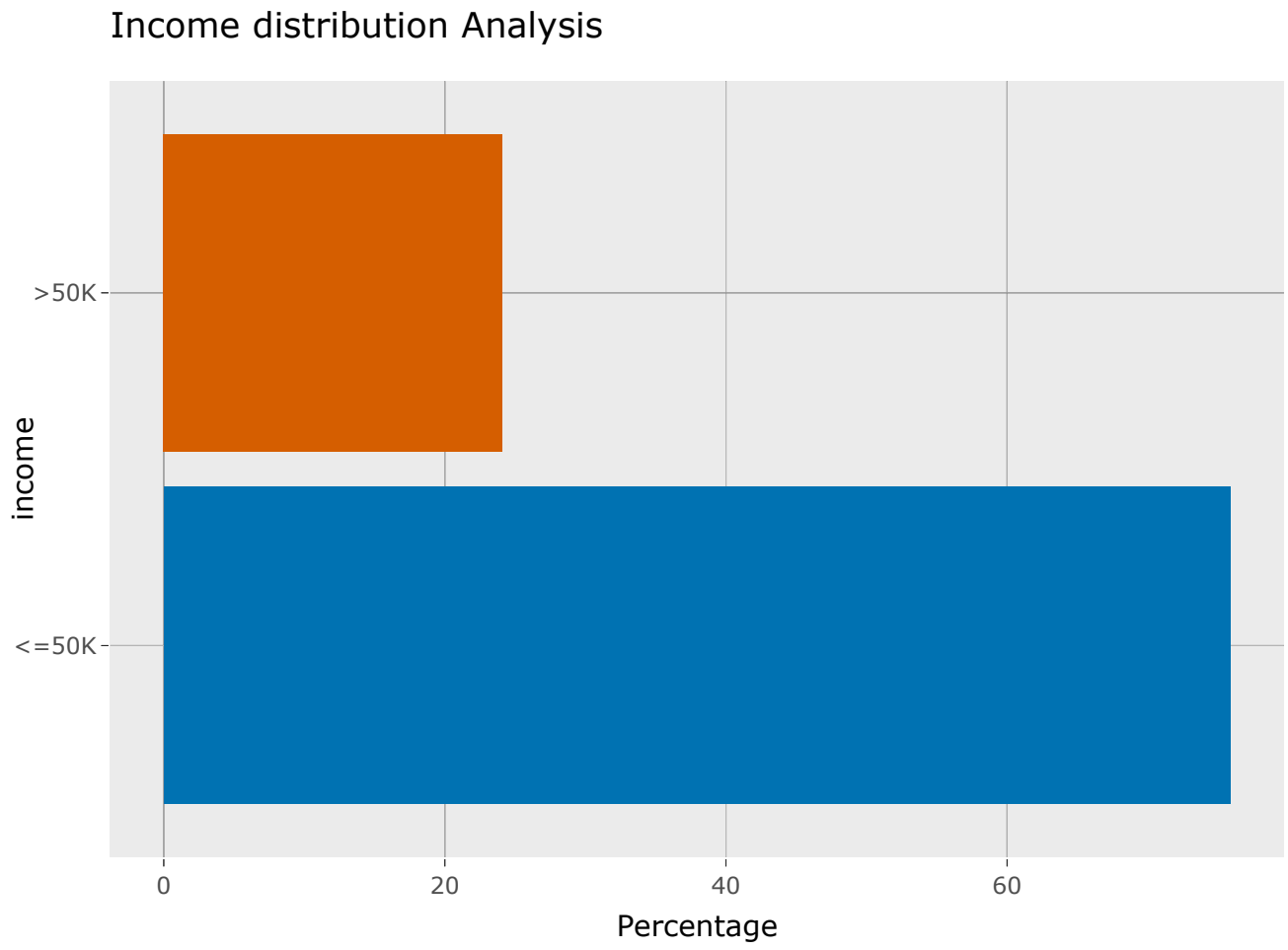
3.1 Categorical Variable

The idea is to analyze the distribution of different categorical variables like income, race, native.country, marital.status, education and sex.

3.1.1 Income Analysis

"**income**" is a categorical column and consists of two values "**<=50K**" and "**>50K**".

```
p <- ggplot(DataCensus, aes(x = income)) +
  geom_bar(aes(y = (..count..)/sum(..count..)*100  ) , fill=c("#0072B2", "#D55E
  00"),stat="count"  )+
  labs(title = "Income distribution Analysis",x = "income",y = "Percentage" )+
  coord_flip()

ggplotly(p)
```



Followings are the deductions from the above graph:

- The data belonging to each category **<=50K** and **>50K** is in approximate ratio of 3:1.
- Maximum population belongs to **<=50K** category.

3.1.2 Workclass Analysis

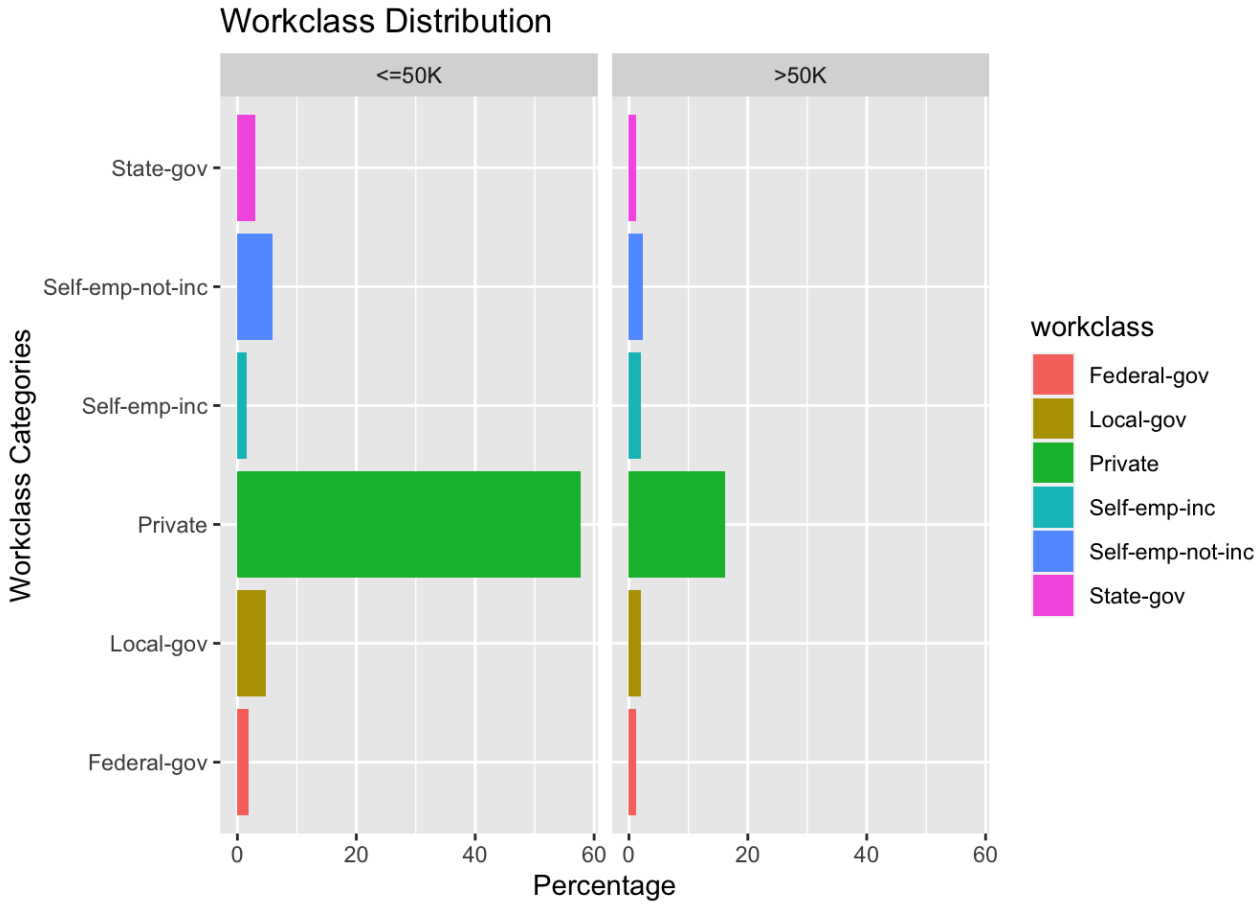
workclass describe the sectors in which each individual is working. workclass column consists of 8 different categories and for some of the records data is missing, so these records belongs to “NA” category.

Category	workClass_Categories
category1	NA
category2	Federal-gov
category3	Local-gov
category4	Never-worked
category5	Private
category6	Self-emp-inc
category7	Self-emp-not-inc
category8	State-gov
category9	Without-pay

Below is the % distribution of different workclass categories belongs to different income categories except for the following categories:

1. **NA**
2. **Without-pay**
3. **Never-worked**

```
p <- ggplot(filter(DataCensus,workclass!= "NA" & workclass!= "Without-pay" & wo
  rkclass!= "Never-worked") , aes(x = workclass)) +
  geom_bar(aes(y = (..count..)/sum(..count..)*100 , fill=workclass ) ,stat="cou
  nt"  )+
  labs(title = "Workclass Distribution ",
       x = "Workclass Categories",
       y = "Percentage" )+
    facet_grid(~income)+

  scale_color_hue()+
  coord_flip()
p
```



Followings are the deductions from the above graph:

- Highest percentage of the population works in the private sector.
- Second-highest percentage of the population works in **self-emp-not-inc** followed by **Local-gov** and **Federal-gov** consecutively.

3.1.3 Sex Distribution Analysis

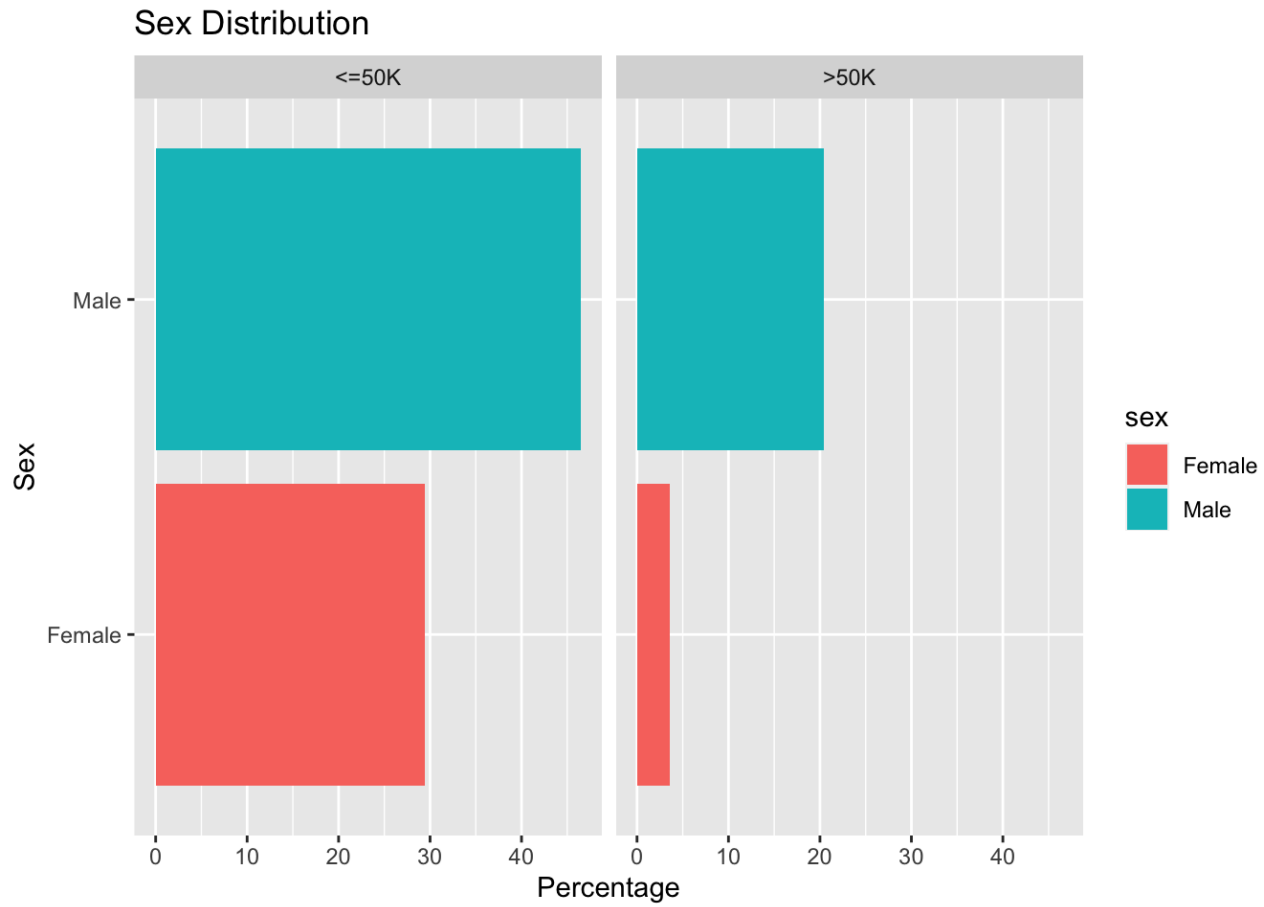
sex column describes male or female distribution in the census dataset.

