**Project Idea:**

Diamond Dataset represent 54k rows of diamonds with some properties and diamonds price, we need to study this attribute to predict the price of diamond using linear regression model.

**Dataset Description:**

A data frame with 53940 rows and 10 variables

Price: Price in US Dollar.

Carat: Weight of the diamond.

Color: Diamond Color from J(Worst) to D(Best).

Cut: Quality of Cut.

Clarity: a measurement of how clear the diamond is I1(Worst) IF(Best).

X: length in mm.

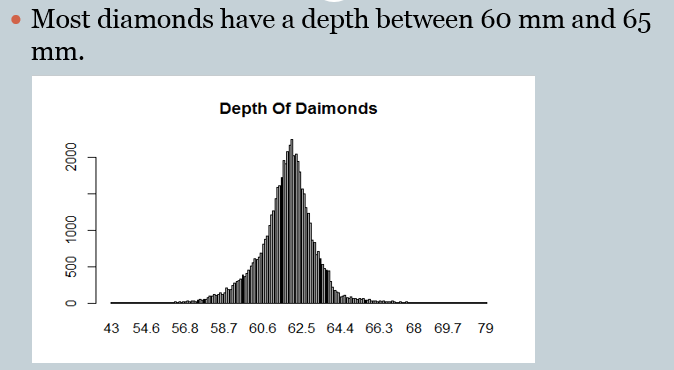
Y: width in mm.

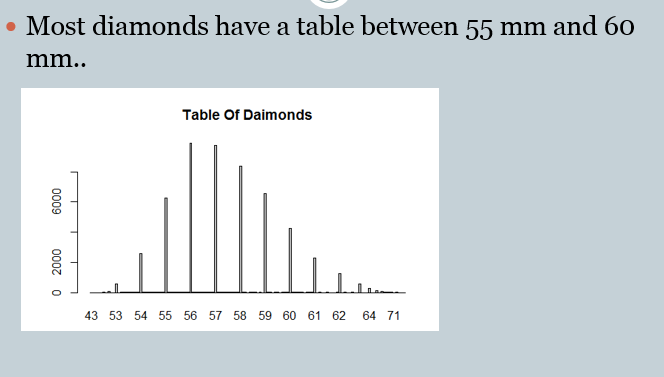
Z: depth in mm.

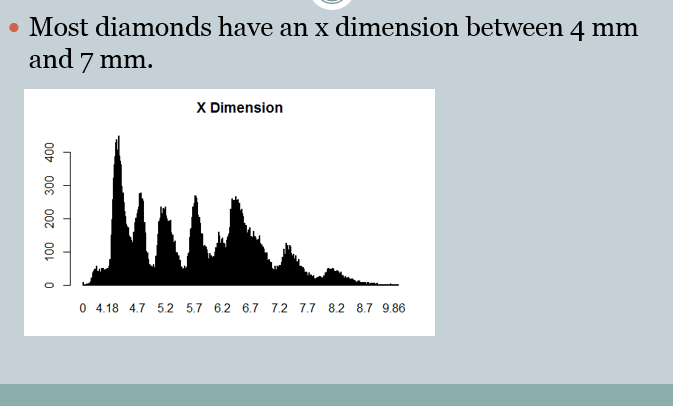
Depth: total depth percentage.

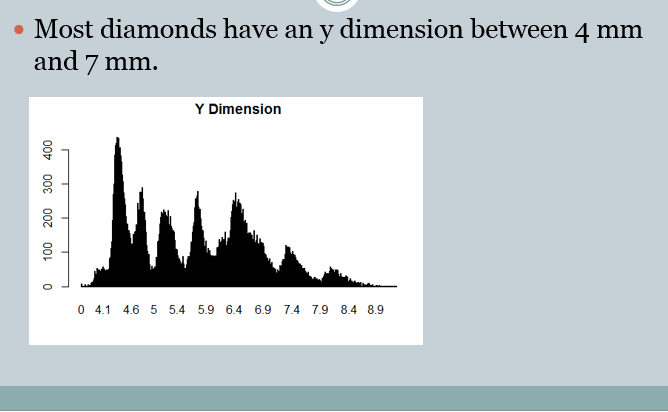
Table: width of top of diamond relative to widest point.

