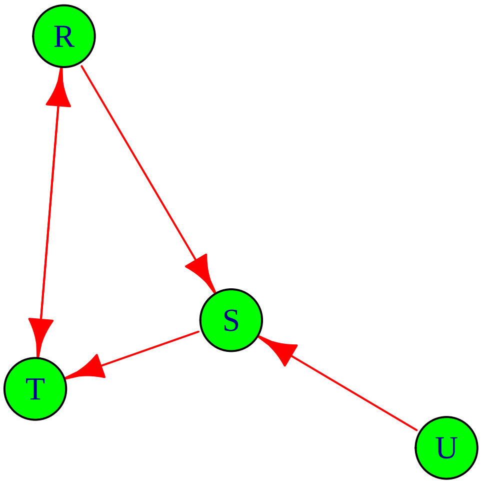
33-Final.R

Suman Paudel

2024-06-27

```
# group b

# question number 6

# load the necessary library
library(igraph, warn.conflicts = F)
# library(Rgraphviz, warn.conflicts = F)

# define the graph object
# a
g1 <- graph(c("R","S","S","T","T","R","R","T","U", "S"))


# b
plot(g1, vertex.color='green',vertex.size=30,edge.size=5,edge.color='red')
```



```
# c
# degree
igraph::degree(g1)
```

```
## R S T U
## 3 3 3 1
```

```r
# closeness
closeness(g1,mode = 'in')
```

```
##         R         S         T         U
## 0.1666667 0.2500000 0.2500000       NaN
```

```r
# betweeness
betweenness(g1)
```

```
## R S T U
## 1 2 2 0
```

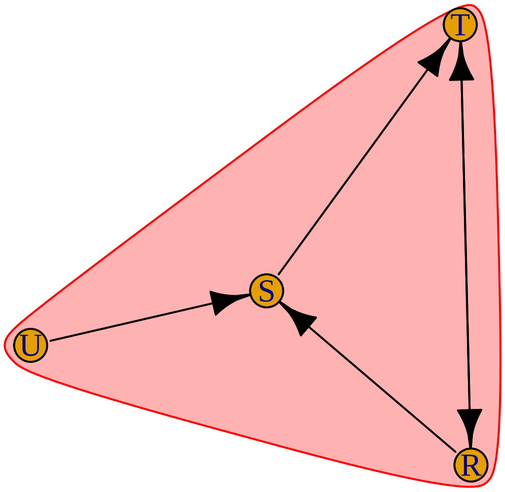
```r
#d
# Identify hubs in the graph
hubs <- which(degree(g1) == max(degree(g1)))
hubs
```

```
## R S T
## 1 2 3
```