Summary Table

Category	S_Categories
category1	Female
category2	Male

Male/Female Distribution(%)

```
p <- ggplot(DataCensus, aes(x = sex)) +
  geom_bar(aes(y = (..count..)/sum(..count..)*100 , fill=sex ) ,stat="count"  )
  +
  labs(title = "Sex Distribution ",x = "Sex",y = "Percentage" )+
  facet_grid(~income)+
  scale_color_hue()+
  coord_flip()
p
```



Followings are the deductions from the above graph:

- For both income categories % of male is significantly higher than % of the female category.

3.1.4 Education Distribution Analysis

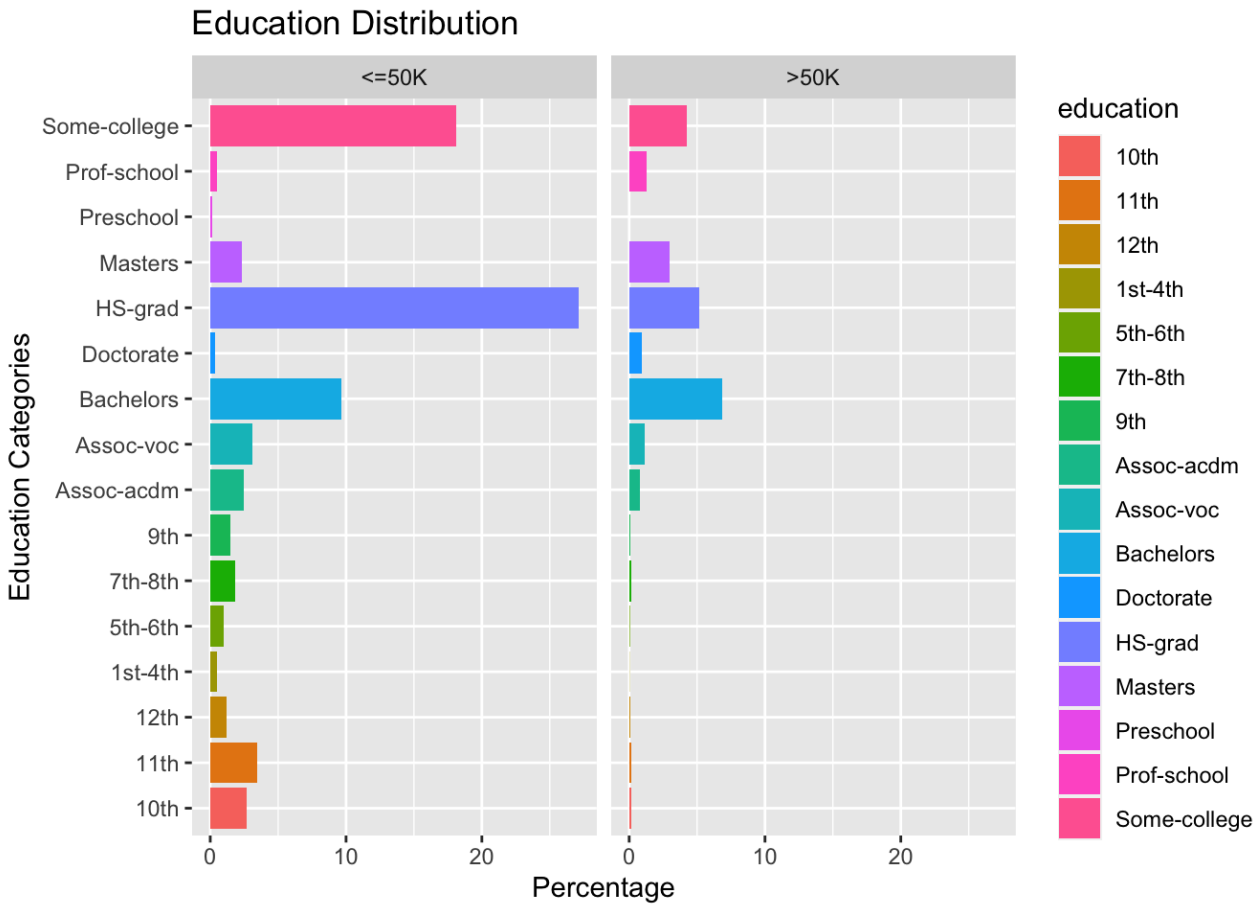
The education column describes the highest level of education for all the individuals in the dataset. Related to education there is another redundant column **education.num**, which contains integer keys corresponding to each education level described. The education column consists of 16 different categories

Category	Education_Categories
category1	10th
category2	11th

Category	Education_Categories
category3	12th
category4	1st-4th
category5	5th-6th
category6	7th-8th
category7	9th
category8	Assoc-acdm
category9	Assoc-voc
category10	Bachelors
category11	Doctorate
category12	HS-grad
category13	Masters
category14	Preschool
category15	Prof-school
category16	Some-college

Percentage of highest education level per income categories

```
p <- ggplot(DataCensus, aes(x = education)) +
  geom_bar(aes(y = (..count..)/sum(..count..)*100 , fill=education ) ,stat="cou
  nt"  )+
  labs(title = "Education Distribution ",x = "Education Categories",y = "Percen
  tage" )+
  facet_grid(~income)+
  scale_color_hue()+
  coord_flip()
p
```



Followings are the deductions from the above graph:

- *High School Graduation* is the education of maximum % of population for income category *<=50K*.
- Second-highest percentage of population for income categories is having *some-college* as highest level of education.

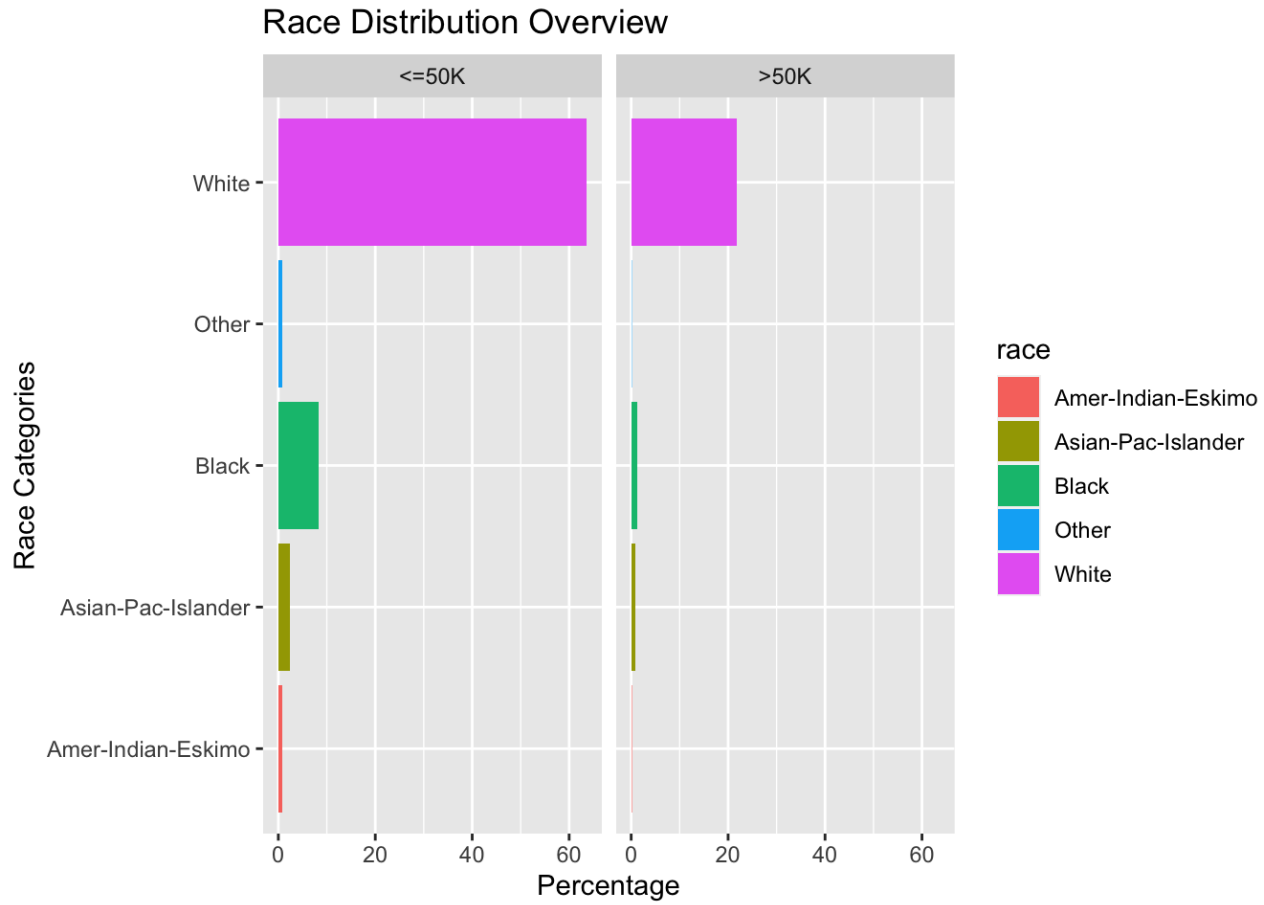
3.1.5 race distribution analysis

The **race** column describes the race related information for all people listed in the census data.

