**Course Code: ITA0447**

**Course Title: STATISTICS WITH R PROGRAMMING FOR NLP**

**LAB DAY : 05**

1. Explore the air quality dataset. It contains daily air quality measurements from New York during a period of five months: • Ozone: mean ozone concentration (ppb), • Solar.R: solar radiation (Langley), • Wind: average wind speed (mph), • Temp: maximum daily temperature in degrees Fahrenheit, • Month: numeric month (May=5, June=6, and so on),• Day: numeric day of the month (1-31).

(i)Get the Summary Statistics of air quality dataset

(ii)Melt airquality data set and display as a long – format data?

(iii)Melt airquality data and specify month and day to be “ID variables”?

(iv)Cast the molten airquality data set with respect to month and date features

(v) Use cast function appropriately and compute the average of Ozone, Solar.R , Wind and temperature per month?

CODE:

#subdiv1

data(airquality)

summary(airquality)

#subdiv2

library(reshape2)

melted\_data <- melt(airquality)

melted\_data

#subdiv3

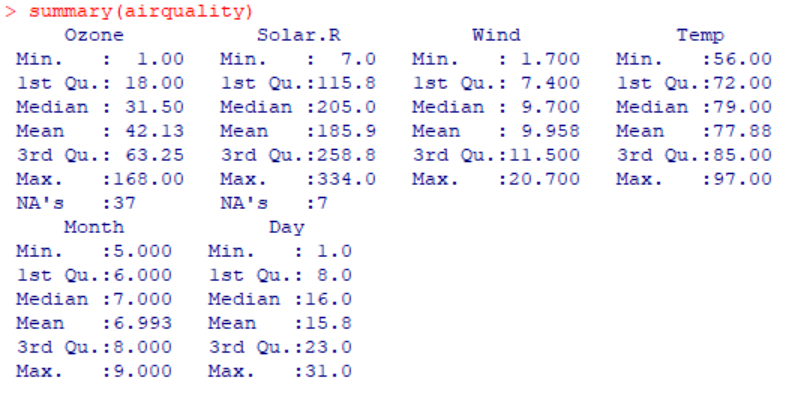
melted\_data <- melt(airquality, id.vars = c("Month", "Day"))

melted\_data

#subdiv4

casted\_data <- dcast(melted\_data, Month + Day ~

OUTPUT:



1. i)Find any missing values(na) in features and drop the missing values if its less than 10% else replace that with mean of that feature.

(ii) Apply a linear regression algorithm using Least Squares Method on “Ozone” and “Solar.R” (iii)Plot Scatter plot between Ozone and Solar and add regression line created by above model

CODE:

#subdiv1

data(airquality)

missing\_values <- colSums(is.na(airquality))

percent\_missing <- missing\_values / nrow(airquality) \* 100

ifelse(percent\_missing < 10, airquality <- na.omit(airquality), )

ifelse(percent\_missing >= 10, airquality[is.na(airquality)] <- mean(airquality, na.rm = TRUE), )

#subdiv2

data(airquality)

model <- lm(Ozone ~ Solar.R, data = airquality)

summary(model)

#subdiv3

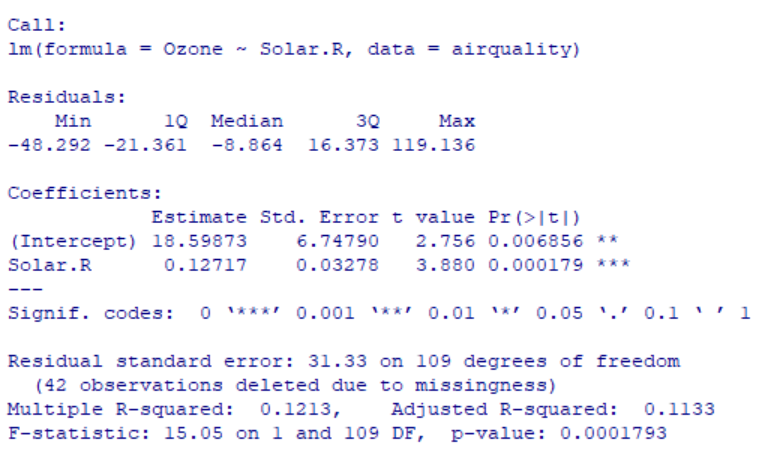
data(airquality)