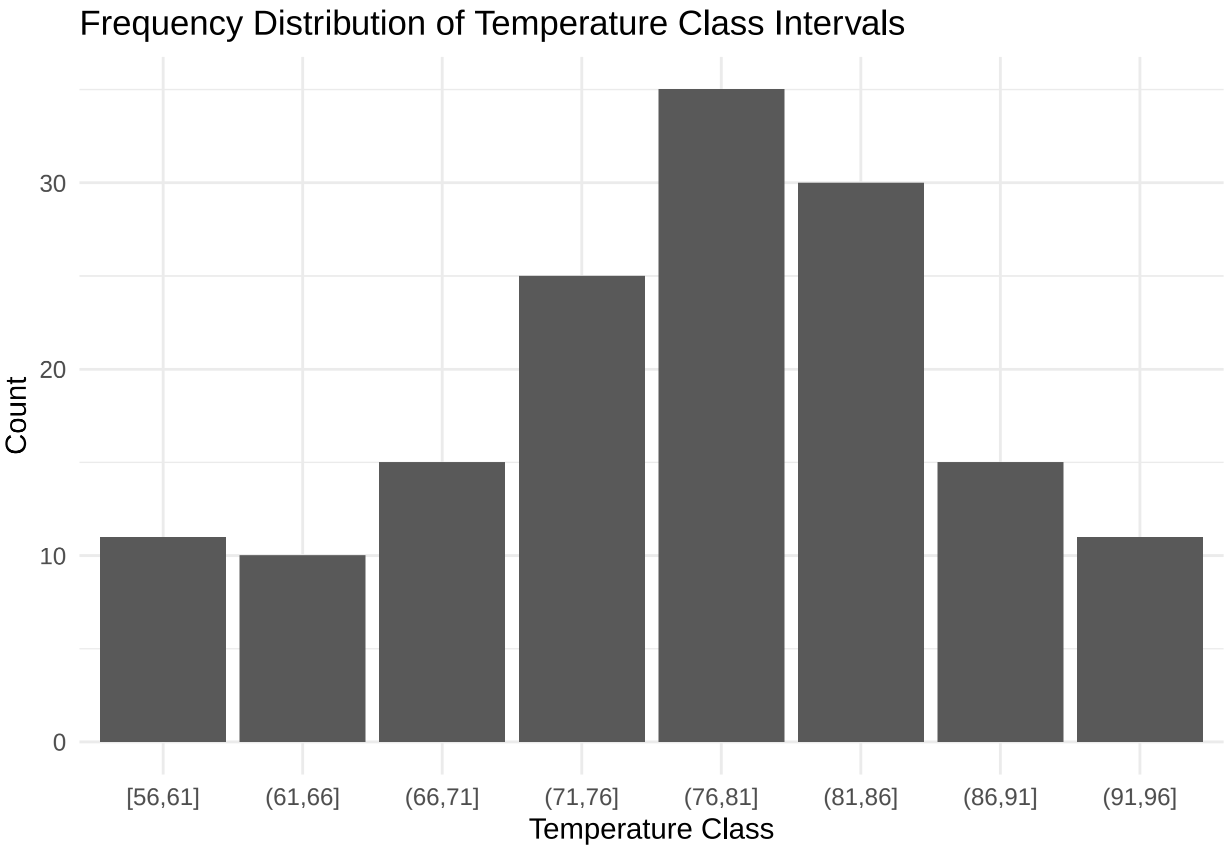
```r
#Hubs in a graph refer to nodes with high connectivity or degree that
# serve as central points of the network.

# Find communities in the graph
communities <- cluster_walktrap(g1)
cat("Number of communities: ", length(communities), "\n")
```

```
## Number of communities:  1
```