Category	**workClass_Categories**
category1	Amer-Indian-Eskimo
category2	Asian-Pac-Islander
category3	Black
category4	Other
category5	White

Race Distribution in entire Dataset

```
p <- ggplot(DataCensus, aes(x = race)) +
  geom_bar(aes(y = (..count..)/sum(..count..)*100 , fill=race ) ,stat="count"
   )+
  labs(title = "Race Distribution Overview",x = "Race Categories",y = "Percenta
  ge" )+
  facet_grid(~income)+
  scale_color_hue()+
  coord_flip()
p
```



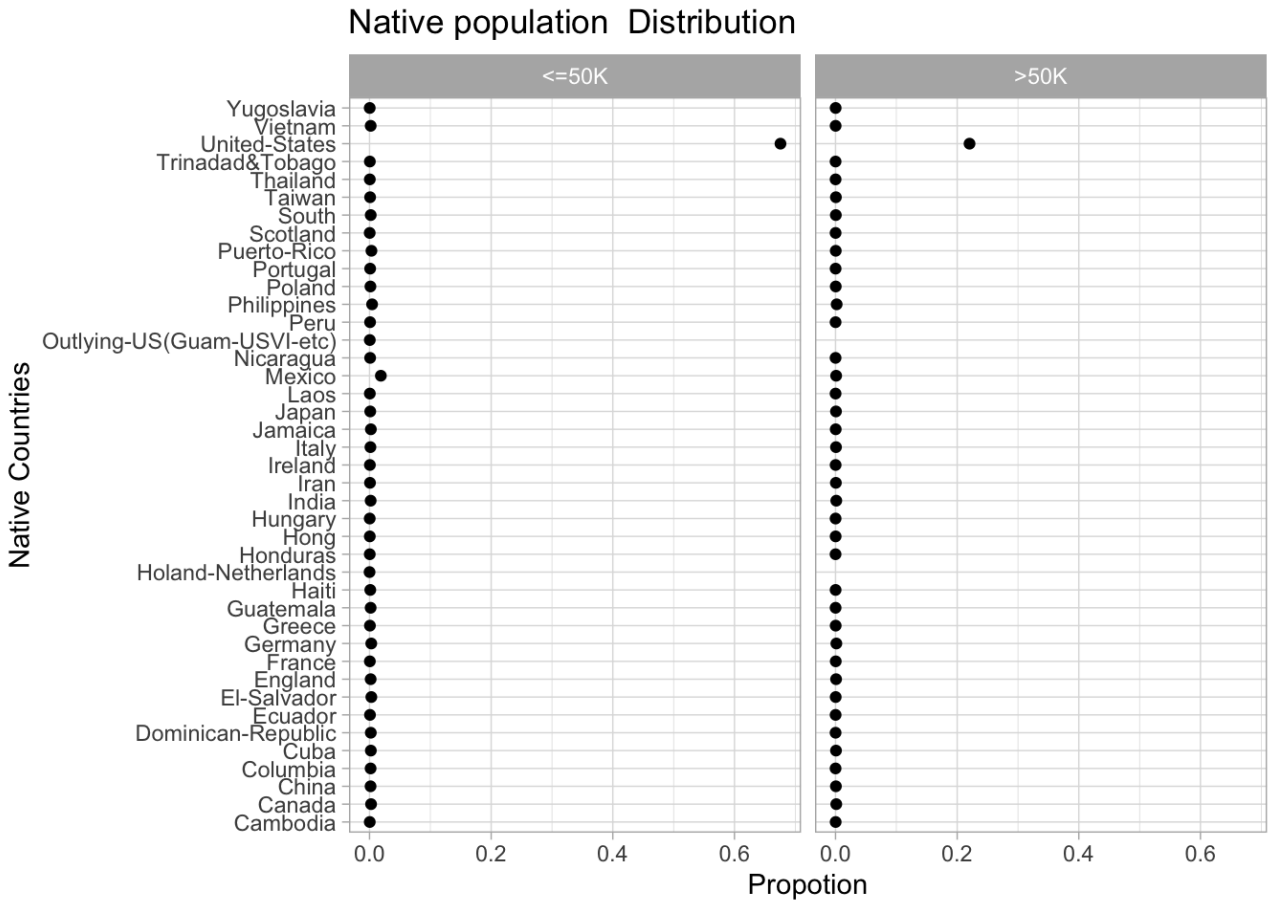
Followings are the deductions from the above graph:

- Majority of population belonging to the White race.
- Black race attributes to second-highest % of entire population.

3.1.6 native.country analysis

The **native.country** column describes details of native countries of the population.

```
p <- DataCensus %>%
  filter(native.country != "NA") %>%
  group_by(income,native.country) %>%
  summarize(Proportion = n()/ nrow(DataCensus)) %>%
ggplot(aes(x =  native.country , y = Proportion))+
 geom_point()+
  theme(axis.text.x = element_text(angle = 90, hjust = 1))+
  coord_flip()+
   labs(title = "Native population  Distribution ",x = "Native Countries",y =
  "Propotion" )+
  facet_grid(~income)+
  theme_light()

p
```



Followings are the deductions from the above graph:

- Majority of the population of the dataset are indigenous.
- Natives from Mexico are the second-highest in the entire population (~ <5%).

3.1.7 occupation Analysis

The **occupation** column describes occupation related information for entire population.

```
p <- ggplot(DataCensus, aes(x = occupation)) +
  geom_bar(aes(y = (..count..)/sum(..count..)*100 , fill=occupation ) ,stat="c
ount"  )+
  labs(title = "Occupation Distribution ",x = "occupation Categories",y = "Per
centage" )+
  facet_grid(~income)+
  scale_color_hue()+
  coord_flip()
p
```

Followings are the deductions from the above graph:

- Majority % of population in income category **<=50K** are working as **Adm-clerical**.
- Majority % of population in income category **>50K** are working as **Exec-managerial**.

3.2 Continuous Variable

The idea is to analyze the distribution of different continuous variables like *age*, *work per hour*, *capital gain* and *capital loss*.

3.2.1 age Analysis

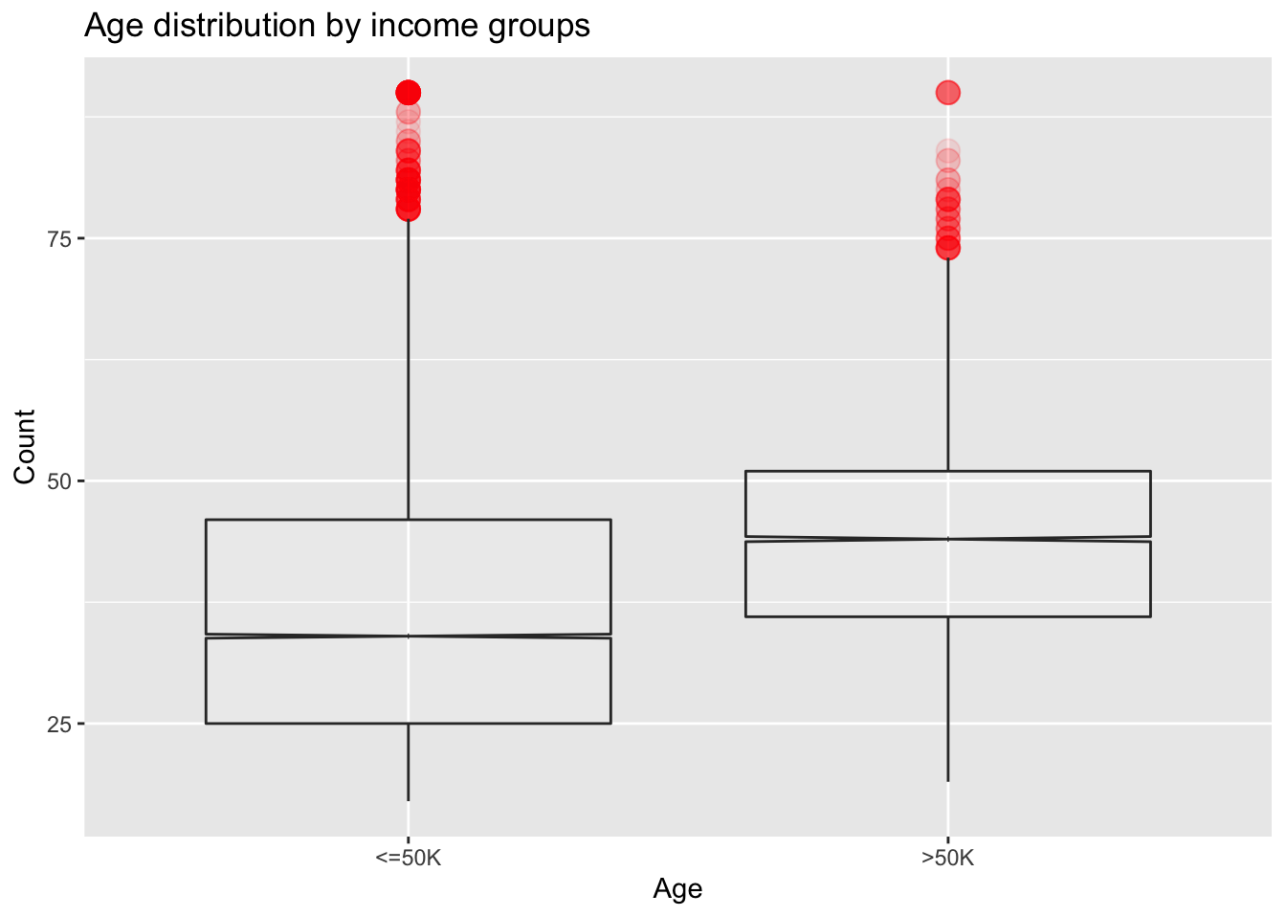
The age is in the range of 17 to 90 years and the age distribution looks like:

```
summary(DataCensus$age)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   17.00   28.00   37.00   38.59   48.00   90.00
```

Age distribution for income categories

```
g <- ggplot(data =DataCensus )+
  geom_boxplot(aes(x=income, y= age),outlier.size=4, outlier.colour='red', alph
  a=0.1,notch = TRUE,notchwidth = 0.003 )+
  labs(title = "Age distribution by income groups",x="Age",y = "Count")+
  scale_color_hue()
g
```



Followings are the deductions from the above graph:

- Majority of outliers lie in upper age range.
- For **<=50K** *income* category 25 to 75 percentile of the population lies in age range of 25-46 years.
- For *income* category **>50K** 25 to 75 percentile of the population lies in age range of 36-51 years.

3.2.2 hours.per.week Analysis

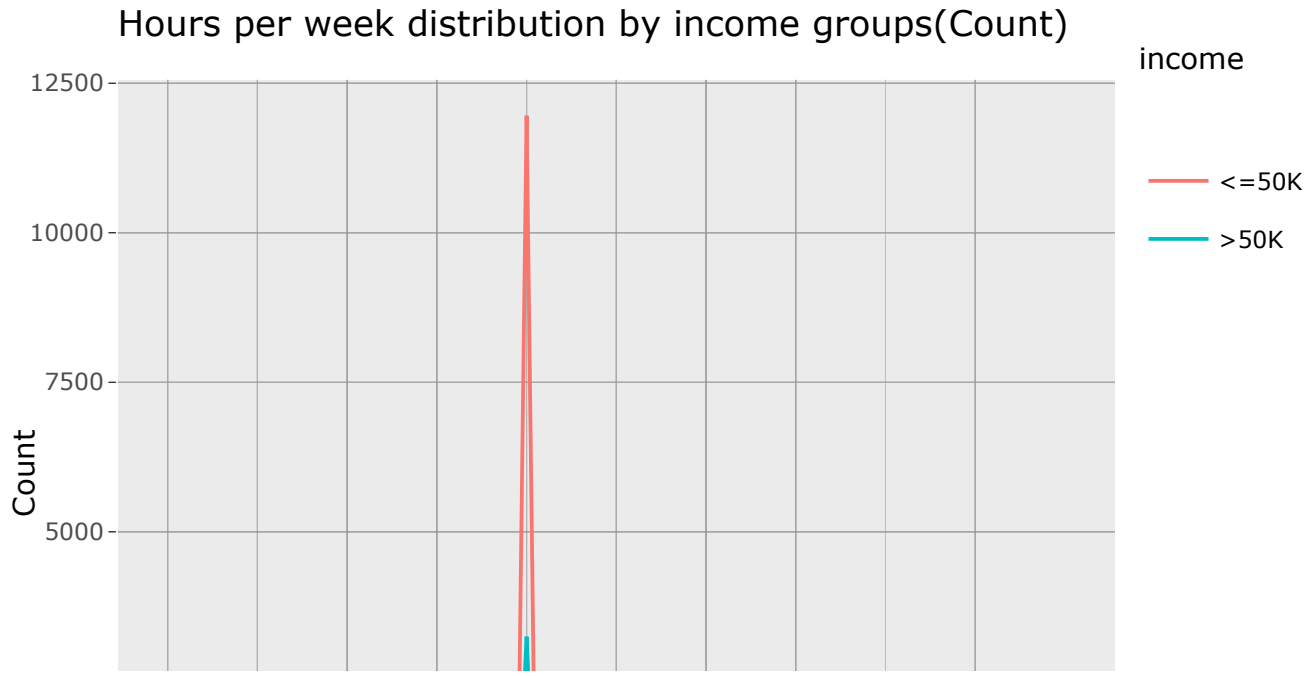
hours.per.week column distribution indicates the number of hours per week worked by individuals.