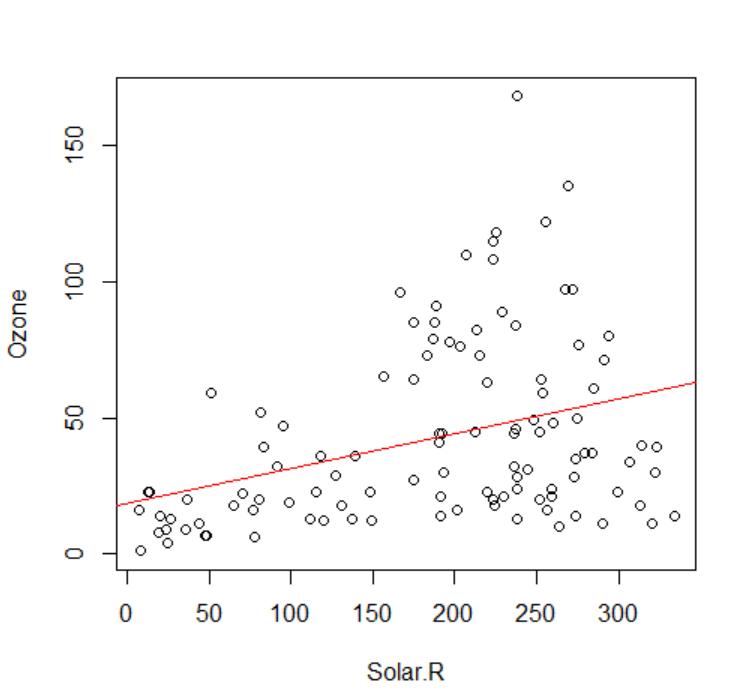
model <- lm(Ozone ~ Solar.R, data = airquality)

plot(Ozone ~ Solar.R, data = airquality)

abline(model, col = "red")

OUTPUT:





1. Load dataset named ChickWeight, ( i).Order the data frame, in ascending order by feature name “weight” grouped by feature “diet” and Extract the last 6 records from order data frame. (ii).a Perform melting function based on “Chick", "Time", "Diet" features as ID variables b. Perform cast function to display the mean value of weight grouped by Diet c. Perform cast function to display the mode of weight grouped by Diet

CODE:

#subdiv1

data(ChickWeight)

ordered\_data <- ChickWeight[order(ChickWeight$weight), ]

tail(ordered\_data, 6)

#subdiv2

data(ChickWeight)

melted\_data <- melt(ChickWeight, id.vars = c("Chick", "Time", "Diet"))

head(melted\_data)

#subdiv3

data(ChickWeight)

melted\_data <- melt(ChickWeight, id.vars = c("Chick", "Time", "Diet"))

mean\_weight\_by\_diet <- cast(melted\_data, Diet ~ variable, mean)

mean\_weight\_by\_diet

#divc

data(ChickWeight)

melted\_data <- melt(ChickWeight, id.vars = c("Chick", "Time", "Diet"))

get\_mode <- function(x) {

ux <- unique(x)

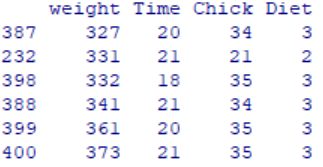
ux[which.max(tabulate(match(x, ux)))]

}

mode\_weight\_by\_diet <- cast(melted\_data, Diet ~ variable, get\_mode)

mode\_weight\_by\_diet

OUTPUT:



1. a. Create Box plot for “weight” grouped by “Diet”

b. Create a Histogram for “weight” features belong to Diet- 1 category

c. Create Scatter plot for “weight” vs “Time” grouped by Diet

CODE:

#subdiv1

library(ggplot2)

data(ChickWeight)

ggplot(ChickWeight, aes(x = factor(Diet), y = weight)) +

geom\_boxplot() +

labs(x = "Diet", y = "Weight") +

ggtitle("Weight Distribution by Diet")

#subdiv2

ggplot(ChickWeight[ChickWeight$Diet == 1, ], aes(x = weight)) +

geom\_histogram(binwidth = 10, color = "black", fill = "white") +

labs(x = "Weight", y = "Count") +

ggtitle("Weight Distribution for Diet-1")

#subdiv3

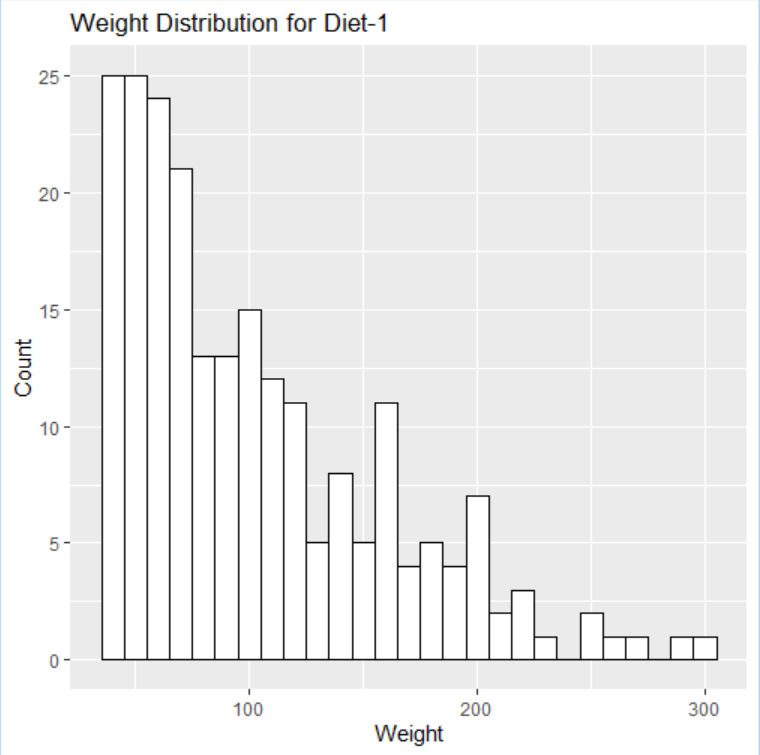
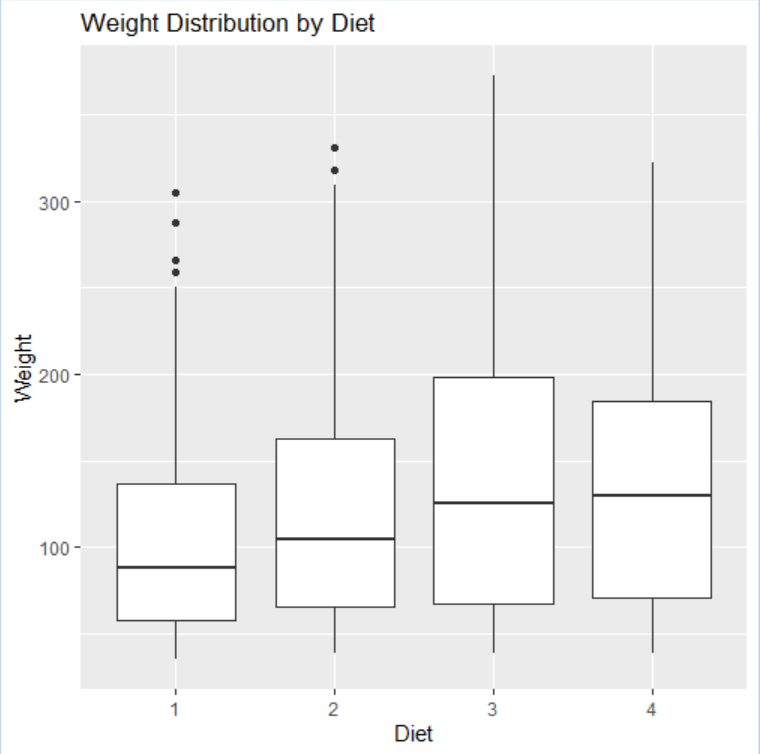
ggplot(ChickWeight, aes(x = Time, y = weight, color = factor(Diet))) +