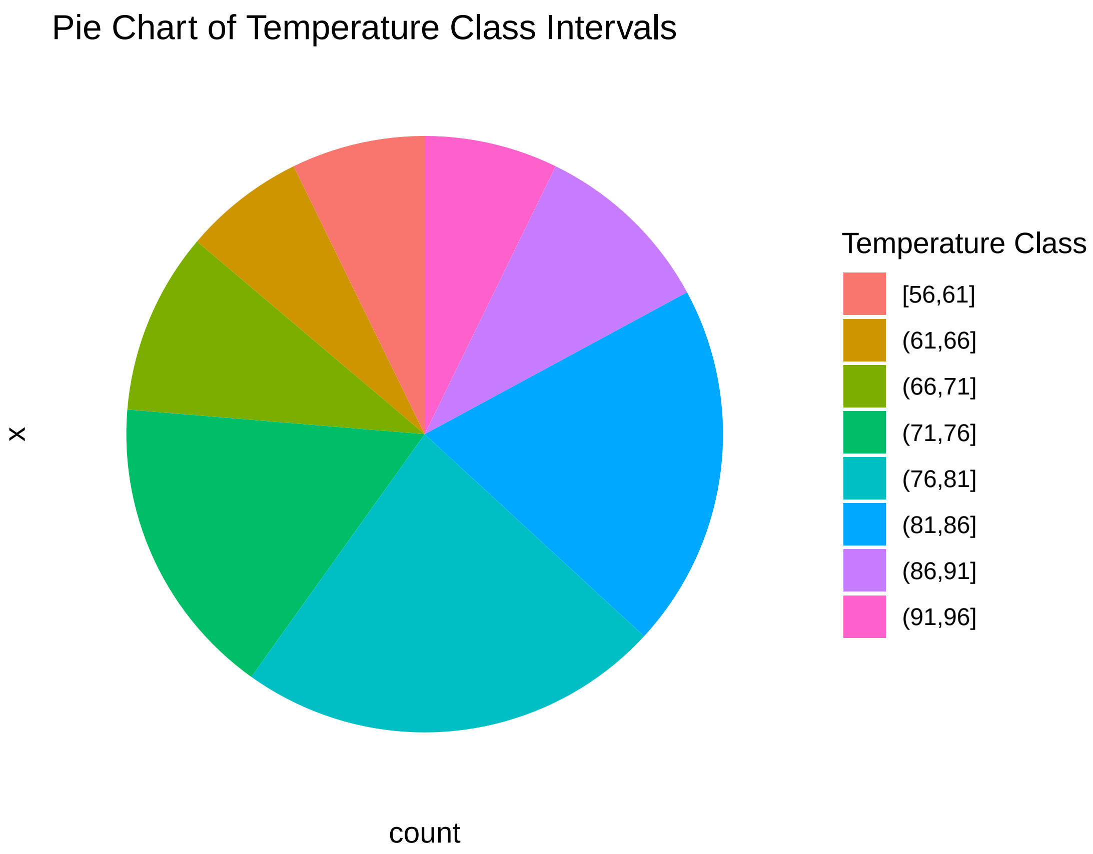
```r
# Visualize the graph with communities highlighted
plot(communities, g1)
#Communities in a graph represent groups of nodes that are more densely connected
#within the group compared to connections between groups

# Interpretation

# plotted a graph with all of the node color as green and node size as 30, edge color as red and link s
# got the degree which show the adjacent edges between nodes,
# got the closeness which shows how many steps are required to access each vertex from given, here I ha
# got the betweenness which shows the numbers paths going thorough one particular vertex
# Communities in a graph represent groups of nodes that are more densely connected
#within the group compared to connections between groups

# question number 7

library(ggplot2, warn.conflicts = F)
```