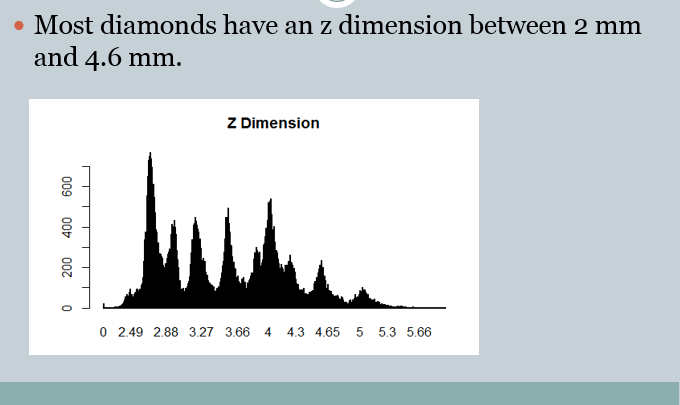
**Dataset Visualization & Analytics:**

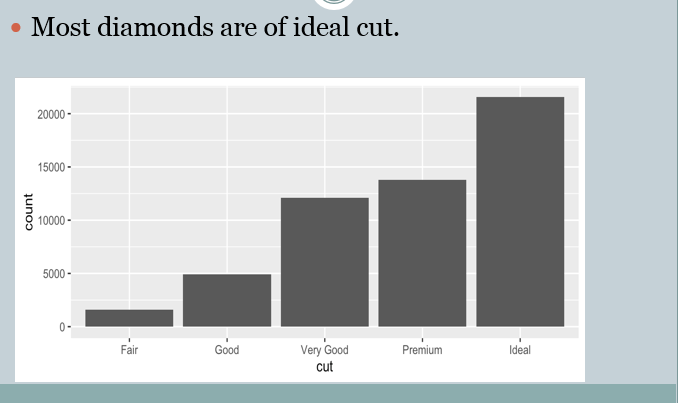


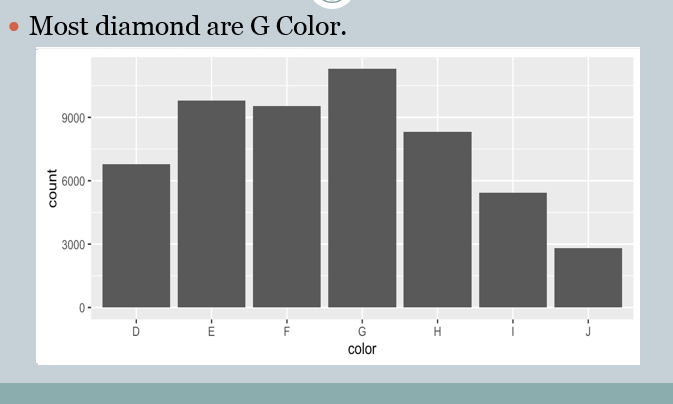


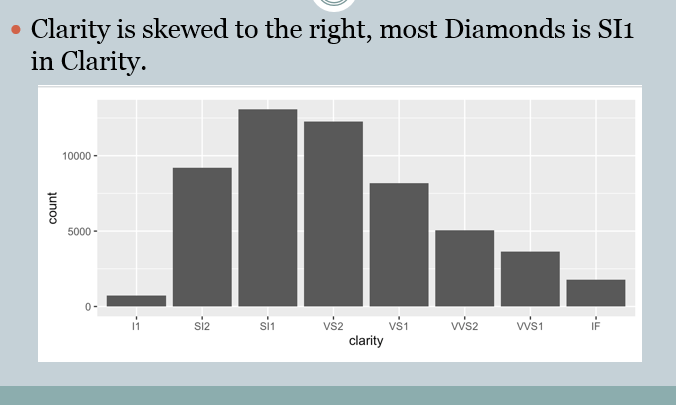


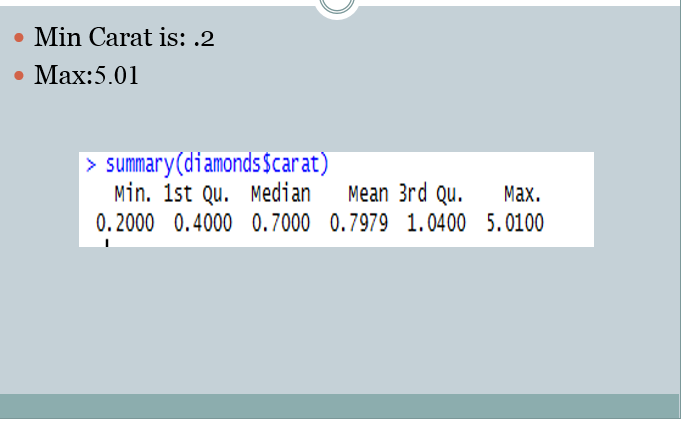


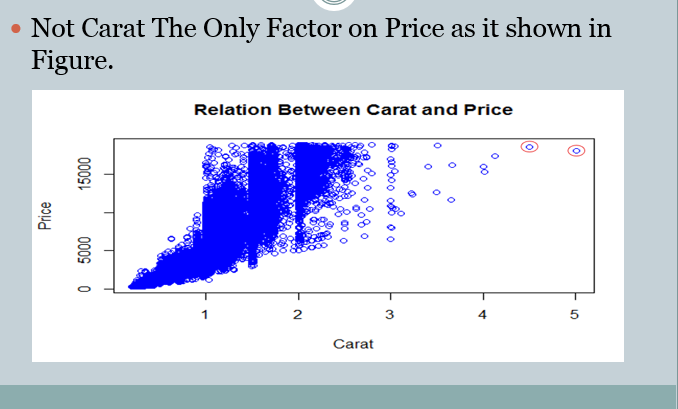


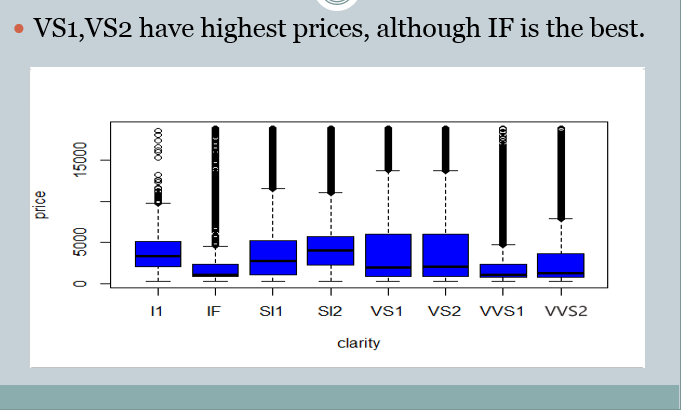


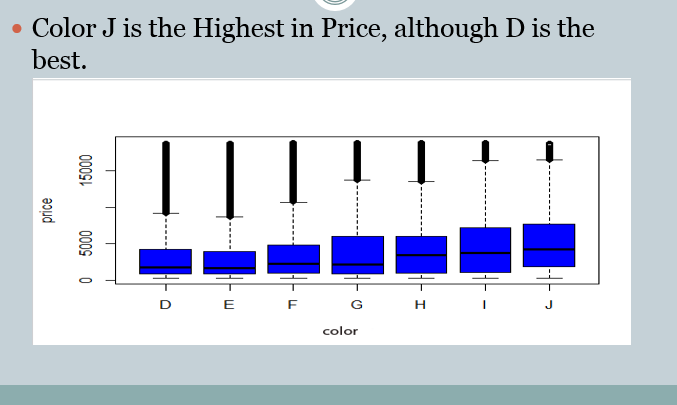


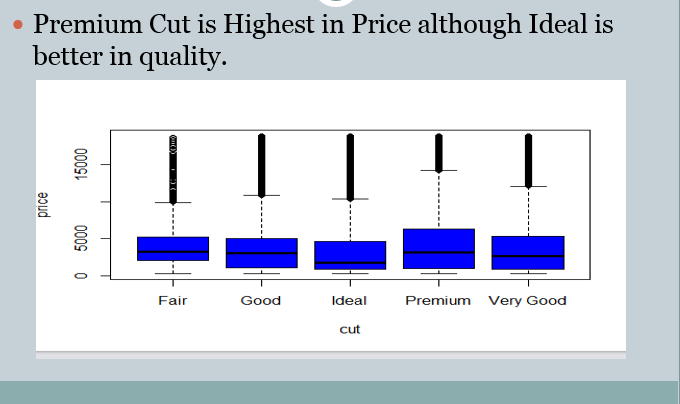


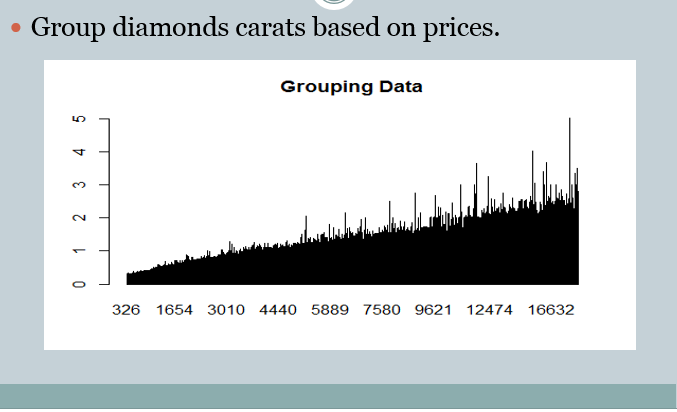












**Used tools:**

R Language.

**Project Code:**

# Set Working Directory

setwd("F:\\")

# Read Data

read.csv("diamonds.csv",sep=",")

# Show Name of Columns

names(diamonds)

# More Details of Columns

str(diamonds)

# Detect Missing Values

NoOfNull<-apply(diamonds,2,function(x)sum(is.na(x)))

NoOfNull

# Show Frequency of Diamonds based on Depth

count<-table(diamonds$depth)

barplot(count,main="Depth Of Diamonds")

# Show Frequency of Diamonds based on Table

count<-table(diamonds$table)

barplot(count,main="Table Of Diamonds")

# Show Frequency of Diamonds based on X Dimension

count<-table(diamonds$x)

barplot(count,main="X Dimension")

# Show Frequency of Diamonds based on Y Dimension

count<-table(diamonds$y)

barplot(count,main="y Dimension")

# Show Frequency of Diamonds based on Z Dimension

count<-table(diamonds$z)

barplot(count,main="Z Dimension")

# Show Frequency of Diamonds based on Cut

count<-table(diamonds$cut)

barplot(count,main="Cut Of Diamonds",ylim=c(0,20000),xlab="Cut",ylab = "Count")

# Show Frequency of Diamonds based on Color

count<-table(diamonds$color)

barplot(count,main="Color Of Diamonds",ylim=c(0,9000),xlab="Color",ylab = "Count")

# Show Frequency of Diamonds based on Clarity

count<-table(diamonds$clarity)

barplot(count,main="Clarity Of Diamonds",ylim=c(0,10000),xlab="Clarity",ylab = "Count")

# Show Details of Cart

summary(diamonds$carat)

# show Price based on Carat

plot(diamonds$carat,diamonds$price,col="blue")

# show Price of diamond based on clarity

plot(diamonds$clarity,diamonds$price,xlab="clarity",ylab="price",col="blue",main="show price of diamond based on clarity")

# show Price of diamond based on color

plot(diamonds$color,diamonds$price,xlab="Color",ylab="Price",col="blue",main="show price of diamond based on color")

# show Price of diamond based on cut

plot(diamonds$cut,diamonds$price,xlab="Cut",ylab="price",col="blue",main="show price of diamond based on cut")

