Exploring Data with R

This tutorial is an attempt to demonstrate how to explore data with R by using summary command and visualization. In this lab, we will be using **customer** dataset which can be downloaded from eLearn@USM.

Load the dataset into R, and name the data frame as custData. This time we will be using a function from readr package which is one of the packages in tidyverse.

```
> library(tidyverse)
```

readr provides several seven read_ functions. Let's use read_delim() which allows separator (delimiter) to be specified. The data is separated by tab. Thus, we specify it as the delim parameter.

```
> custData <- read_delim('cust.data.csv', delim='\t')
```

We can also use read_tsv() which can be used to read tab separated files.

```
> custData <- read_tsv('cust.data.csv')
```

Using summary command

Use summary command to examine the distribution of the dataset.

```
> summary(custData)
```

A summary such as follows will be shown in the console.

```
     custid            sex         is.employed         income                    marital.stat
Min.   :   2068   F:440   Mode :logical   Min.   : -8700   Divorced/Separated:155
1st Qu.: 345667   M:560   FALSE:73        1st Qu.: 14600   Married            :516
Median : 693403           TRUE :599       Median : 35000   Never Married      :233
Mean   : 698500           NA's :328       Mean   : 53505   Widowed            : 96
3rd Qu.:1044606                           3rd Qu.: 67000
Max.   :1414286                           Max.   :615000

health.ins                              housing.type recent.move      num.vehicles
Mode :logical   Homeowner free and clear    :157   Mode :logical   Min.   :0.000
FALSE:159       Homeowner with mortgage/loan:412   FALSE:820       1st Qu.:1.000
TRUE :841       Occupied with no rent       : 11   TRUE :124       Median :2.000
                Rented                      :364   NA's :56        Mean   :1.916
                NA's                        : 56                   3rd Qu.:2.000
                                                                   Max.   :6.000
                                                                   NA's   :56
      age              state.of.res
Min.   :  0.0   California  :100
1st Qu.: 38.0   New York    : 71
Median : 50.0   Pennsylvania: 70
Mean   : 51.7   Texas       : 56
3rd Qu.: 64.0   Michigan    : 52
Max.   :146.7   Ohio        : 51
                (Other)     :600
```

The command will show summary statistics on the numerical columns (attributes) and count statistics on the categorical columns.

As we can see, a third of is.employed values are missing.

Attribute housing.type, recent.move and num.vehicles are missing 56 values

Attribute income has some negative values (might be invalid)

The average value of attribute age seems plausible but the minimum and maximum values seem unlikely.

We can display a specific statistic (such as mean, variance, median, min, max and quantile) on numerical columns

Using Visualization

We will be using a plotting package called ggplot2. The package is loaded when we load tidyverse.

Visualizing a Single Attribute

Histogram bin an attribute into fixed-width groups and returns the number of data points that falls into each group. To plot a histogram.

```
> ggplot(custData, aes(x=age)) + geom_histogram(binwidth=5, fill="gray")
```

ggplot is used to initialize the plot object. We pass the data and specify the attribute that we want to map to the plot i.e. age. Following that, we have to specify the type of plotting that we want to display i.e. a histogram (geom_histogram). As for the histogram, we have to specify the binwidth which is equal to 5.

We can display the histogram according to sex.

```
> ggplot(custData, aes(x=age, fill=sex)) + geom_histogram(binwidth=5)
```

Density plot is a continuous histogram which can be used to examine the overall shape of the curve. The area under the density plot is equal to 1. To plot a density plot, type the following statement.

```
> ggplot(custData, aes(x=income)) + geom_density()
```

Bar chart is a histogram for categorical data. It records the frequency of every value of a categorical data.

```
> ggplot(custData, aes(x=marital.stat)) + geom_bar()
```

Let's specify the color of the bar chart according to marital.stat.

```
> ggplot(custData, aes(x=marital.stat, fill=marital.stat)) + geom_bar()
```

Visualizing Two Attributes

It is important to examine the relationship between two attributes. Here, we will be scatter plot to examine the relationship between two attributes. We will examine the relationship between attribute age and attribute income. Before that, let's calculate the correlation between the attributes.