geom\_point() +

labs(x = "Time", y = "Weight") +

ggtitle("Weight vs Time by Diet")

OUTPUT:





1. a. Create Box plot for “weight” grouped by “Diet”

b. Create a Histogram for “weight” features belong to Diet- 1 category

c. Create Scatter plot for “weight” vs “Time” grouped by Diet.

CODE:

#sub\_div\_1

library(ggplot2)

data(ChickWeight)

ggplot(ChickWeight, aes(x = factor(Diet), y = weight)) +

geom\_boxplot() +

labs(x = "Diet", y = "Weight") +

ggtitle("Weight Distribution by Diet")

#subdiv2

ggplot(subset(ChickWeight, Diet == 1), aes(x = weight)) +

geom\_histogram(binwidth = 10, color = "black", fill = "white") +

labs(x = "Weight", y = "Count") +

ggtitle("Weight Distribution for Diet-1")

#subdiv3

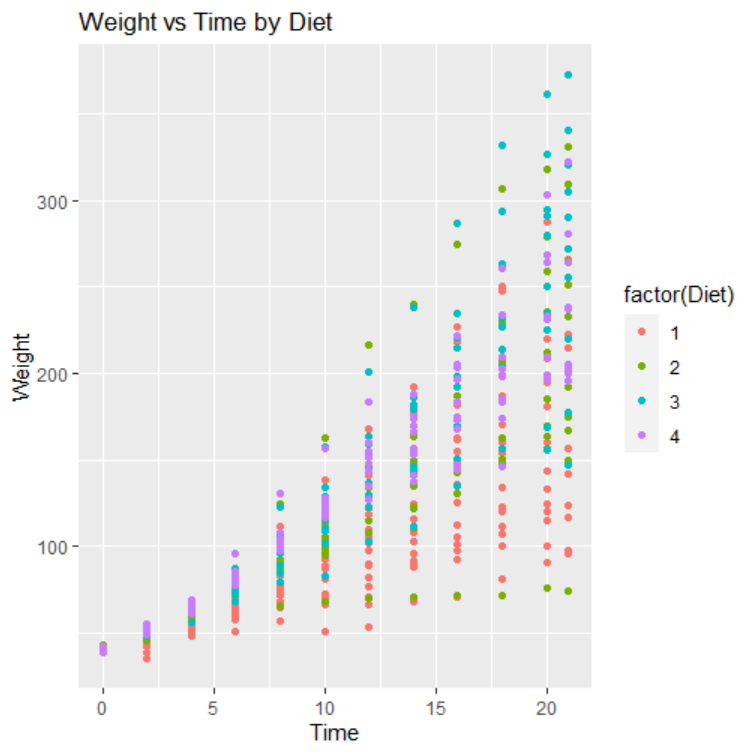
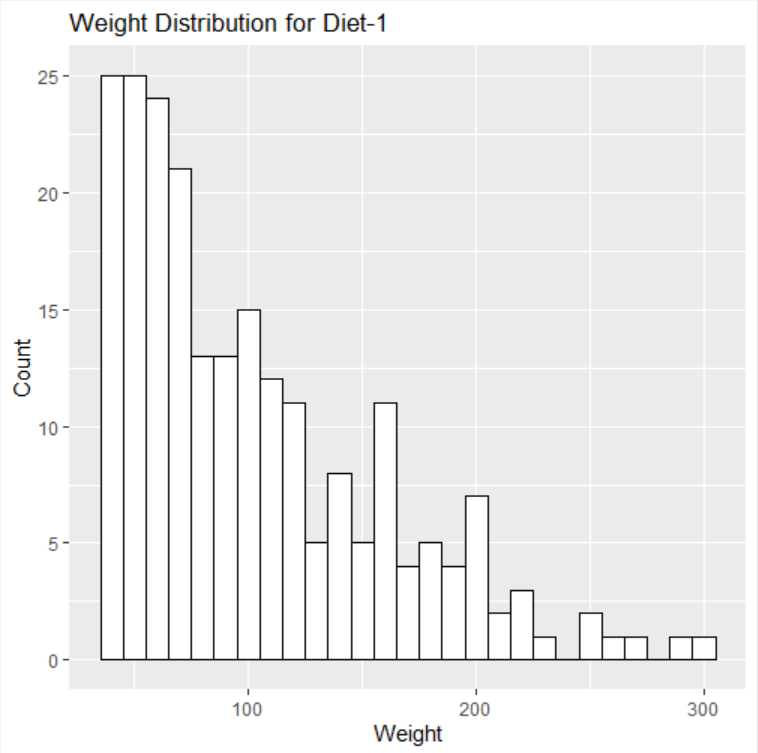