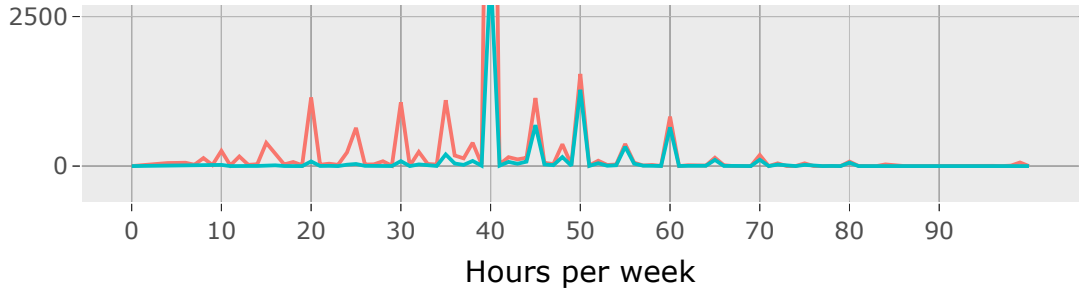
Summary of hours per week for entire data:

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1.00   40.00   40.00   40.44   45.00   99.00
```

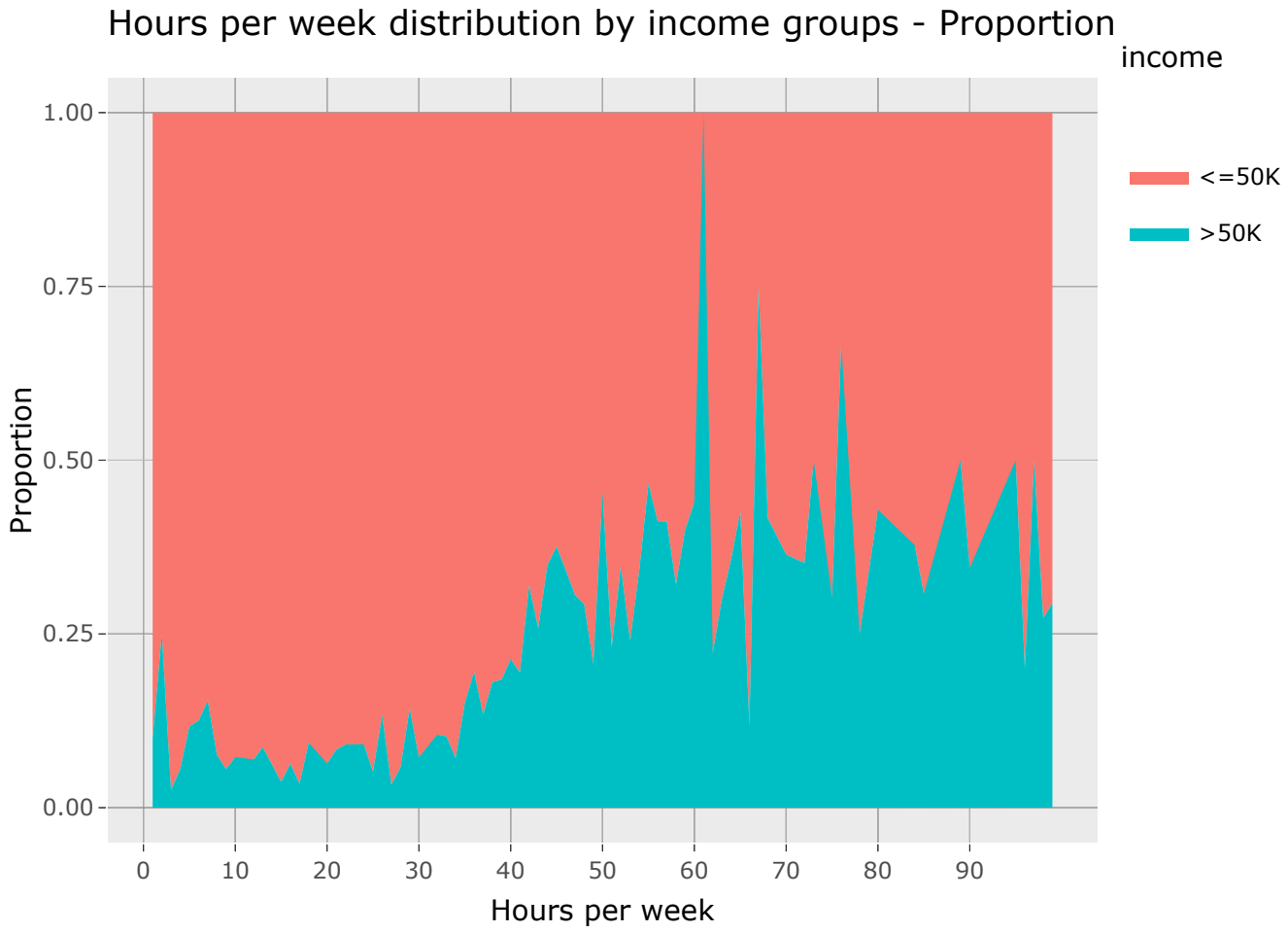
The hours per week distribution for both income categories

```
par(mfrow=c(2,1))
 p<- ggplot(DataCensus, aes(hours.per.week))+
  geom_freqpoly(aes(col= income), binwidth = 1)+
  scale_x_continuous(breaks = round(seq(min(DataCensus$hours.per.week), max(Dat
  aCensus$hours.per.week), by = 10),-1))+
    scale_colour_hue()+
  theme(legend.position = "right")+
  labs(title = "Hours per week distribution by income groups(Count)",
       x ="Hours per week",
       y = "Count")

g <- DataCensus %>%
  group_by(income, hours.per.week) %>%
  summarize(count = n()) %>%
  ungroup() %>%
  group_by(hours.per.week) %>%
  mutate(prop = count / sum(count)) %>%
  ggplot(., aes(hours.per.week, prop, fill = income))+
  geom_area()+
  scale_x_continuous(breaks = round(seq(min(DataCensus$hours.per.week), max(Dat
  aCensus$hours.per.week), by = 10),-1))+
  theme(legend.position = "right") +
  labs(title = "Hours per week distribution by income groups - Proportion",
       x ="Hours per week",
       y = "Proportion")


ggplotly(p)
```






```
ggplotly(g)
```




```
par(mfrow=c(1,1))
```

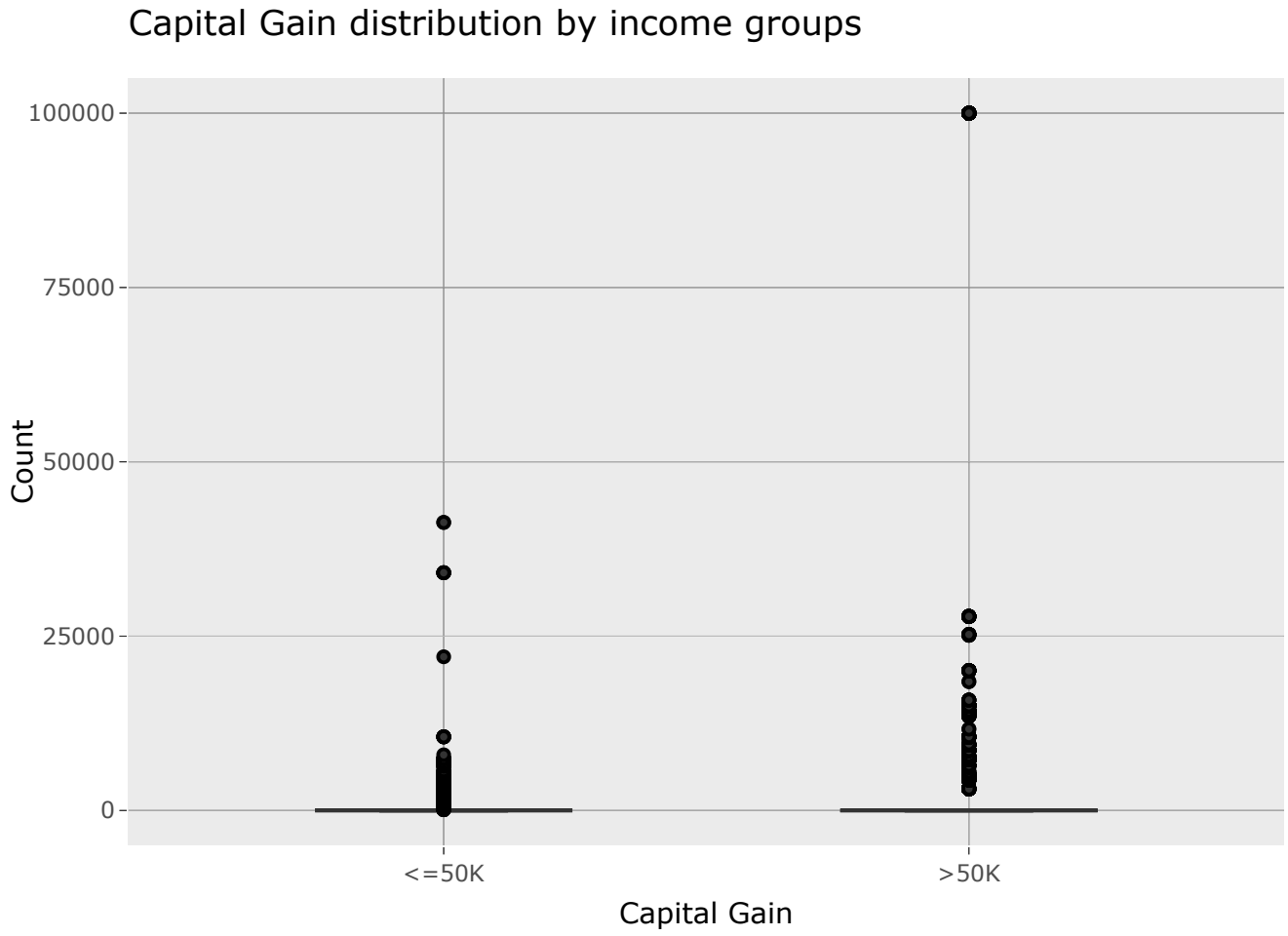
Followings are the deductions from the above graph:

- Highest number of individuals in both income categories are working 40 hours per week.
- Some extreme cases where people are working as high as 99 hours per week.

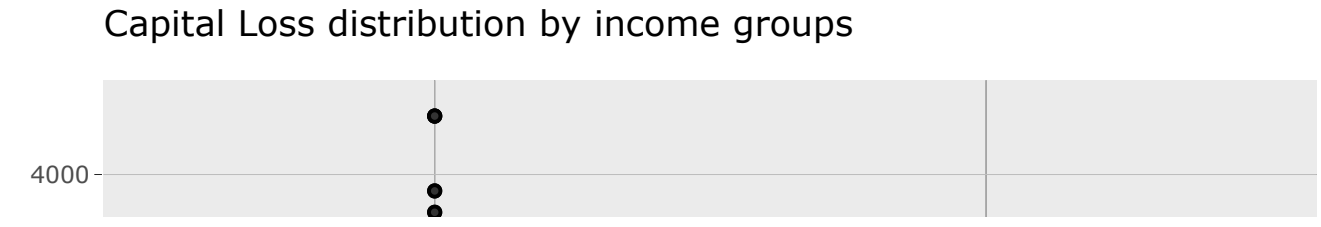
3.2.3 capital.gain and capital.loss Analysis

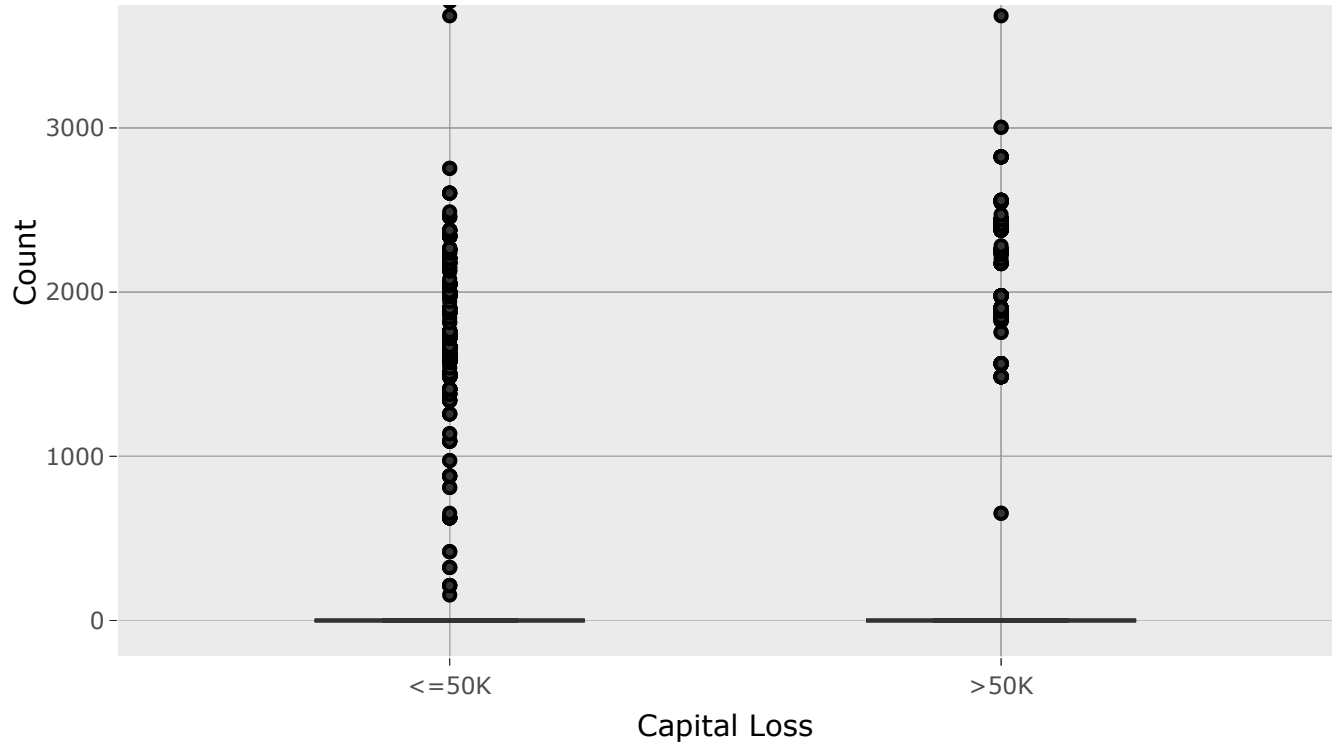
capital.gain and **capital.loss** columns depicts the loss and gain for entire population listed in this census dataset.

