Introduction to Data Science

Midterm Project

|  |  |  |
| --- | --- | --- |
| Name | ID | Section |
| Amit Podder | 20-42273-1 | C |
| Writhik Banik Barshan | 20-43124-1 | C |

**Topic:** Heart Attack Analysis & Prediction Dataset

**Dataset Description:**

The dataset is about Heart Attack Analysis & Prediction. Datasets like these are typically used for data analysis and machine learning projects to understand the factors that may contribute to heart attack analysis and also predict the likelihood of a heart attack based on certain variables. The dataset contains information about age, sex, chest pain type, resting BP, cholesterol, fasting BS, resting ECG, max HR, exercise angina, old peak, st slope, and heart disease of the patients. In this dataset, we can see four types of chest pain:

ASY : Asymptomatic

TA : Typical Angina

ATA : Atypical Angina

NAP : Non-Anginal Pain

If the fasting blood sugar level is less than 120mg/dl, then the value will be 1 and if the fasting blood sugar level is greater than 120mg/dl, then the value will be 0. Here, 1= true and 0 = false.

It seems the dataset has some missing values as indicated by blanks in certain cells. This data might be used for heart attack analysis and prediction in future heart attack data analysis.

**Import the dataset as CSV and print the dataset:**

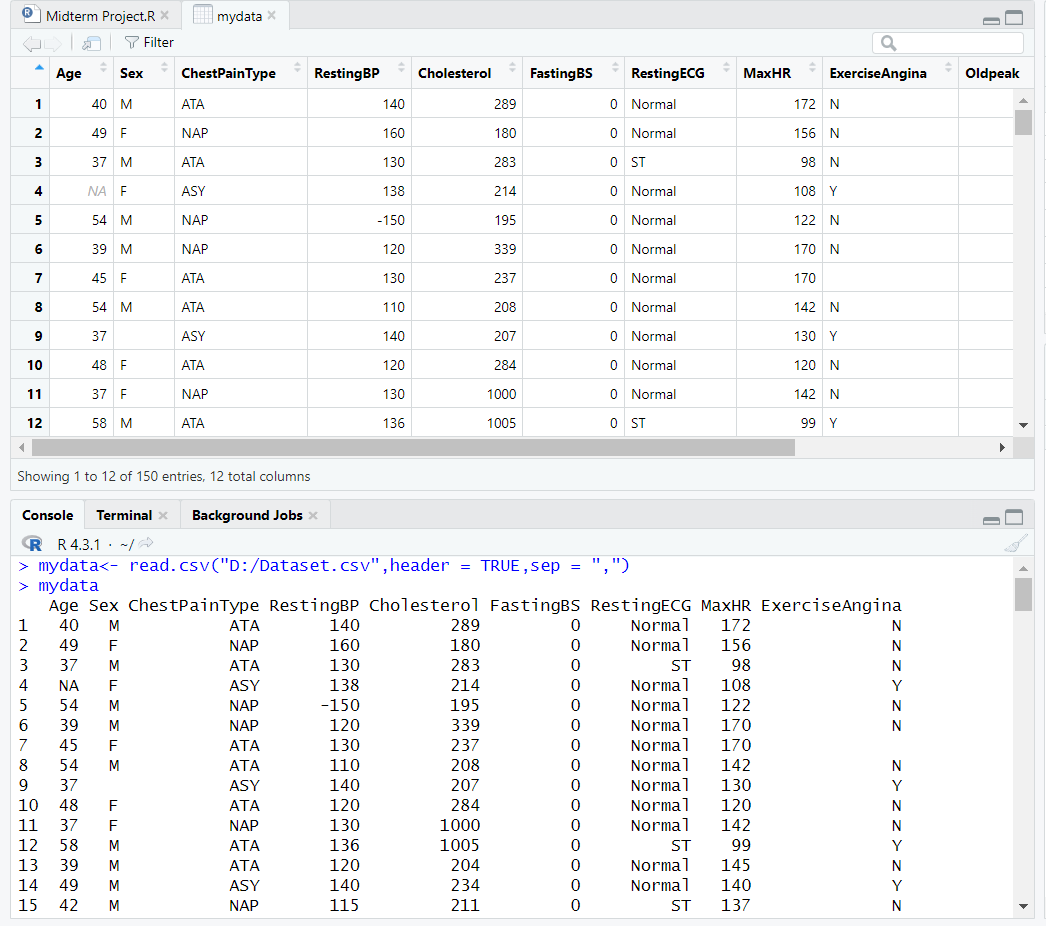
**Code:**

mydata<- read.csv("D:/Dataset.csv",header = TRUE,sep = ",")

mydata



**Output:**



**Description:**

Here we have imported the code of the dataset as a csv file. We can also see the output of the dataset imported in RStudio.

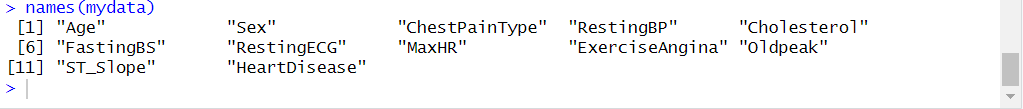
**To see the column name of the dataset:**

**Code:**

names(mydata)



**Output:**



**Description:**

In this code, we can see the column name of the dataset. Here, with the help of the code name(), we can see all the attribute names present in the dataset.

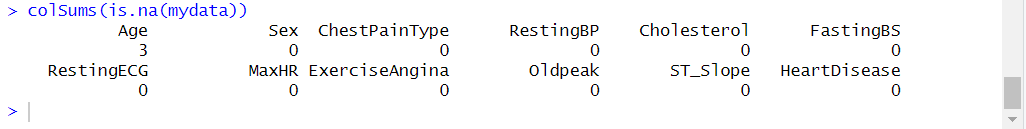