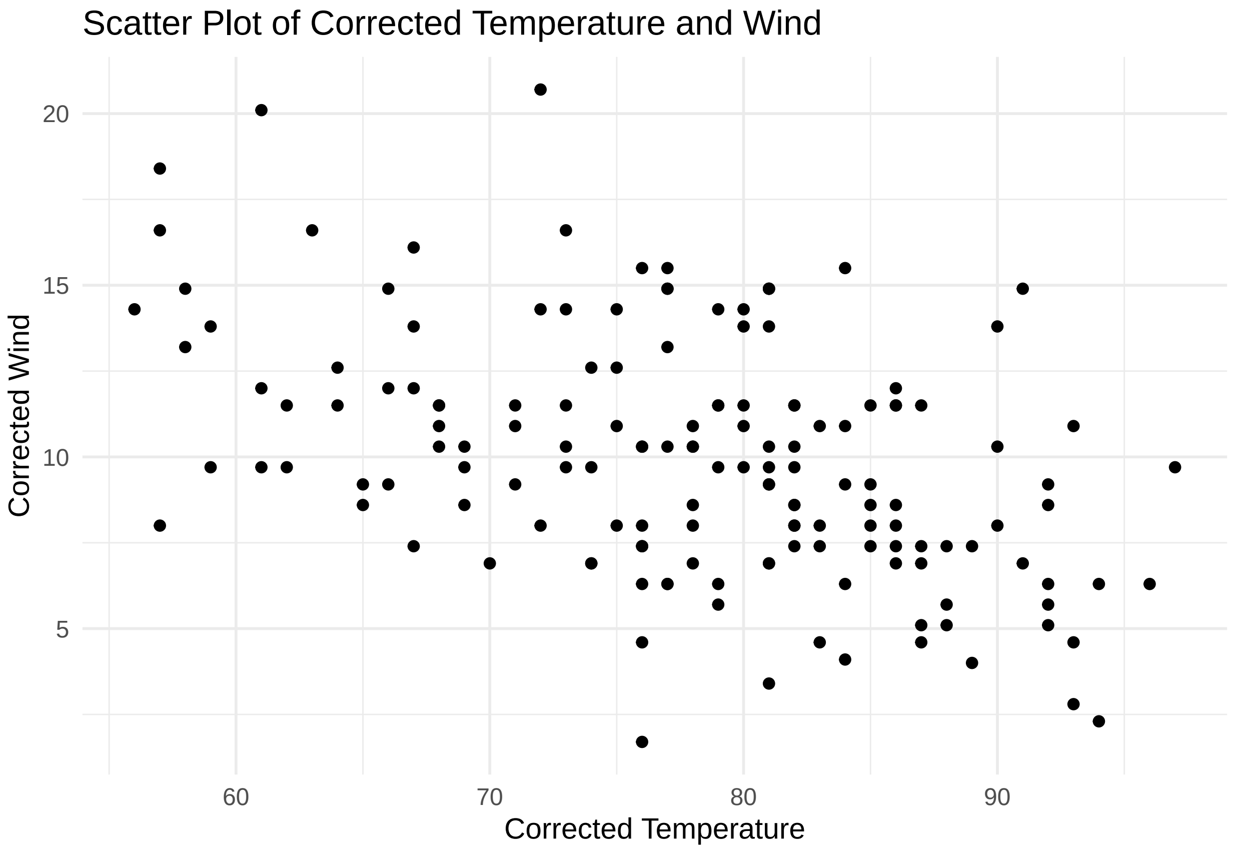
```
library(dplyr, warn.conflicts = F)
data <- airquality

# answer a
data <- data %>%
  mutate(Temp = ifelse(is.na(Temp), mean(Temp, na.rm = TRUE), Temp))

# b
class_intervals <- data %>%
  mutate(Temp_class = cut(Temp, breaks = seq(min(Temp), max(Temp), by = 5), include.lowest = TRUE)) %>%
  group_by(Temp_class) %>%
  summarize(count = n()) %>%
  na.omit()

# b
ggplot(class_intervals, aes(x = Temp_class, y = count)) +
  geom_bar(stat = "identity") +
  labs(title = "Frequency Distribution of Temperature Class Intervals",
       x = "Temperature Class",
       y = "Count") +
  theme_minimal()
```