```
par(mfrow=c(2,1))
g <- ggplot(data =DataCensus )+
  geom_boxplot(aes(x=income, y= capital.gain),outlier.size=4, outlier.colour='r
  ed', alpha=0.1,notch = TRUE,notchwidth = 0.003 )+
  labs(title = "Capital Gain distribution by income groups",x="Capital Gain",y
   = "Count")+
  scale_color_hue()

h <- ggplot(data =DataCensus )+
  geom_boxplot(aes(x=income, y= capital.loss),outlier.size=4, outlier.colour='r
  ed', alpha=0.1,notch = TRUE,notchwidth = 0.003 )+
  labs(title = "Capital Loss distribution by income groups",x="Capital Loss",y
   = "Count")+
  scale_color_hue()

ggplotly(g)
```




```
ggplotly(h)
```





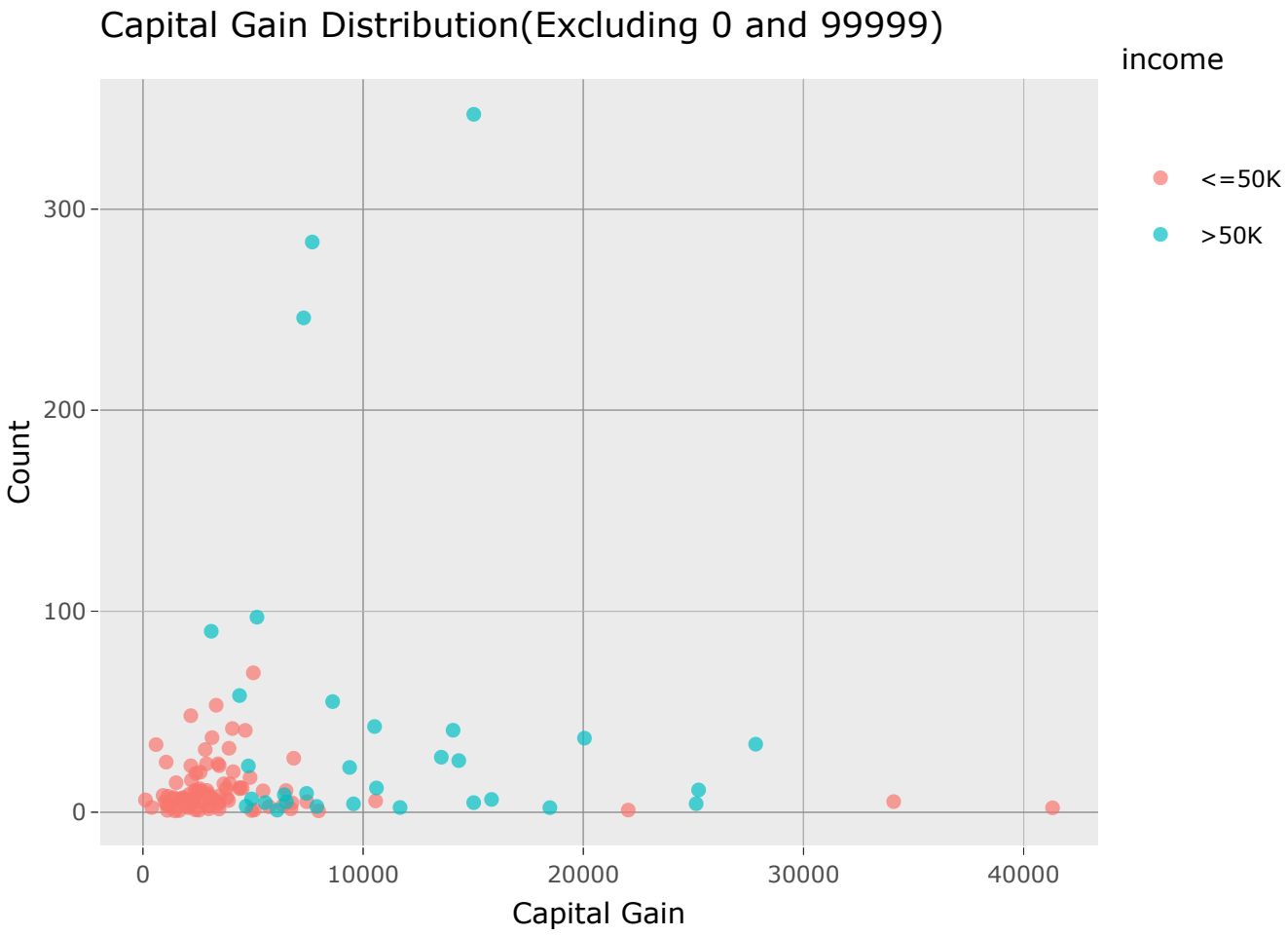

```
par(mfrow=c(1,1))
```

Followings are the deductions from the above graph:

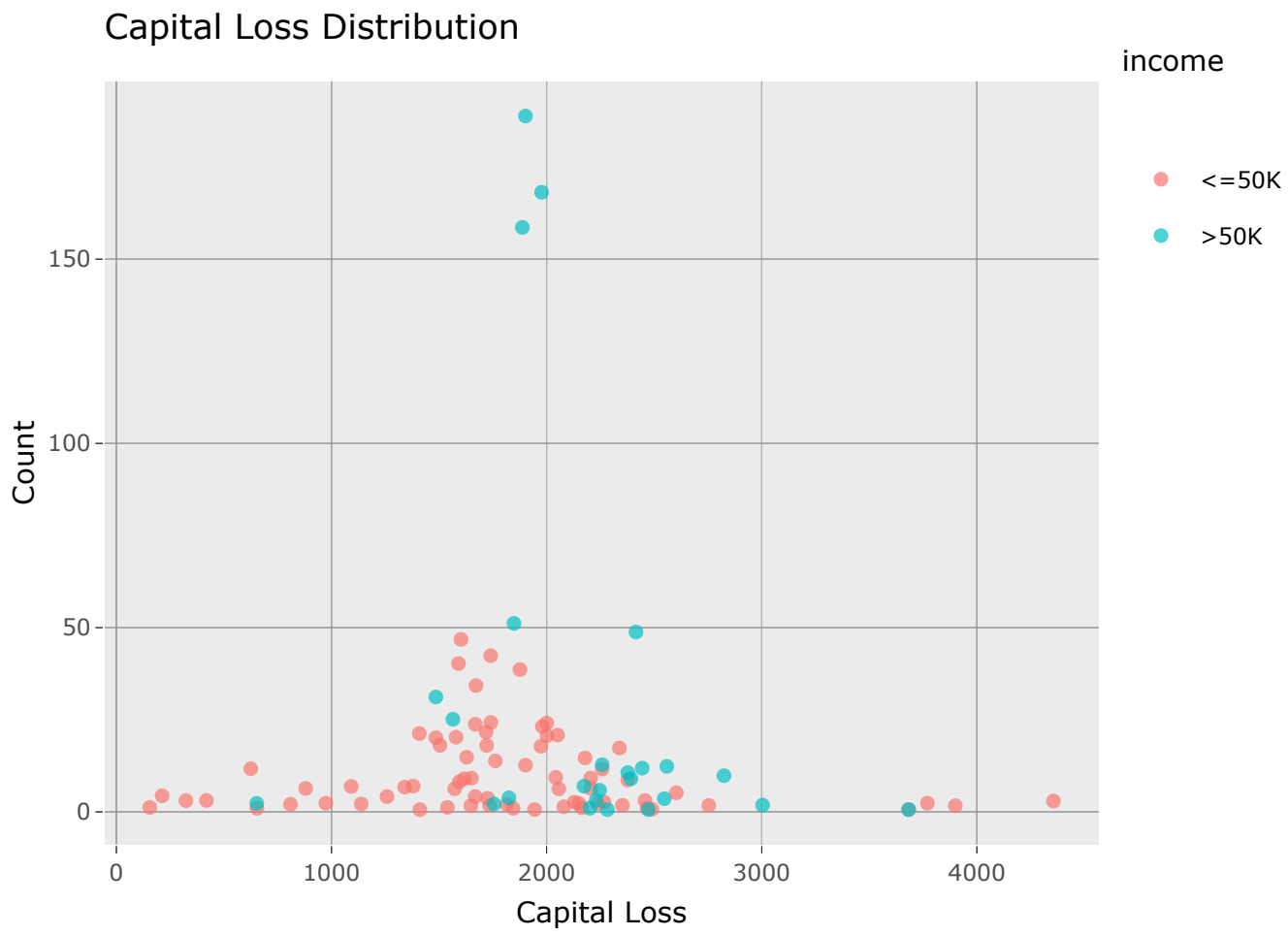
- For **capital gain** and **capital loss** minimum, 1st, 2nd and 3rd quantile values are equal to “0”.
- At least **75%** of population has no **capital gain** or **loss**.
- The distributions of these variables is extremely skewed.
- Capital gain values have got multiple outliers ranging from **3103** to **100000** for people earning more than **>50K**.

To understand the rest of distribution we removed the values ‘0’ and ‘99999’ of capital gain .