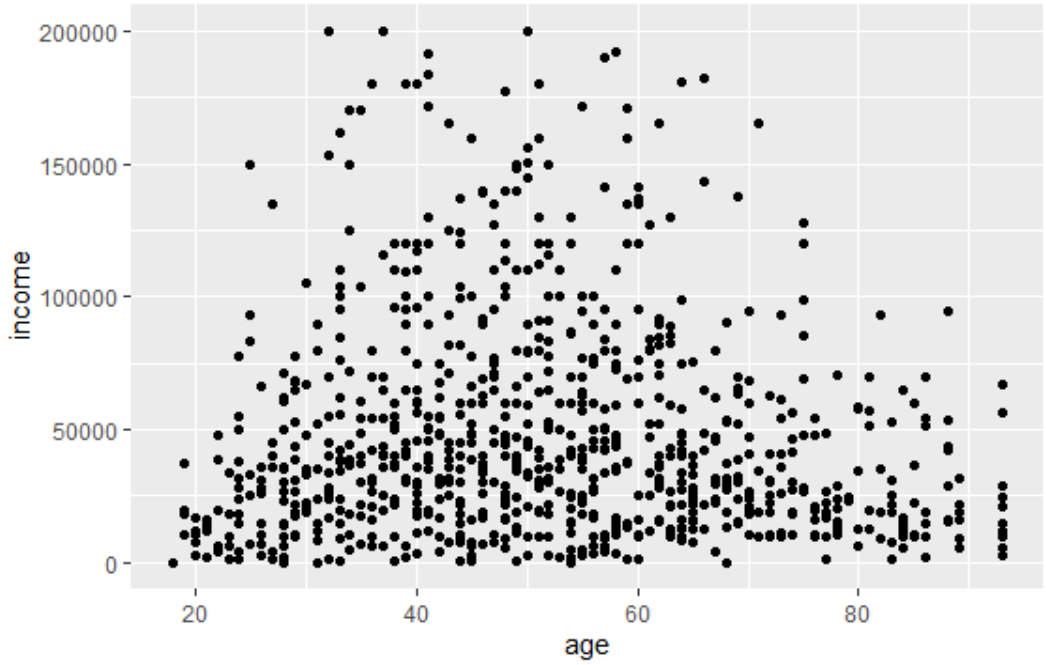
```
> custData2 <- subset(custData, (custData$age > 0 & custData$age < 100 &
  custData$income > 0))
```

```
> cor(custData2$age, custData2$income)
```

We will see that the correlation is -0.02240845 which is a negative correlation. We would expect the variable to have positive correlation since income would increase as we get older.

Let's use scatter plot to gain more insight.