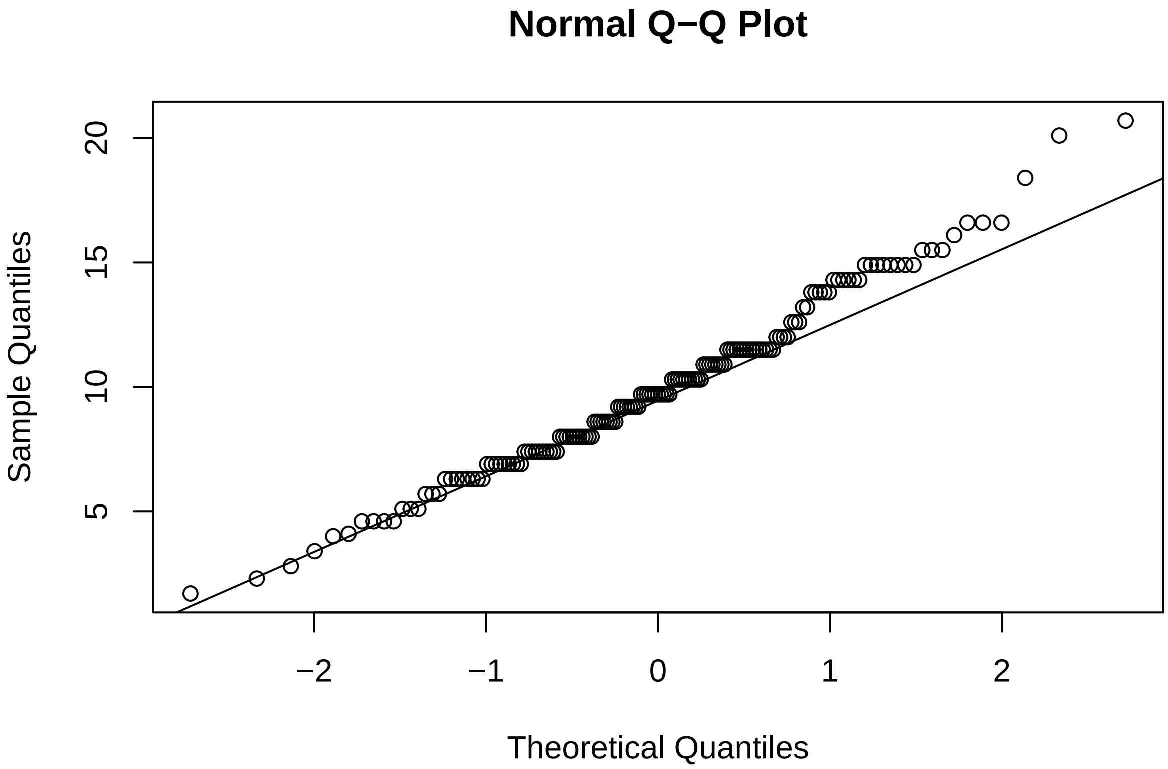
```
# c
ggplot(class_intervals, aes(x = "", y = count, fill = Temp_class)) +
  geom_bar(stat = "identity", width = 1) +
  coord_polar("y", start = 0) +
  labs(title = "Pie Chart of Temperature Class Intervals",
       fill = "Temperature Class") +
  theme_minimal() +
  theme(axis.text.x = element_blank(),
        axis.ticks.x = element_blank(),
        panel.grid = element_blank())
```



```r
# d
# Create a scatter plot of the corrected Temp and wind variables
ggplot(data, aes(x = Temp, y = Wind)) +
  geom_point() +
  labs(title = "Scatter Plot of Corrected Temperature and Wind",
       x = "Corrected Temperature",
       y = "Corrected Wind") +
  theme_minimal()
```



```
# Interpretation:
# a. there seems to be no outliers in the data.
# b. created the class interval of the given using cut and break with bin size 10,
# and showed plotted the frequency distribution. the distribution seems to be normal
# c. created the pie chart of class variable that was created in b.
# d. created the scatter plot of Temp and wind variable, it is seen that there tends to be somewhat
# negative correlation meaning increase in wind tends the lower the Temperature


# question number 8

data <- airquality

# a
# confirmative test
qqnorm(data$Wind)
qqline(data$Wind)
```



```
#  the seem to follow the normal distribution
# suggestive test
shapiro_test_result <- shapiro.test(data$Temp)
shapiro_test_result
```

```
##
##  Shapiro-Wilk normality test
##
## data:  data$Temp
## W = 0.97617, p-value = 0.009319
```

```
# if p > 0.05 it follows the normal distribution and since p < 0.05 it we can confirm that it doesnot f

# b

bartlett_test <- bartlett.test(Wind~Month, data = data)
# and since p value is greater than 0.05 we can say that the variances of wind are not significantly di

# c
library(car)
```

```
## Loading required package: carData
```

```
##
## Attaching package: 'car'
```

```
## The following object is masked from 'package:dplyr':
##
##     recode
```

```r
library(caret)
```

```
## Loading required package: lattice
```

```r
anova_model <- aov(Wind ~ as.factor(Month), data = data)
summary(anova_model)
```

```
##                   Df Sum Sq Mean Sq F value  Pr(>F)
## as.factor(Month)   4  164.3   41.07   3.529 0.00879 **
## Residuals        148 1722.3   11.64
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