```
par(mfrow=c(2,1))
 p<- DataCensus %>%
   filter(capital.gain!= 0 &  capital.gain!= 99999) %>%
   group_by(income, capital.gain) %>%
   summarize(count = n()) %>%
   ungroup() %>%
   ggplot(aes(x = capital.gain, y = count, colour = income)) +
   geom_point(alpha = 0.7,  position = position_jitter())+
  theme(legend.position = "right") +
  labs(title = "Capital Gain Distribution(Excluding 0 and 99999)",
       x ="Capital Gain",
       y = "Count")


q<-  DataCensus %>%
   filter(capital.loss!= 0 ) %>%
   group_by(income, capital.loss) %>%
   summarize(count = n()) %>%
   ungroup() %>%
   ggplot(aes(x = capital.loss, y = count, colour = income)) +
   geom_point(alpha = 0.7,  position = position_jitter())+
  labs(title = "Capital Loss Distribution",
       x ="Capital Loss",
       y = "Count")
ggplotly(p)
```



```
ggplotly(q)
```




```
par(mfrow=c(1,1))
```

Followings are the deductions from the above graph(after removing extreme outliers):

capital.gain Distribution

- Population earning **>50K** maximum **capital gain** is **15024** and maximum number of people are belonging to this category.
- Population earning **<=50K** maximum **capital gain** is **41310** but very few people belong to this category.
- **Capital Gain** for people earning **>50K** population is higher as compared to **<=50K**.

capital.loss Distribution

- People earning **>50K** have incurred the **highest Capital Loss**.
- For all people earning **<=50K** observation counts for each individual is less than **50**.

4. Data Exploration

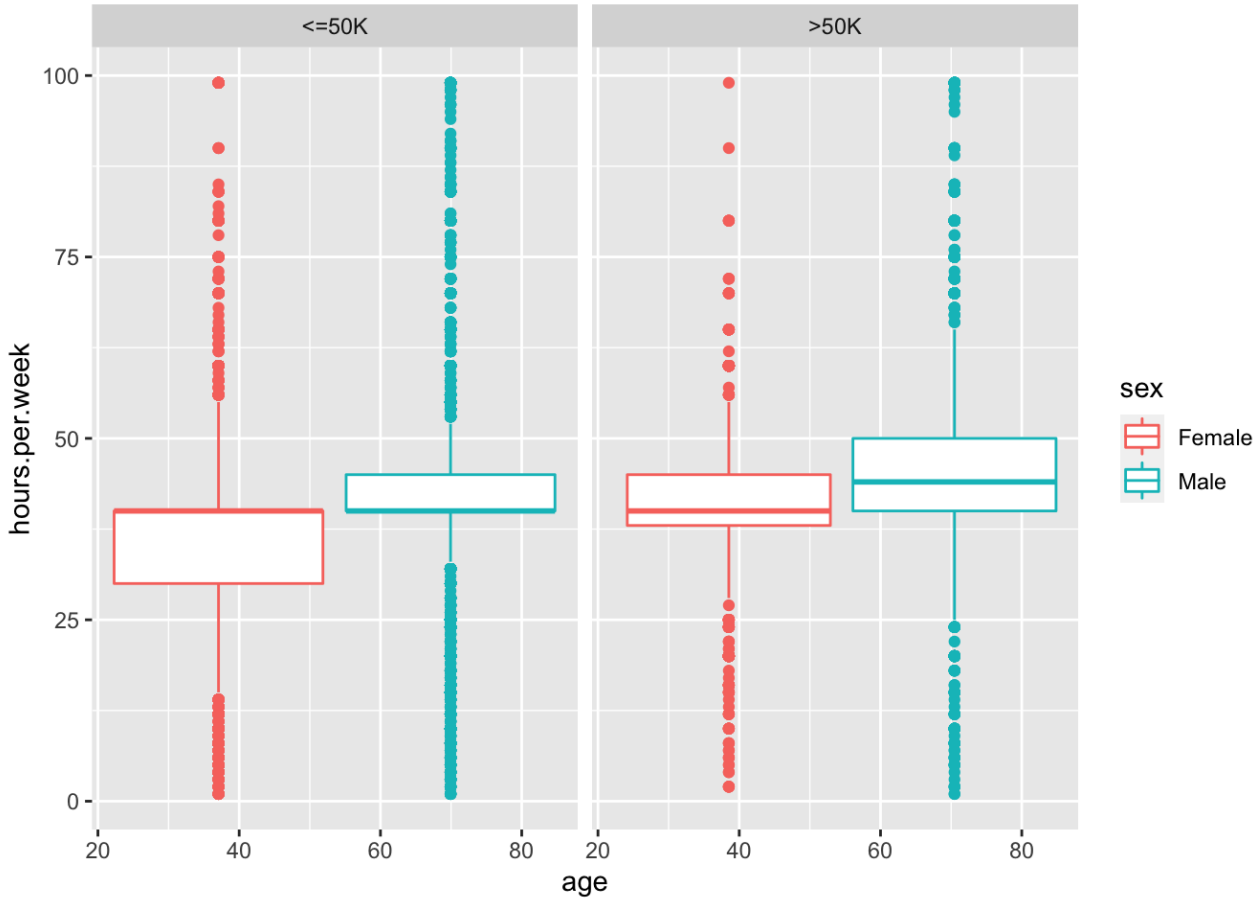
To understand the data in more depth, i.e how these variables are impacting one another, this dataset is explored in depth.

4.1 hours.per.week and age

To understand the distribution of population based on *age* and *hours.per.week* , we have plotted below point plot

```
par(mfrow=c(2,1))
P <- ggplot(DataCensus , aes(x = age , y = hours.per.week) )+
  geom_point(aes(color = sex) , alpha = 0.3,position = position_jitter())+
   labs(title = "Hours work per week distribution over Age",x = "age",y = "Hour
  s Per Week" )+
  facet_grid(~income)

P
```


```
qplot(x= age , y = hours.per.week ,data =DataCensus ,geom = "boxplot", colour =
  sex , facets =~income)
```




```
par(mfrow=c(1,1))
```

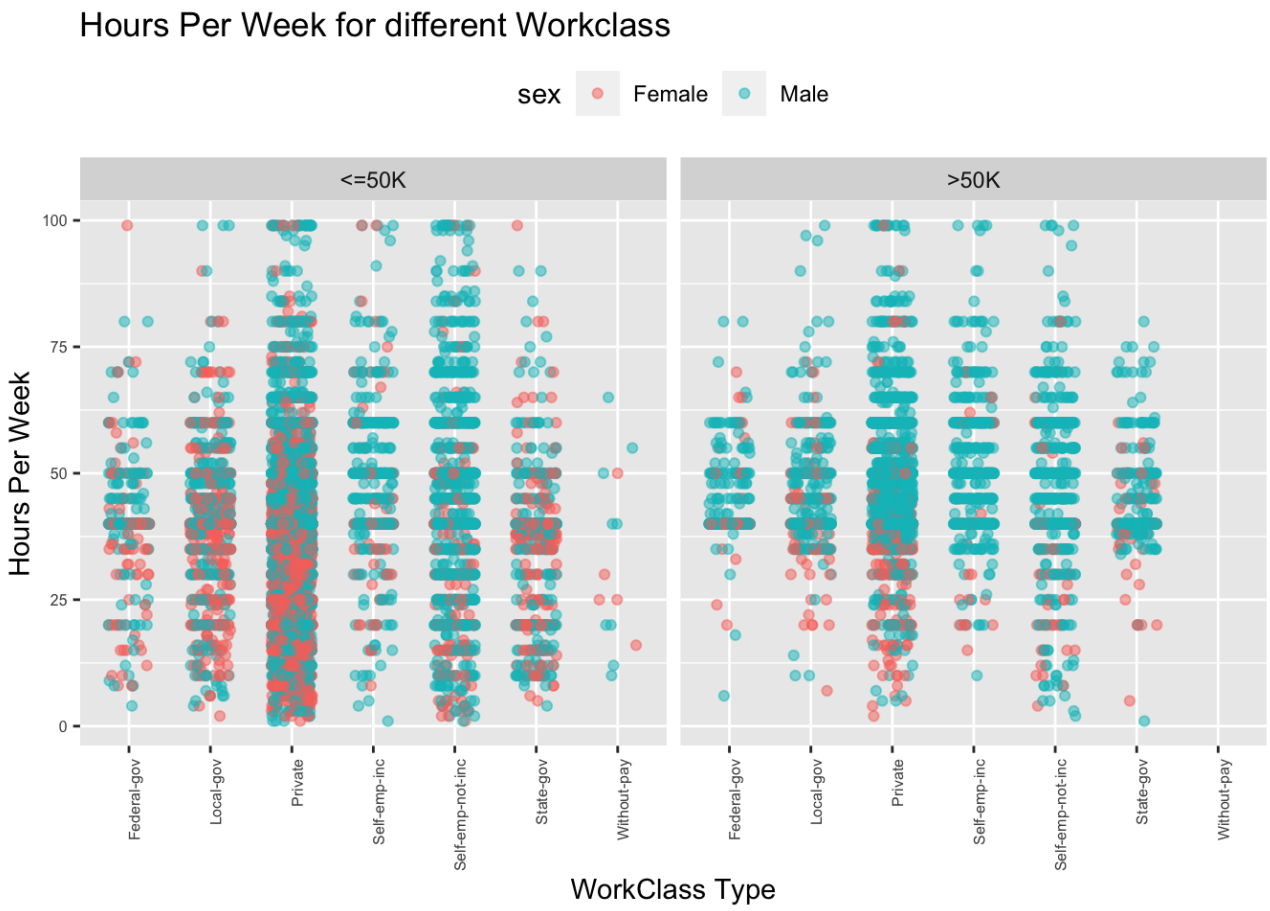
Followings are the deductions from the above graph:

- Population earning **<=50K** only one male aged 90 years is working for 99 hours per week, which is an anomaly.
- Population in income category "**<=50K**" are working for longer hours as compared to population in income category "**>50K**".
- For both the income categories maximim population is working 40 hours per week.

4.2 workclass and hours.per.week

For understanding the workload in various workclass category we plotted a dot plot. Since Never-worked and NA(missing values) categories doesn't bring value in analysis so the corresponding data is filtered out.