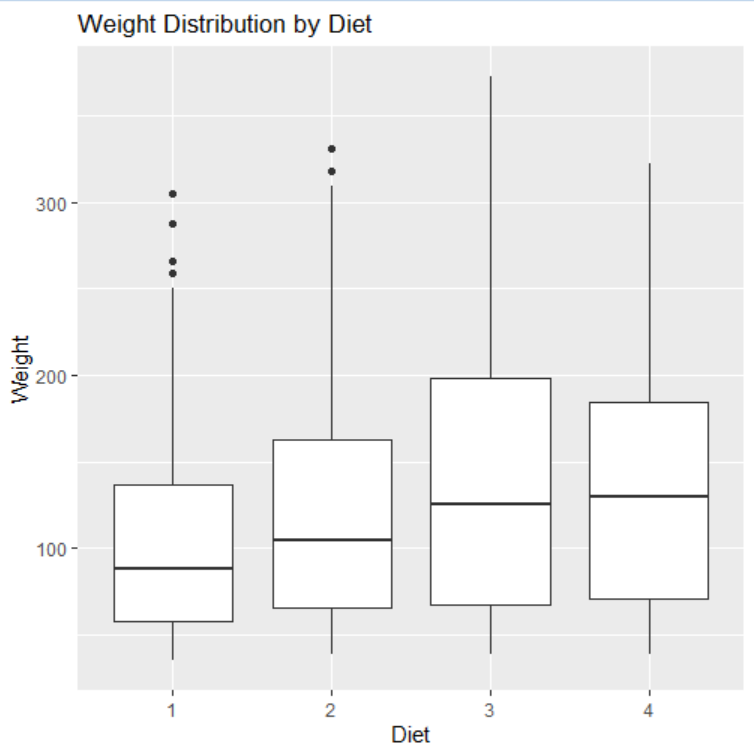
ggplot(ChickWeight, aes(x = Time, y = weight, color = factor(Diet))) +

geom\_point() +

labs(x = "Time", y = "Weight") +

ggtitle("Weight vs Time by Diet")

OUTPUT:



1. For this exercise, use the (built-in) dataset Titanic. a. Draw a Bar chart to show details of “Survived” on the Titanic based on passenger Class b. Modify the above plot based on gender of people who survived c. Draw histogram plot to show distribution of feature “Age”

CODE:

#subdiv1

data(Titanic)

library(ggplot2)

ggplot(data = as.data.frame(Titanic), aes(x = Class, y = Freq, fill = Survived)) +

geom\_bar(stat = "identity", position = "dodge") +

labs(x = "Passenger Class", y = "Count", fill = "Survived") +

ggtitle("Titanic Survival by Passenger Class")