```r
#### Since the p-value (0.00879) is less than the common significance level of 0.05, we reject the null
#### This suggests that there is a statistically significant difference in mean wind speed across diffe


# d

# Apply TukeyHSD to the ANOVA model

tukey_result <- TukeyHSD(anova_model,conf.level = 0.95)
print(tukey_result)
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = Wind ~ as.factor(Month), data = data)
##
## $‘as.factor(Month)‘
##            diff       lwr        upr     p adj
## 6-5 -1.35591398 -3.768713  1.0568846 0.5305524
## 7-5 -2.68064516 -5.073585 -0.2877054 0.0197174
## 8-5 -2.82903226 -5.221972 -0.4360925 0.0117066
## 9-5 -1.44258065 -3.855379  0.9702179 0.4674045
## 7-6 -1.32473118 -3.737530  1.0880674 0.5535894
## 8-6 -1.47311828 -3.885917  0.9396803 0.4456532
## 9-6 -0.08666667 -2.519162  2.3458285 0.9999786
## 8-7 -0.14838710 -2.541327  2.2445527 0.9998052
## 9-7  1.23806452 -1.174734  3.6508631 0.6176733
## 9-8  1.38645161 -1.026347  3.7992502 0.5081147
```

```r
#### Since, adjusted p value for june and may (6-5) is 0.53 which is greater than 0.05, that means the
#### there is no significant difference in windspeed betweeen these two months.
#### From the above table, we can say that there is a significant difference in windspeeds in
#### July-May (7-5) and August-May(8-5) but in all other months there is no such difference in windspee

#


# Question Number 9
# Load the data
df <- USArrests

index <- sample(2, size=nrow(df), replace = T, prob = c(0.7,0.3))

train_df <- df[index == 1, ]
test_df <- df[index ==2, ]

print(nrow(train_df))
```

```
## [1] 34
```

```r
print(nrow(test_df))
```

```
## [1] 16
```

```r
# Fit a supervised linear regression with training data
linear_reg <- lm(formula = UrbanPop ~ ., data = train_df)

# Checking vif scores to see if there is any multicollinearity present in the dataset
vif(linear_reg)
```

```
##   Murder  Assault     Rape 
## 3.459803 4.635274 2.139804
```

```r
# Since vif score less than 10, we can say that the features are not correlated with each other to a hi
print(summary(linear_reg))
```

```
## 
## Call:
## lm(formula = UrbanPop ~ ., data = train_df)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -34.412  -5.253   0.043   9.133  26.547 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 48.81814    5.80074   8.416 2.16e-09 ***
## Murder      -0.85548    0.90011  -0.950   0.3495    
## Assault      0.04508    0.05620   0.802   0.4288    
## Rape         0.68074    0.37130   1.833   0.0767 .  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 13.68 on 30 degrees of freedom
## Multiple R-squared:  0.233,  Adjusted R-squared:  0.1563 
## F-statistic: 3.038 on 3 and 30 DF,  p-value: 0.04424
```

```
# From this we can say that the variables am, wt and hp are statistically significant features.

normalize_func <- function(x) {
  scaled_values <- ((x-min(x))/ (max(x) - min(x)))
  return(scaled_values)
}

scaled_train_df <- train_df
scale_test_df <- test_df

scaled_train_df$Murder <- normalize_func(scaled_train_df$Murder)
scaled_train_df$Assault <- normalize_func(scaled_train_df$Assault)
scaled_train_df$Rape  <- normalize_func(scaled_train_df$Rape)

scale_test_df$Murder <- normalize_func(scale_test_df$Murder)
scale_test_df$Assault <- normalize_func(scale_test_df$Assault)
scale_test_df$Rape  <- normalize_func(scale_test_df$Rape)


# Fit a supervised linear regression with training data
scaled_linear_reg <- lm(formula = UrbanPop ~ ., data = scaled_train_df)