**Finding the Missing(Null) values:**

**Code:**

colSums(is.na(mydata))



**Output:**



**Description:**

In this code, we can see all the null values of the dataset. Here, with the help of the code colSums(is.na()), we can check missing values in each column.

**Finding the specific row number of Missing(Null) Values and remove it from the dataset(for the “Age” attribute):**

**Code:**

which(is.na(mydata$Age))

mydata <- mydata[-4,]

which(is.na(mydata$Age))

mydata <- mydata[-23,]

which(is.na(mydata$Age))

mydata <- mydata[-31,]

which(is.na(mydata$Age))



**Output:**



**Description:**

At first, we found the row numbers where “Age” has missing values. Here, with the help of the code which(is.na(mydata$Age)), we found the row numbers where “Age” has null values. After that, with the help of the code mydata <- mydata[], we removed the row as it has null values in it.

**Finding the Mean and Median value for the “Age” attribute from the dataset:**

**Code:**

mean(mydata$Age)

median(mydata$Age)



**Output:**



**Description:**

After removing the null values from the “Age” attribute, we have calculated the Mean and Median values for the “Age” attribute. At first, with the help of the code mean(mydata$Age), we calculated the Mean value for the “Age” attribute. After that, with the help of the code median(mydata$Age), we calculated the Median value for the “Age” attribute.

**Finding the** **Variance and Standard Deviation value for the “Age” attribute from the dataset:**

**Code:**

var(mydata$Age)

sd(mydata$Age)



**Output:**



**Description:**

After removing the null values from the “Age” attribute, we have calculated the Variance and Standard Deviationvalues for the “Age” attribute. At first, with the help of the code var(mydata$Age), we calculated the Variance for the “Age” attribute. After that, with the help of the code sd(mydata$Age), we calculated the Standard Deviation value for the “Age” attribute.