```
Data_sum <- data.frame(table(DataCensus$workclass, DataCensus$education, DataCe
  nsus$income))

colnames(Data_sum) <- c("workclass" ,"education" , "income" , "count" )

P <- ggplot(subset(DataCensus , workclass != "NA" & workclass != "Never-worked"
  ),aes(x = workclass,y = hours.per.week)) +
   theme(legend.position="top",axis.text=element_text(size = 6))+
   geom_point(aes(color = sex),alpha = 0.5,size = 1.5,
              position = position_jitter(width = 0.25, height = 0))+
   theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
   labs(title = "Hours Per Week for different Workclass",x = "Workclass Categor
  ies",y = "Hours Per Week" )+
   scale_x_discrete(name="WorkClass Type") +
   facet_grid(~income)
P
```



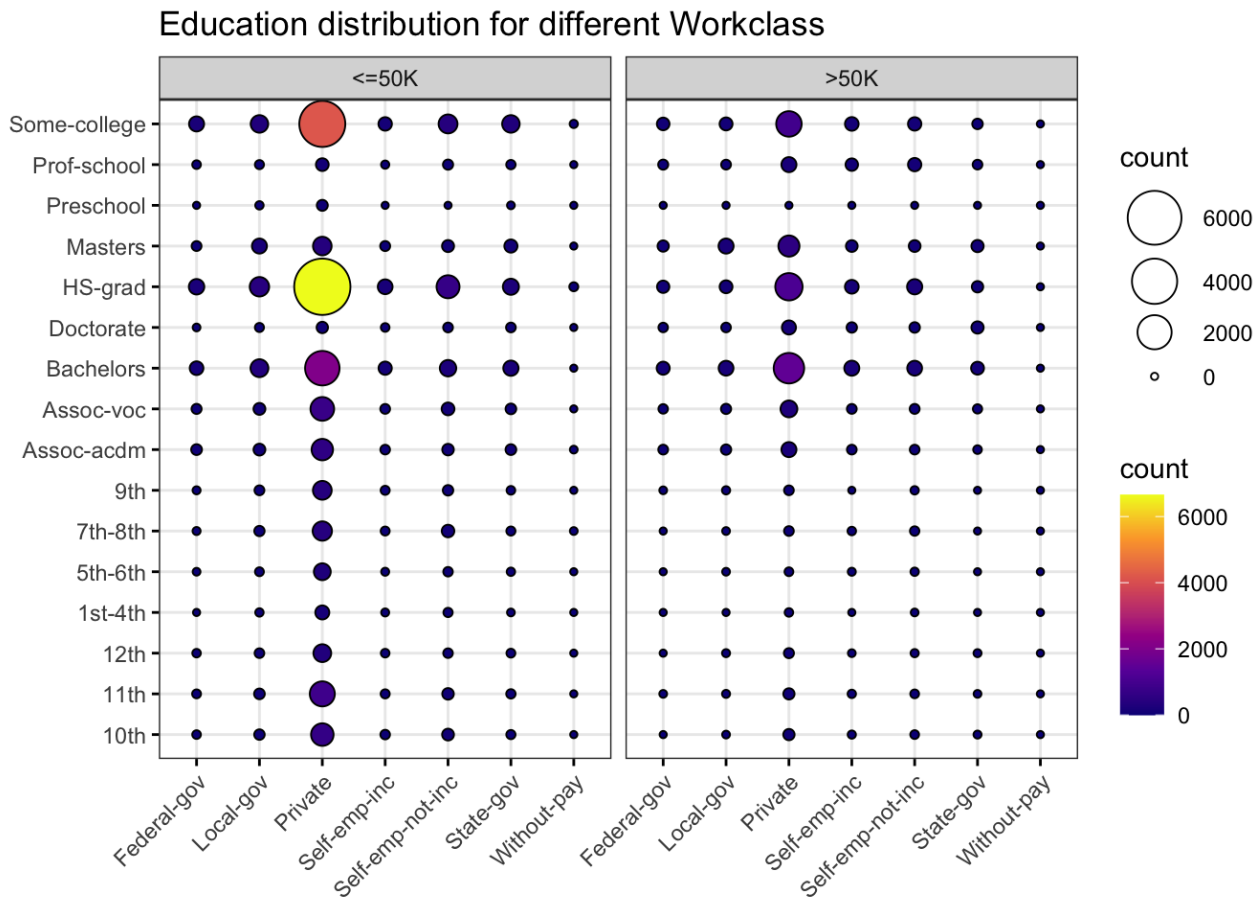
Followings are the deductions from the above graph:

- Only one **Female** employee working for **private** in income range **>50K** is working for 99 hours per week.
- The highest number of hours per week is 99 for females earning **<=50** in four different workclass categories
- Irrespective of their earning categories the males are working 99 hours per week in four different workclass categories.

4.3 workclass and education

For understanding the highest education level of people belonging to different Workclass. Since Never-worked and NA(missing values) categories don't make sense in the plot we filtered out that data.

```
Data_sum <- data.frame(table(DataCensus$workclass,DataCensus$education, DataCen
  sus$income))

colnames(Data_sum) <- c("workclass" ,"education" , "income" , "count" )

library(ggpubr)

P <-ggballoonplot(subset(Data_sum , workclass != "NA" & workclass != "Never-wor
  ked" ), x = "workclass", y = "education", size = "count",
              fill = "count", facet.by = "income",
              ggtheme = theme_bw()) +
  labs(title = "Education distribution for different Workclass",x = "workclass"
  ,y = "Education" )+
  scale_fill_viridis_c(option = "C")

P
```



Followings are the deductions from the above graph:

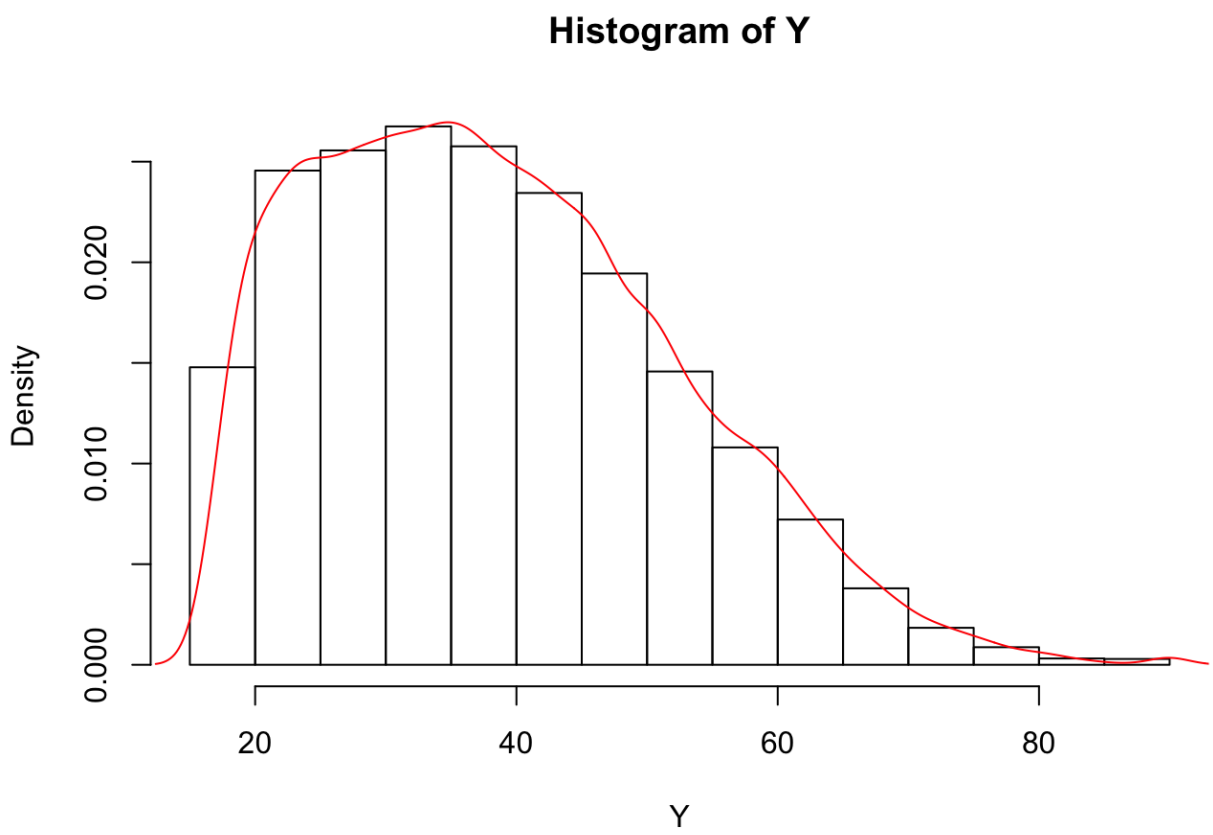
- For income category **<=50K** majority of the population the highest level of education is **HS-Grad** and are working in **private** sector.

- For income category **<=50K** second-highest level of education for the majority of the population is **Some-College** and are working in **private** sector.
- For income category **>50K** highest level of education for the majority of the population is **Bachelors** and are employed in **private** sector.

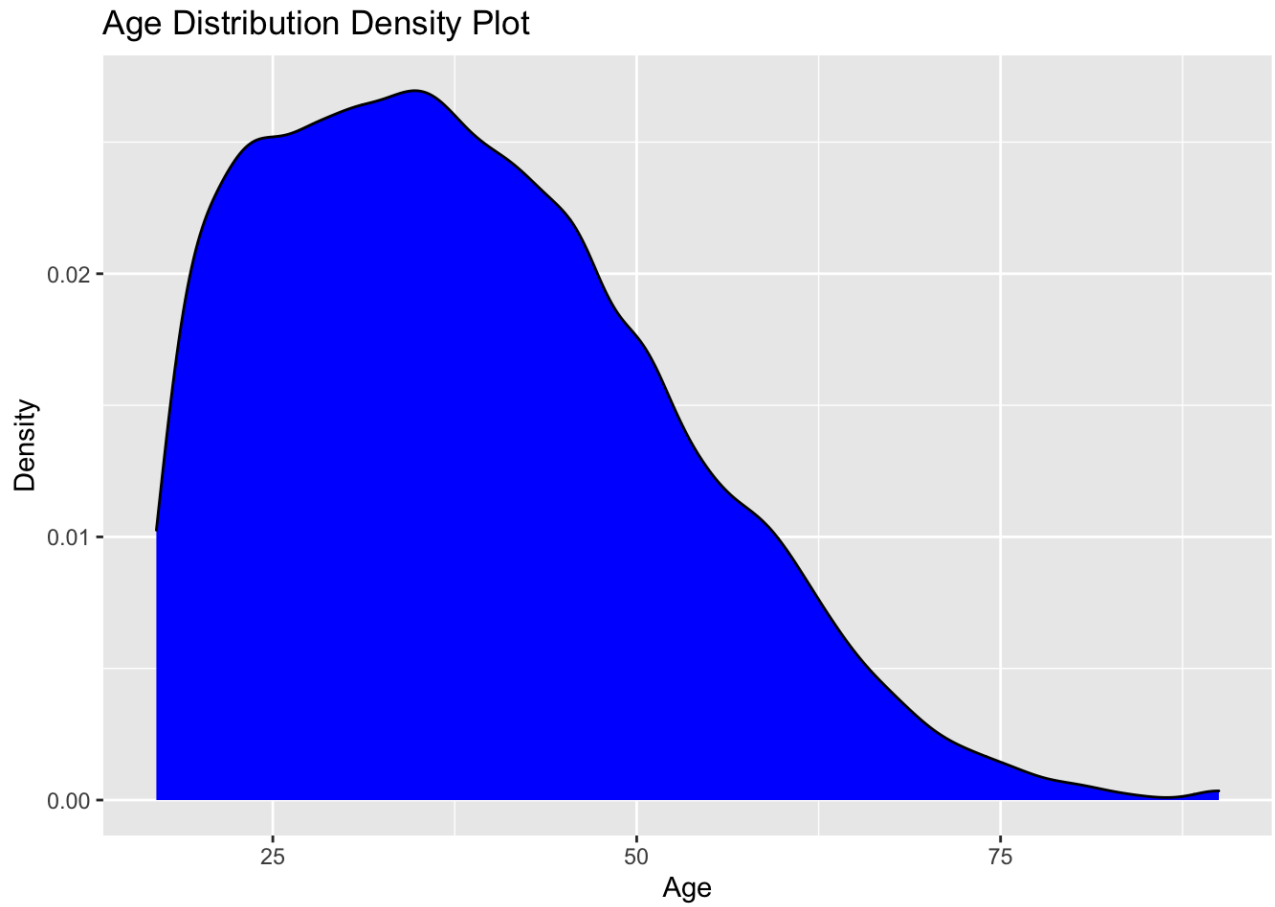
5. Data Distribution Analysis For age

Age is the numerical attribute for which distribution patterns will be analyzed. To understand the distribution pattern category for Age attribute,density plot and histograms are used.