# Checking vif scores to see if there is any multicollinearity present in the dataset
vif(scaled_linear_reg)
```

```
##   Murder  Assault     Rape
## 3.459803 4.635274 2.139804
```

```
# Since vif score less than 10, we can say that the features are not correlated with each other to a hi
print(summary(scaled_linear_reg))
```

```
##
## Call:
## lm(formula = UrbanPop ~ ., data = scaled_train_df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -34.412  -5.253   0.043   9.133  26.547
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   55.132      4.457  12.371  2.6e-13 ***
## Murder       -14.201     14.942  -0.950   0.3495
## Assault       13.163     16.410   0.802   0.4288
## Rape          25.324     13.812   1.833   0.0767 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.68 on 30 degrees of freedom
## Multiple R-squared:  0.233,  Adjusted R-squared:  0.1563
## F-statistic: 3.038 on 3 and 30 DF,  p-value: 0.04424
```

```r
# Fit the model on the test dataset and get the indices and interpret them
predictions <- predict(scaled_linear_reg, scale_test_df)

library(caret)


indices_linear <- data.frame(
  R2 = R2(predictions, scale_test_df$UrbanPop),
  RMSE = RMSE(predictions, scale_test_df$UrbanPop),
  MAE = MAE(predictions, scale_test_df$UrbanPop)
)
indices_linear
```

```
##          R2     RMSE      MAE
## 1 0.2275445 11.72964 8.463308
```

```r
# the r2 value is less < 0.5, so the predictive power of the model is not good.

# KNN

knn_model <- knnreg(formula = UrbanPop ~ ., data = train_df)



# Make predictions on the testing set
predictions <- predict(knn_model, newdata = scale_test_df)
indices_knn <- data.frame(
  R2 = R2(predictions, scale_test_df$UrbanPop,na.rm = T),
  RMSE = RMSE(predictions, scale_test_df$UrbanPop, na.rm = T),
  MAE = MAE(predictions, scale_test_df$UrbanPop, na.rm = T)
)
```

```
## Warning in cor(obs, pred, use = ifelse(na.rm, "complete.obs", "everything")):
## the standard deviation is zero
```

```r
indices_knn
```

```
##   R2     RMSE     MAE
## 1 NA 22.32851 18.8125
```

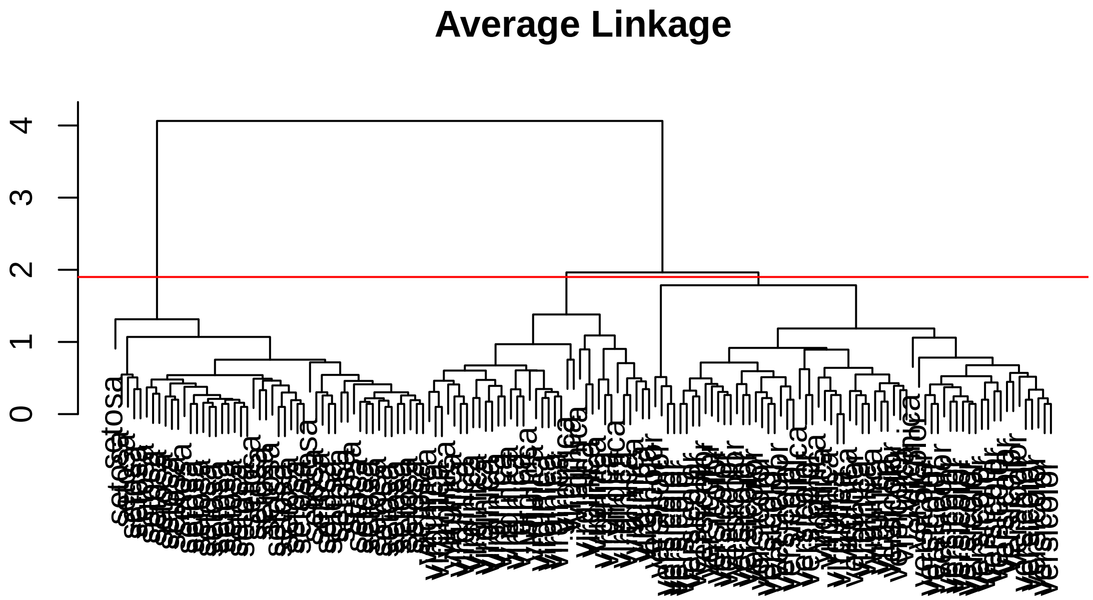
```r
# based on the results from the model, I can conclude that Liner Regression Performs Well.




# question number 10:

iris_data<-iris
head(iris_data)
```

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width Species
## 1          5.1         3.5          1.4         0.2  setosa
## 2          4.9         3.0          1.4         0.2  setosa
## 3          4.7         3.2          1.3         0.2  setosa
## 4          4.6         3.1          1.5         0.2  setosa
## 5          5.0         3.6          1.4         0.2  setosa
## 6          5.4         3.9          1.7         0.4  setosa
```

```r
ir_label <- iris_data$Species
ir_data <- iris_data[,-5]
head(ir_data)
```

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width
## 1          5.1         3.5          1.4         0.2
## 2          4.9         3.0          1.4         0.2
## 3          4.7         3.2          1.3         0.2
## 4          4.6         3.1          1.5         0.2
## 5          5.0         3.6          1.4         0.2
## 6          5.4         3.9          1.7         0.4
```

```r
data<-dist(ir_data)
data.dist <- dist(ir_data)
#a

plot(hclust(data.dist, method = "average"), xlab = "", sub = "", ylab = "",
     labels = ir_label, main = "Average Linkage")

# plotted the cluster using average linkage

#b
# Best hierarchical model

plot(hclust(data.dist, method = "average"), xlab = "", sub = "", ylab = "",
     labels = ir_label, main = "Average Linkage")

abline(h = 1.9, col = "red")
```



```
sd.data <- scale(iris-5)
```

```
## Warning in Ops.factor(left, right): '-' not meaningful for factors
```

```
hc.out <- hclust(dist(sd.data))
hc.clusters <- cutree(hc.out, 4)



# the best value for cluster is 3
hc.out <- hclust(dist(sd.data))
hc.clusters <- cutree(hc.out, 4)
#c
kmeans.c3<-kmeans(ir_data,centers = 3,nstart = 20)
kmeans.c3
```

```
## K-means clustering with 3 clusters of sizes 38, 62, 50
##
## Cluster means:
##   Sepal.Length Sepal.Width Petal.Length Petal.Width
## 1     6.850000    3.073684     5.742105    2.071053
## 2     5.901613    2.748387     4.393548    1.433871
## 3     5.006000    3.428000     1.462000    0.246000
##
## Clustering vector:
##   [1] 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
##  [38] 3 3 3 3 3 3 3 3 3 3 3 3 3 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
##  [75] 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 1 1 1 1 2 1 1 1 1
## [112] 1 1 2 2 1 1 1 1 2 1 2 1 2 1 1 2 2 1 1 1 1 1 2 1 1 1 1 2 1 1 1 2 1 1 1 2 1
## [149] 1 2
##
## Within cluster sum of squares by cluster:
## [1] 23.87947 39.82097 15.15100
##  (between_SS / total_SS =  88.4 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
## [6] "betweenss"    "size"         "iter"         "ifault"
```

```
# the output K-means clustering has 3 clusters with sizes 50, 62, 38
#d
cm<-table(iris$Species,kmeans.c3$cluster)
cm
```

```
##
##               1  2  3
##   setosa      0  0 50
##   versicolor  2 48  0
##   virginica  36 14  0
```

```
# from the result the true positive cases for setosa, versicolor and virginicia is 50, 48, 36

(accuracy<-
    sum(diag(cm))/sum(cm))
```

```
## [1] 0.32
```

```
#
# the accuracy of the fitted model is 089 percent
```