#subdiv2

data(Titanic)

ggplot(data = as.data.frame(Titanic), aes(x = Class, y = Freq, fill = Sex)) +

geom\_bar(stat = "identity", position = "dodge") +

labs(x = "Passenger Class", y = "Count", fill = "Gender") +

ggtitle("Titanic Survival by Gender and Class") +

theme(legend.position = "top")

#subdiv3

data(Titanic)

ggplot(data = as.data.frame(Titanic), aes(x = Age, fill = Survived)) +

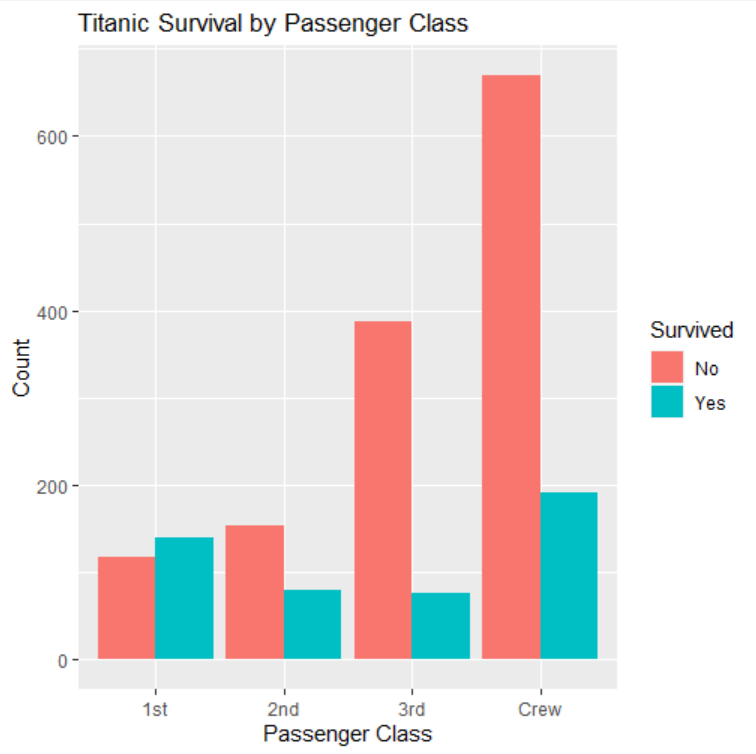
geom\_histogram(binwidth = 5, alpha = 0.7, position = "identity") +

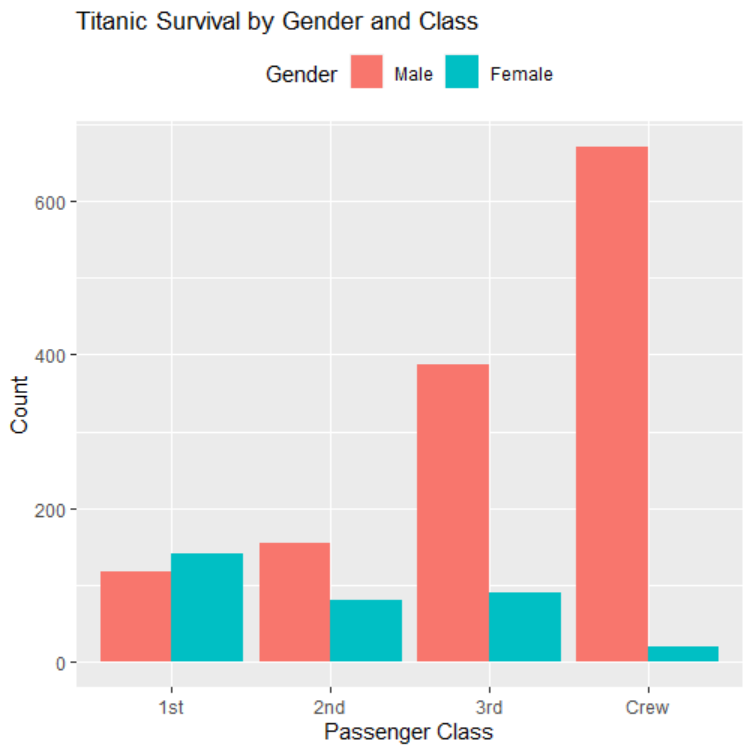
labs(x = "Age", y = "Count", fill = "Survived") +