**Annotating Datasets:**

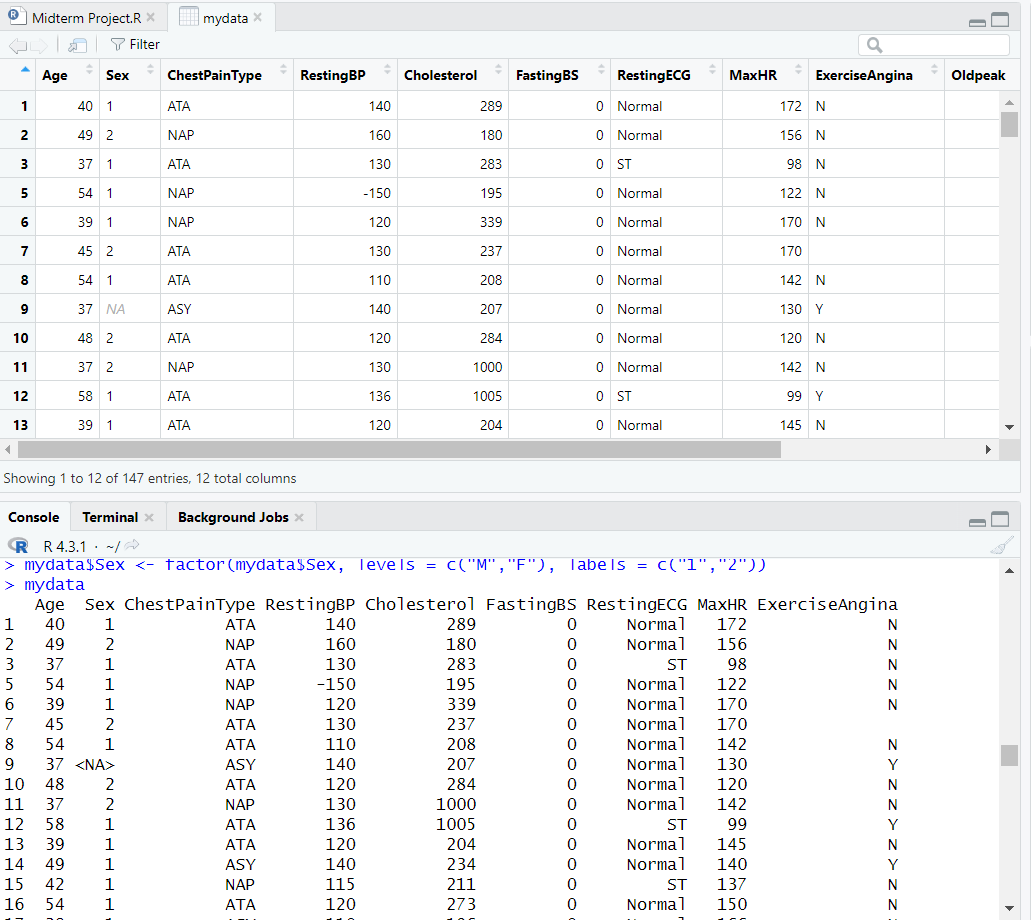
**Code:**

mydata$Sex <- factor(mydata$Sex, levels = c("M","F"), labels = c("1","2"))

mydata



**Output:**



**Description:**

Here, the “Sex” column is converted into numeric(1 and 2) where ‘1’ represents ‘M’ and ‘2’ represents ‘F’. With the help of the code mydata$Sex <- factor(mydata$Sex, levels = c("M","F"), labels = c("1","2")), we were able to successfully converted ‘M’ and ‘F’ into numeric ‘1’ and ‘2’.

**Finding the specific row number of Missing(Null) Values and remove it from the dataset(for the “Sex” attribute):**

**Code:**

which(is.na(mydata$Sex))

mydata <- mydata[-8,]

which(is.na(mydata$Sex))

mydata <- mydata[-23,]

which(is.na(mydata$Sex))

mydata <- mydata[-35,]

which(is.na(mydata$Sex))



**Output:**



**Description:**

At first, we found the row numbers where “Sex” has missing values. Here, with the help of the code which(is.na(mydata$Sex)), we found the row numbers where “Sex” has null values. After that, with the help of the code mydata <- mydata[], we removed the row as it has null values in it.