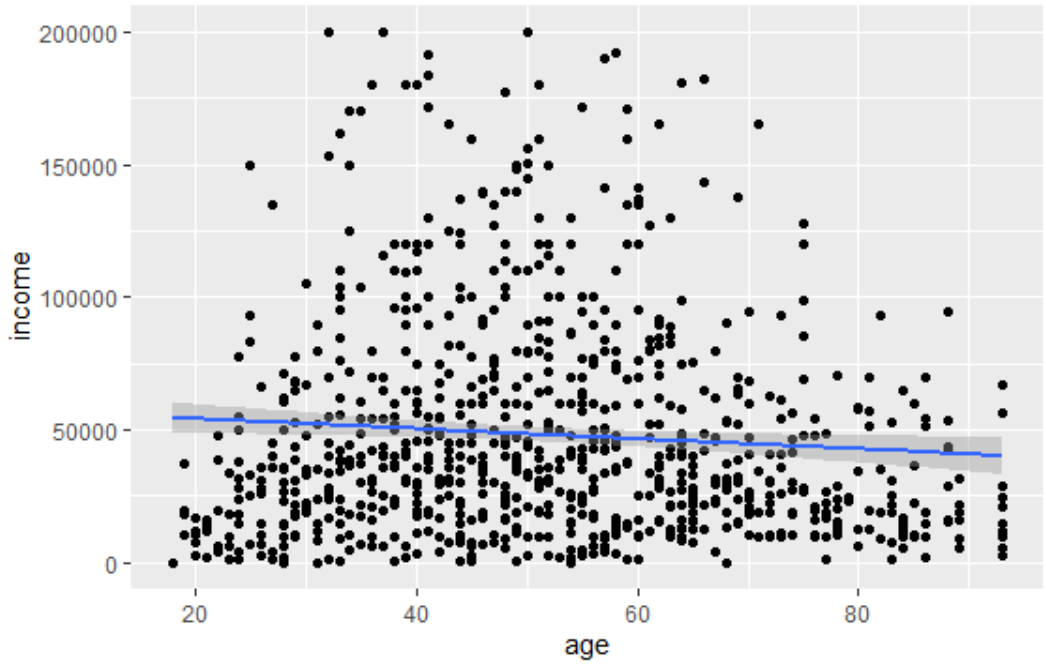
```
> ggplot(custData2, aes(x=age, y=income)) + geom_point() + ylim(0, 200000)
```



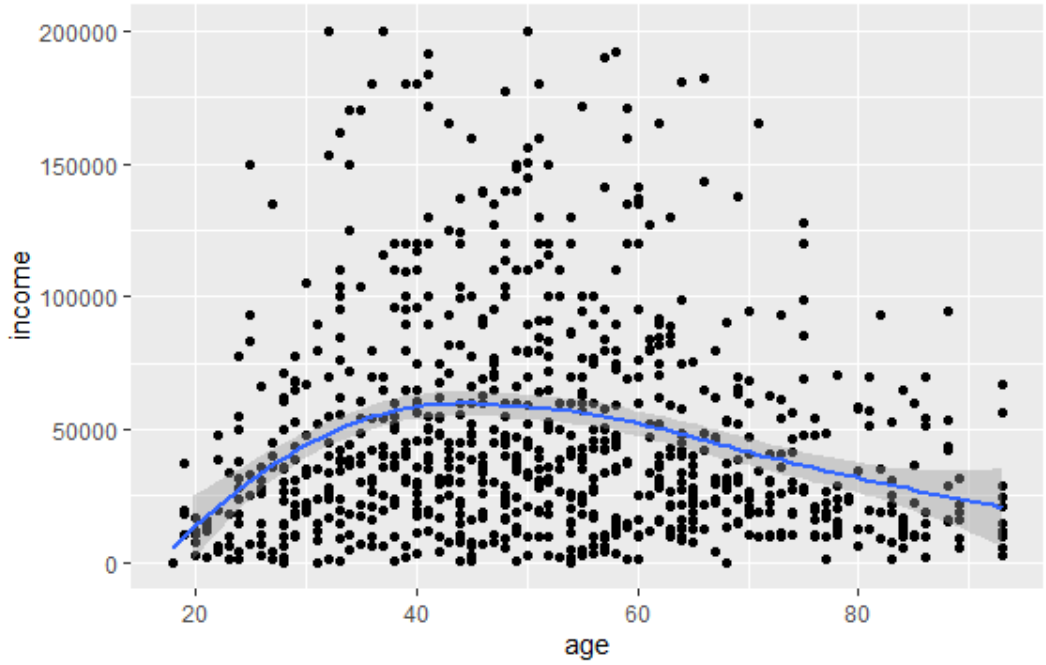
As we can see the income trend is increasing between age 0 and age 50, and the trend is decreasing for age 50 and above.