```
Y <- as.numeric(as.character(DataCensus$age))
g <- DataCensus %>%
  ggplot(aes(age)) +
  geom_density(fill="blue")+
  labs(title = "Age Distribution Density Plot",x = "Age",y = "Density" )
d <- density(Y)
hist(Y , prob = TRUE)
lines(d, col="red")
```




```
g
```



The histogram and density plots indicate the Age distribution are **right skewed**. This probably can be a result of either the lower age limit or the outliers of upper values.

```
rm(d)
```

6. Central Limit Theorem application on Age

The **Central Limit Theorem** states that the distribution of the sample means for a given sample size of the population has the shape of the normal distribution.

For testing the applicability of **Central Limit Theorem** on Age column following steps are performed

- Fetch the summary of Age variable

```
summary(DataCensus$age)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   17.00   28.00   37.00   38.59   48.00   90.00
```

- Decide random samples and sample sizes. To test the applicability of **Central Limit Theorem** we will be selecting **10000** sample of sizes <**5000, 6000, 7000, 8000**> respectively.
- Calculate the mean of all the sample sizes.

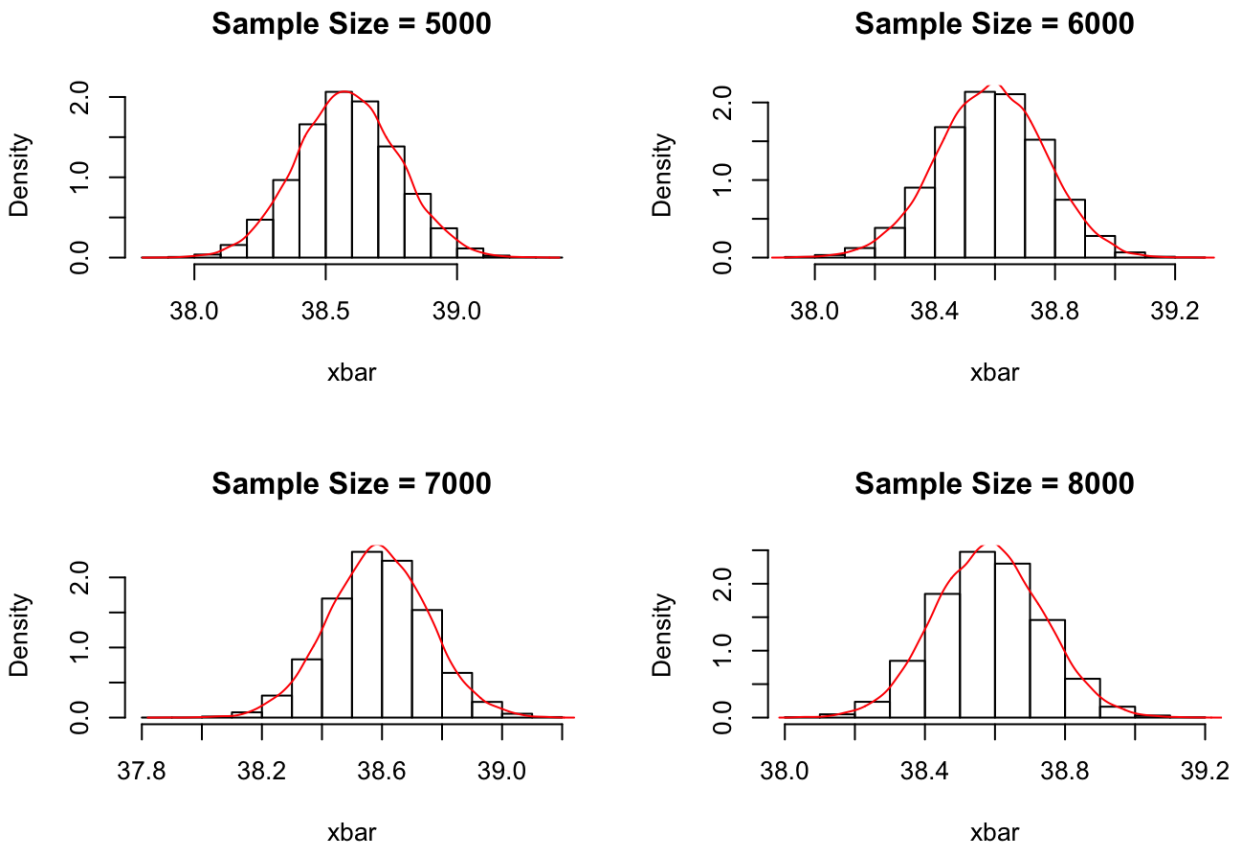
- Plot a histogram of Mean values.
- Compare the standard deviation for each sample size.

```
#initialize Sample variables
samples <- 10000
xbar <- numeric(samples)
set.seed(11)
#Vector For Storing The SD
Actual_sd <- vector()
par(mfrow = c(2,2))

for (size in c(5000, 6000, 7000, 8000)) {

  for (i in 1:samples) {
    xbar[i] <- mean(sample(Y, size = size,
                           replace = TRUE))

  }
  #Histogram for All sample size
  hist(xbar, prob = TRUE,main = paste("Sample Size =", size))
  d <- density(xbar)
  lines(d, col="red")
  #display the Mean and standard deviation for each sample size
  cat("Sample Size = ", size, " Mean = ", mean(xbar),
      " SD = ", sd(xbar), "\n")
  #store the  SD for each sample size
  Actual_sd <- c( Actual_sd , sd(xbar))
}
```

```
## Sample Size =  5000  Mean =  38.58441  SD =  0.1906765
```

```
## Sample Size =  6000  Mean =  38.58618  SD =  0.1764915
```

```
## Sample Size =  7000  Mean =  38.58771  SD =  0.163693
```




```
## Sample Size =  8000  Mean =  38.58349  SD =  0.1523
```

```
par(mfrow = c(1,1))
```

- Calculation of expected SD for each sample size.

```
options(digits = "3")
Sample_size <- c(5000, 6000, 7000, 8000)
Expected_sd <- sd(DataCensus$age)/sqrt(Sample_size)
options(digits = "7")

df <- cbind(Sample_size , Actual_sd , Expected_sd)

tableFormat(df)
```

Sample_size	Actual_sd	Expected_sd
5,000	0.19	0.19
6,000	0.18	0.18
7,000	0.16	0.16
8,000	0.15	0.15

Below are the findings from the histograms and standard deviation table

- Histograms of mean indicate that the mean values follow a normal distribution.
- The actual and expected standard deviation is the same for the randomly picked sample sizes.

The above fact indicates that **Central Limit Theorem** holds for Age distribution.

7. Sampling

To understand the impact of various sampling techniques on the dataset, below listed sampling techniques are applied on the dataset

- **Simple random sampling with Replacement(SRSWR)**
- **Simple random sampling without Replacement(SRSWOR)**
- **Systematic Sampling**

7.1 Simple random sampling with Replacement(SRSWR)

Using *simple random sampling with replacement* technique we will be selecting **12000** samples out of **32537** records.

```
#size Of entire Dataset
N <- nrow(DataCensus)
# Sample size
n <- 12000

#fetch the sample
s <-  srswr(n, N)
##Sample
rows <- (1:N)[s!=0]

#generate the row sequence
rows <- rep(rows, s[s != 0])

#sample Data
Data_srswr <- na.omit(DataCensus[rows , ])
#store the Mean value of Age column in Original D.s and sample
Sampling_mean <- c ( Sampling_mean , mean(Data_srswr$age) )
Sampling_SD <- c(Sampling_SD , sd(Data_srswr$age))
cat("Mean and standard deviation of Sample generated using SRSWR : <" ,  Sampli
  ng_mean[2] , ",",Sampling_SD[2],  ">\n")
```

```
## Mean and standard deviation of Sample generated using SRSWR : < 38.67267 , 1
3.58321 >
```

7.2 Simple random sampling without Replacement(SRSWOR)

Using *simple random sampling without replacement* technique we will be selecting **9000** samples out of **32537** records.

```
n <- 9000

s <- srswor(n, N)

rows <- (1:N)[s!=0]

rows <- rep(rows, s[s != 0])

#Store the samples
Data_srswor <- na.omit(DataCensus[rows, ])
#store the Mean value of Age column in Original D.s and sample
Sampling_mean <- c (Sampling_mean , mean(Data_srswor$age) )
Sampling_SD <- c(Sampling_SD , sd(Data_srswor$age))
cat("Mean and standard deviation of Sample generated using SRSWOR : <" ,  Sampl
  ing_mean[3] , ",",Sampling_SD[3],  ">\n")
```

```
## Mean and standard deviation of Sample generated using SRSWOR : < 38.65278 ,
13.71945 >
```

7.3 Systematic Sampling

Using *systematic sampling* technique we will be selecting **6000** samples out of **32537** records.

```
#Calculate group size
n <- 6000
k <- ceiling(N / n)
#sample size
r <- sample(k, 1)
#generate row numbers
rows <- seq(r, by = k, length = n)

#Draw the sample  Data
Data_Systematic <- na.omit(DataCensus[rows, ])

Sampling_mean <- c( Sampling_mean , mean(Data_Systematic$age))
Sampling_SD <- c(Sampling_SD , sd(Data_Systematic$age))

cat("Mean of Sample generated using Systematic Sampling :" ,  Sampling_mean[4]
   , ",",Sampling_SD[4],  ">\n")
```

```
## Mean of Sample generated using Systematic Sampling : 38.48571 , 13.58098 >
```

7.4 Sampling Result Analysis

For understanding the impact of sampling the sample generated are compared with the original dataset based on mean and standard deviations for age attribute of the dataset.

```
df <- data.frame(cbind(Sampling_method , Sampling_Size , Sampling_SD ,Sampling_
  mean ))

tableFormat(df)
```

Sampling_method	Sampling_Size	Sampling_SD	Sampling_mean
Original Dataset	32537	13.6379835184697	38.5855487598734
Simple Random Sampling With Replacement	12000	13.5832088079001	38.6726666666667
Simple Random Sampling Without Replacement	90000	13.7194515354659	38.6527777777778
Systematic	6000	13.5809840751212	38.4857090171492

From the summary above it's evident that using SRWR, SRSWOR and Systematic sampling despite of different sample sizes there is a very small difference in standard deviation and mean values.

8. Bibliography

Dataset Source: https://www.kaggle.com/ (https://www.kaggle.com/)

External Refrences:

- https://ggplot2.tidyverse.org/ (https://ggplot2.tidyverse.org/)
- https://cran.r-project.org/web/packages/kableExtra/vignettes/awesome_table_in_html.html (https://cran.r-project.org/web/packages/kableExtra/vignettes/awesome_table_in_html.html)
- ~/ggplot2/Tutorial_ggplot2/Rgraphics.html