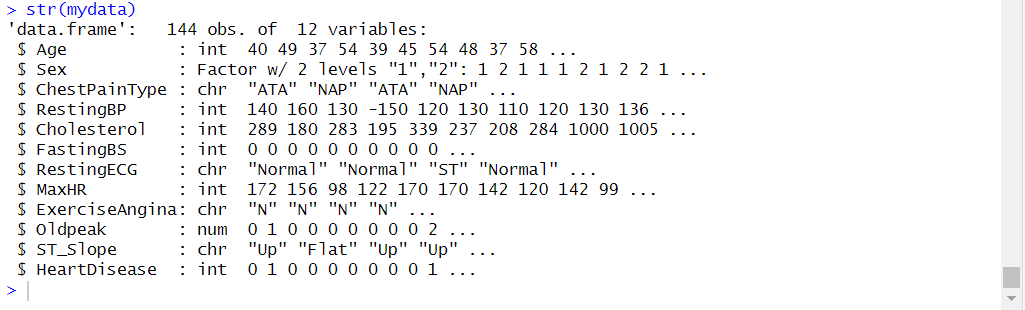
**Summary of the structure of the dataset:**

**Code:**

str(mydata)



**Output:**



**Description:**

The structure of the dataset is displayed with the help of the code str().

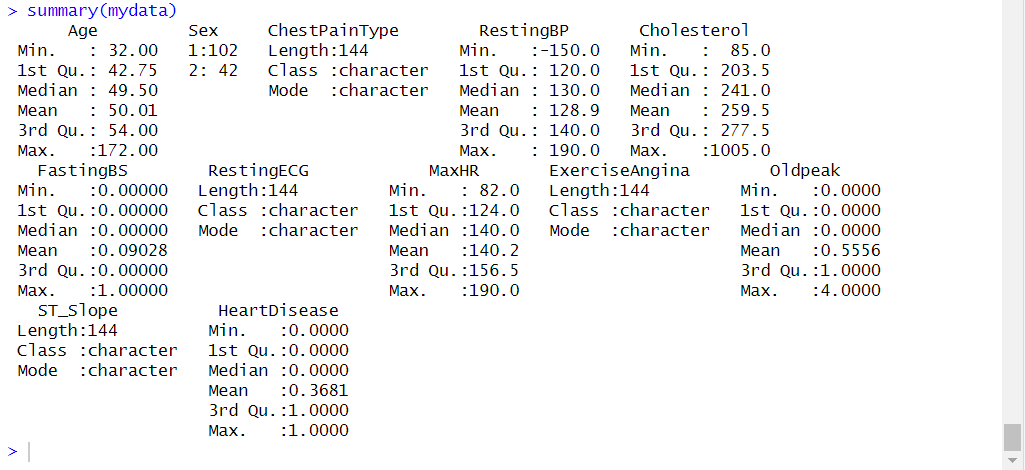
**Descriptive Statistics:**

**Code:**

summary(mydata)



**Output:**



**Description:**

Here, we are using this code to see the descriptive statistics. To see the descriptive statistics, we use the summary() function. We can also see the min, max, mean, and median values of the dataset.

**Summary in Standard Deviation** **of numeric columns in the dataset:**

**Code:**

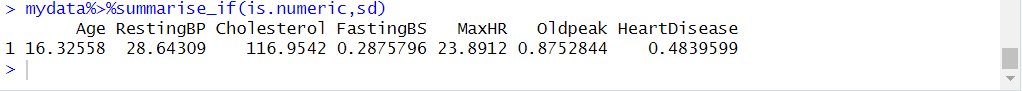
install.packages("dplyr")

library("dplyr")

mydata%>%summarise\_if(is.numeric,sd)



**Output:**



**Description:**

Here, we are using this code to see the standard deviation of numeric columns in the dataset. The standard deviation of numeric columns in the dataset is calculated using the dplyr package.

**Standard Deviation of the values stored in a CSV file:**

**Code:**

s <- mydata$Age

sd(s)

a <- mydata$Cholesterol

sd(a)



**Output:**



**Description:**

Here, we are using this code to directly calculate the standard deviation of a specific numeric column in the dataset. Using the code “s <- mydata$ sd(s)”, the standard deviation of the “Age” and “Cholesterol” columns is directly calculated.

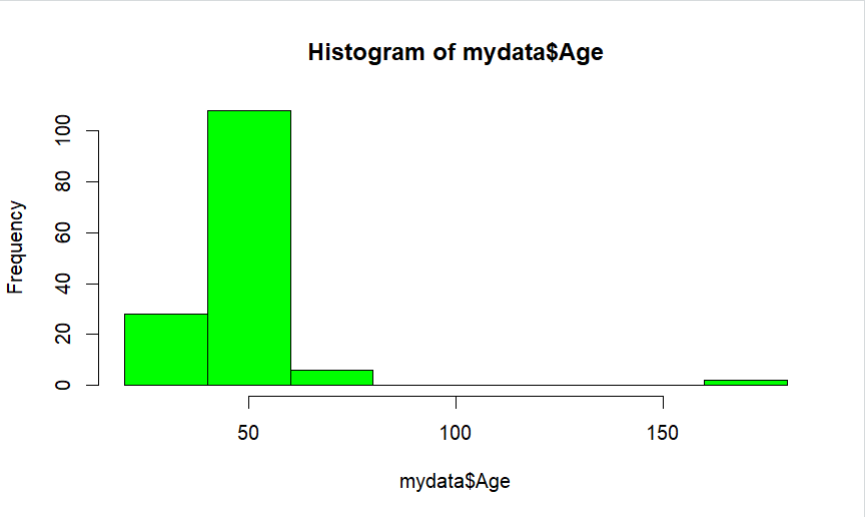
**Histogram:**

**Code:**

hist(mydata$Age, col = "Green")



**Output:**



**Description:**

Here, we are using this code to make a histogram. Using the code “hist(mydata$Age, col = "Green")”, the histogram for the “Age” columns is plotted.

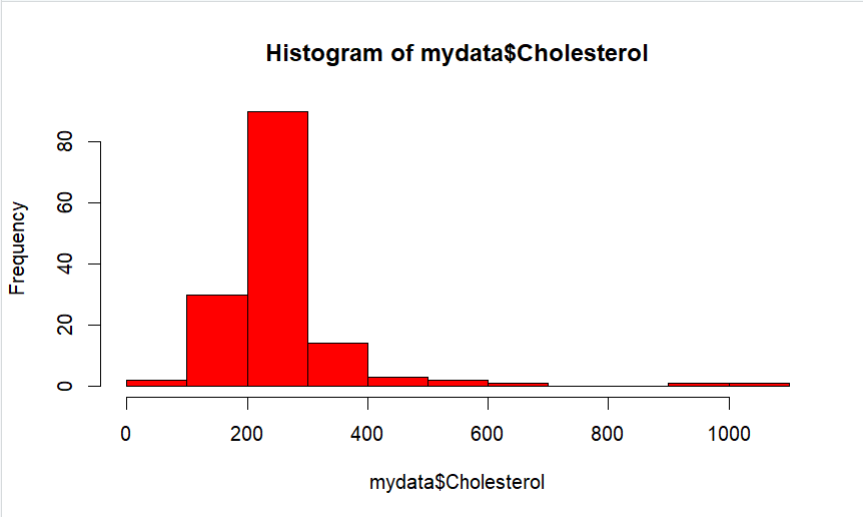
**Histogram:**

**Code:**

hist(mydata$Cholesterol, col = "Red")



**Output:**



**Description:**

Here, we are using this code to make a histogram. Using the code “hist(mydata$Cholesterol, col = "Red")”, the histogram for the “Cholesterol” columns is plotted.