Let's us fit a linear line through the data.

```
> ggplot(custData2, aes(x=age, y=income)) + geom_point() + ylim(0, 200000)
  + stat_smooth(method='lm')
```



```
> ggplot(custData2, aes(x=age, y=income)) + geom_point() + ylim(0, 200000)
  + geom_smooth()
```

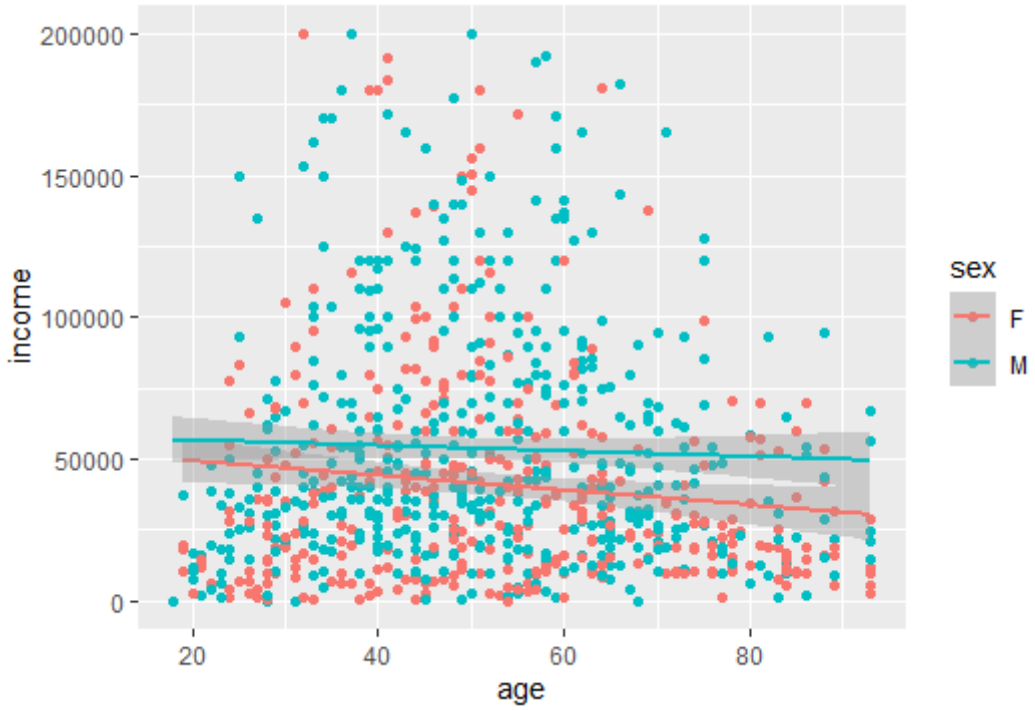



```
> ggplot(custData2, aes(x=age, y=income, color=sex)) + geom_point() +
  ylim(0, 200000)
```


```
> ggplot(custData2, aes(x=age, y=income, shape=sex)) + geom_point() +
  ylim(0, 200000)
```

Let's us fit a linear line through the data. Notice that there are two lines representing the two classes.

```
> ggplot(custData2, aes(x=age, y=income, color=sex)) + geom_point() +
  ylim(0, 200000) + stat_smooth(method='lm')
```




```
> ggplot(custData2, aes(x=age, y=income, color=sex)) + geom_point() +
  ylim(0, 200000) + geom_smooth()
```

Exercises

1. Download boston dataset from eLearn@USM.
2. Load the dataset.
3. Detect any missing value in the dataset.
4. Detect any outlier in column DIS using boxplot
5. Write a programming function for IQR rule. The function should accept first quartile and third quartile as arguments and return the lower and upper bounds as a vector. Use the function to detect any outlier in column LSTAT.
6. Examine the relationship between attribute AGE and attribute TAX. Determine if it is a positive or negative correlation or no correlation.
7. Visualize the relationship between AGE and TAX and fit a linear line through the data. Observe the slope of the linear line.