ggtitle("Age Distribution on Titanic") +

theme(legend.position = "top")

OUTPUT:





1. Explore the USArrests dataset, contains the number of arrests for murder, assault, and rape for each of the 50 states in 1973. It also contains the percentage of people in the state who live in an urban area.
2. a. Explore the summary of Data set, like number of Features and its type. Find the number of records for each feature. Print the statistical feature of data

b. Print the state which saw the largest total number of rape

c. Print the states with the max & min crime rates for murder

(ii) a. Find the correlation among the features

b. Print the states which have assault arrests more than median of the country

c. Print the states are in the bottom 25% of murder

iii) a. Create a histogram and density plot of murder arrests by US stat

b. Create the plot that shows the relationship between murder arrest rate and proportion of the population that is urbanised by state. Then enrich the chart by adding assault arrest rates (by colouring the points from blue (low) to red (high)).

c. Draw a bar graph to show the murder rate for each of the 50 states.

CODE:

#subdiv1

data(USArrests)

summary(USArrests)

str(USArrests)

data(USArrests)

USArrests[which.max(USArrests$Rape), "State"]

data(USArrests)

max\_murder\_states <- USArrests[which.max(USArrests$Murder), "State"]

min\_murder\_states <- USArrests[which.min(USArrests$Murder), "State"]

cat("State(s) with maximum murder rate:", max\_murder\_states, "\n")

cat("State(s) with minimum murder rate:", min\_murder\_states, "\n")

#subdiv2

data(USArrests)

cor(USArrests)

data(USArrests)

median\_assault <- median(USArrests$Assault)

USArrests[USArrests$Assault > median\_assault, "State"]

data(USArrests)

bottom25\_murder <- quantile(USArrests$Murder, 0.25)

USArrests[USArrests$Murder < bottom25\_murder, "State"]

#subdiv3

data(USArrests)

hist(USArrests$Murder, main = "Histogram of Murder Arrests by US State", xlab = "Murder Arrests")

lines(density(USArrests$Murder), col = "red")

data(USArrests)

library(ggplot2)

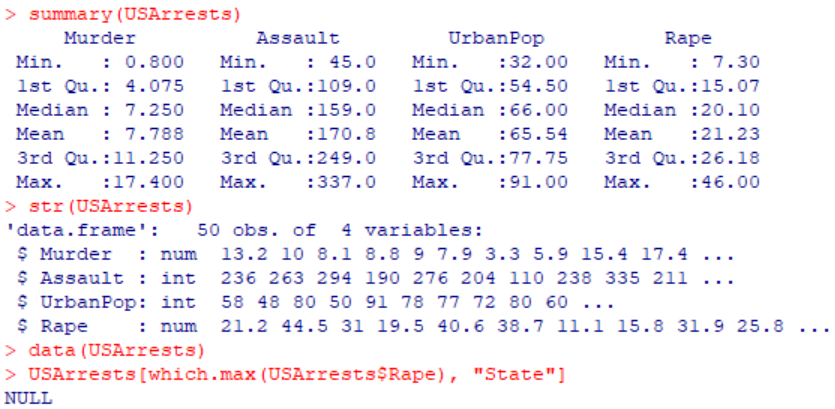
ggplot(USArrests, aes(UrbanPop, Murder)) +