# Group Carats based on Prices

avgcarat<-aggregate(diamonds$carat,list(diamonds$price),mean)

avgcarat

barplot(avgcarat$x,names.arg = avgdiamond$Group.1,main = "Frequency of carat")

# Copy Data

diamondsTmp<-diamonds

# Convert Columns

diamondsTmp$color<-as.integer(diamonds$color)

diamondsTmp$cut<-as.integer(diamonds$cut)

diamondsTmp$clarity<-as.integer(diamonds$clarity)

# Calculate Correlation

cor(diamonds$price,diamondsTmp$carat)

cor(diamonds$price,diamondsTmp$cut)

cor(diamonds$price,diamondsTmp$color)

cor(diamonds$price,diamondsTmp$clarity)

cor(diamonds$price,diamondsTmp$depth)

cor(diamonds$price,diamondsTmp$table)

cor(diamonds$price,diamondsTmp$x)

cor(diamonds$price,diamondsTmp$y)

cor(diamonds$price,diamondsTmp$y)

# Split Data

ind<-sample(2,nrow(diamonds),prob = c(.8,.2),replace=T)

# Get Training Set

train.data<-diamonds[ind==1,]

# Get Testing Set

test.data<-diamonds[ind==2,]

# Build Model

regressor<-lm(formula = price~carat+x+y+z,data=train.data)

# Summary Of Model

summary(regressor)

# Calculate Accuracy

accuracy(predict(regressor,test.data),test.data["price"])

**Libraries:**

1-Metric