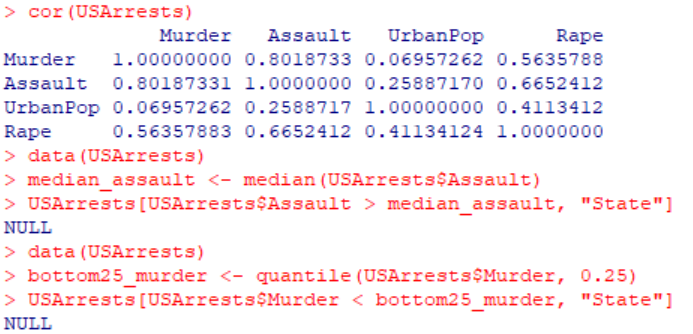
geom\_point(aes(colour = Assault), size = 3) +

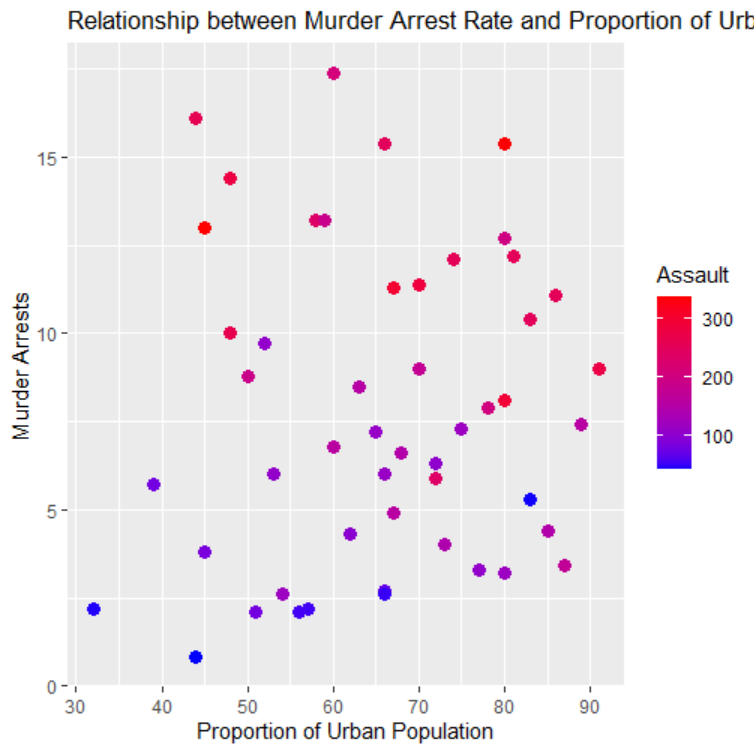
scale\_colour\_gradient(low = "blue", high = "red") +

labs(title = "Relationship between Murder Arrest Rate and Proportion of Urban Population by State", x = "Proportion of Urban Population", y = "Murder Arrests")

OUTPUT:







**8**. a. Create a data frame based on below table.

| Month | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Spends | 1000 | 4000 | 5000 | 4500 | 3000 | 4000 | 9000 | 11000 | 15000 | 12000 | 7000 | 3000 |
| Sales | 9914 | 40487 | 54324 | 50044 | 34719 | 42551 | 94871 | 118914 | 158484 | 131348 | 78504 | 36284 |

b. Create a regression model for that data frame table to show the amount of sales(Sales) based on the how much the company spends (Spends) in advertising

c. Predict the Sales if Spend=13500

CODE:

#subdiv1

df <- data.frame(

Month = 1:12,

Spends = c(1000, 4000, 5000, 4500, 3000, 4000, 9000, 11000, 15000, 12000, 7000, 3000),

Sales = c(9914, 40487, 54324, 50044, 34719, 42551, 94871, 118914, 158484, 131348, 78504, 36284)

)

df

#subdiv2

model <- lm(Sales ~ Spends, data = df)

summary(model)

#subdiv3

new\_data <- data.frame(Spends = 13500)

prediction <- predict(model, newdata = new\_data)

prediction

OUTPUT:

