2nd assessment exam

roll 10

2024-05-31

QUESTION NO 7

```r
data <- airquality
str(data)
```

```
## 'data.frame':    153 obs. of  6 variables:
##  $ Ozone  : int  41 36 12 18 NA 28 23 19 8 NA ...
##  $ Solar.R: int  190 118 149 313 NA NA 299 99 19 194 ...
##  $ Wind   : num  7.4 8 12.6 11.5 14.3 14.9 8.6 13.8 20.1 8.6 ...
##  $ Temp   : int  67 72 74 62 56 66 65 59 61 69 ...
##  $ Month  : int  5 5 5 5 5 5 5 5 5 5 ...
##  $ Day    : int  1 2 3 4 5 6 7 8 9 10 ...
```

```r
# Convert Month to a factor variable
data$Month <- as.factor(data$Month)

# Calculate the mean and standard deviation of Temp by Month
Temp_mean <- tapply(data$Temp, data$Month, mean, na.rm = TRUE)
Temp_sd <- tapply(data$Temp, data$Month, sd, na.rm = TRUE)

# Create a data frame to display the results
monthly_Temp_data <- data.frame(
  Month = names(Temp_mean),
  Mean_Temp = Temp_mean,
  SD_Temp = Temp_sd
)
monthly_Temp_data
```

```
##   Month Mean_Temp  SD_Temp
## 5     5  65.54839 6.854870
## 6     6  79.10000 6.598589
## 7     7  83.90323 4.315513
## 8     8  83.96774 6.585256
## 9     9  76.90000 8.355671
```

```r
#a) Perform goodness-of-fit test on Temp variable by Month variable to check if
# it follows normal distribution or not
```

```r
# Perform Shapiro-Wilk test for normality within each month
result <- tapply(data$Temp, data$Month, shapiro.test)
print(result)
```

```
## $`5`
##
##  Shapiro-Wilk normality test
##
## data:  X[[i]]
## W = 0.94771, p-value = 0.1349
##
##
## $`6`
##
##  Shapiro-Wilk normality test
##
## data:  X[[i]]
## W = 0.97158, p-value = 0.5832
##
##
## $`7`
##
##  Shapiro-Wilk normality test
##
## data:  X[[i]]
## W = 0.94579, p-value = 0.1194
##
##
## $`8`
##
##  Shapiro-Wilk normality test
##
## data:  X[[i]]
## W = 0.96391, p-value = 0.3688
##
##
## $`9`
##
##  Shapiro-Wilk normality test
##
## data:  X[[i]]
## W = 0.9513, p-value = 0.1831
```

```r
## The data follows a normal distribution within each month as p value is greater than 0.05.

#b) Perform goodness-of-fit test on Temp variable by Month variable to check if
#the variances of mpg are equal or not on am variable categories

airquality$Month <- factor(airquality$Month)
bartlett_result <- bartlett.test(Temp ~ Month, data = airquality)
print(bartlett_result)
```

```
##
```

```
##  Bartlett test of homogeneity of variances
##
## data:  Temp by Month
## Bartlett's K-squared = 12.023, df = 4, p-value = 0.01718
```

```r
#c)     Discuss which independent sample test must be used to compare "Temp" variable by "Month"
#variable categories based on the results obtained above.

#Bartlett's test in the above case suggests that the "Temp" variable's variances
#are roughly equal between months. Consequently, the conventional
#one-way ANOVA is appropriate.

#d) perform the best independent sample statistical test for this data now and interpret the result car
data("airquality")
anova_model <- aov(Temp ~ Month, data = airquality)
summary(anova_model)
```

```
##              Df Sum Sq Mean Sq F value   Pr(>F)
## Month         1   2413  2413.0   32.52 6.03e-08 ***
## Residuals   151  11205    74.2
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
airquality$Month <- factor(airquality$Month)
anova_model <- aov(Temp ~ Month, data = airquality)
tukey_result <- TukeyHSD(anova_model)
print(tukey_result)
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = Temp ~ Month, data = airquality)
##
## $Month
##            diff          lwr       upr     p adj
## 6-5 13.55161290   8.84386422 18.259362 0.0000000
## 7-5 18.35483871  13.68583759 23.023840 0.0000000
## 8-5 18.41935484  13.75035372 23.088356 0.0000000
## 9-5 11.35161290   6.64386422 16.059362 0.0000000
## 7-6  4.80322581   0.09547713  9.510974 0.0430674
## 8-6  4.86774194   0.15999325  9.575491 0.0388654
## 9-6 -2.20000000  -6.94617992  2.546180 0.7038121
## 8-7  0.06451613  -4.60448499  4.733517 0.9999995
## 9-7 -7.00322581 -11.71097449 -2.295477 0.0006215
## 9-8 -7.06774194 -11.77549062 -2.359993 0.0005376
```

```r
# Here we can see relationship between temp and month of (6-5),(7-5),(8-5),(9-5) are less significant
#as compared to month of (9-6),(8-7).
```

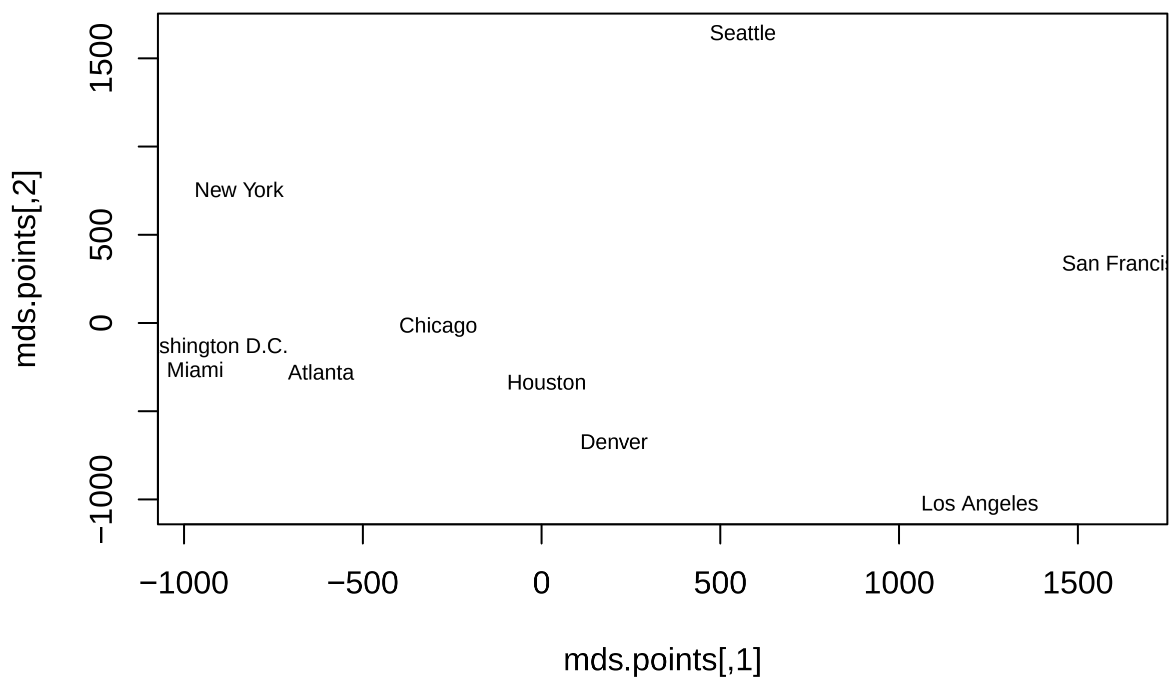
QUESTION NO 9

```r
library(stats)
city_distances <- matrix(c(
  0, 587, 1212, 701, 1936, 604, 748, 2139, 2182, 543,
  587, 0, 920, 940, 1745, 1188, 713, 1858, 1737, 597,
  1212, 920, 0, 879, 831, 1726, 1611, 1949, 2204, 1494,
  701, 940, 879, 0, 1374, 968, 1420, 1645, 1891, 1220,
  1936, 1745, 831, 1374, 0, 2339, 2451, 347, 2734, 2300,
  604, 1188, 1726, 968, 2339, 0, 1092, 2594, 2408, 923,
  748, 713, 1611, 1420, 2451, 1092, 0, 2571, 678, 205,
  2139, 1858, 1949, 1645, 347, 2594, 2571, 0, 678, 2442,
  2182, 1737, 2204, 1891, 2734, 2408, 678, 678, 0, 2329,
  543, 597, 1494, 1220, 2300, 923, 205, 2442, 2329, 0
), nrow = 10, byrow = TRUE)

# Assigning names to row and columns
city_names <- c("Atlanta", "Chicago", "Denver", "Houston", "Los Angeles", "Miami",
                "New York", "San Francisco", "Seattle", "Washington D.C.")
rownames(city_distances) <- city_names
colnames(city_distances) <- city_names

## A)
## Get dissimilarity distance as city.dissimilarity object
city.dissimilarity <- as.dist(city_distances)

## B)
## Fit the classical MDS model using city.dissimilarity object
mds.model <- cmdscale(city.dissimilarity, eig = TRUE, k = 2)  # Dimension 2


## C)
# Summary of model
mds.points <- mds.model$points
print(mds.points)
```

```
##                        [,1]        [,2]
## Atlanta         -616.46326  -277.03319
## Chicago         -288.61063   -22.16151
## Denver           202.61148  -672.61019
## Houston           14.25242  -335.54496
## Los Angeles     1225.78174 -1033.78934
## Miami           -968.45797  -264.31832
## New York        -845.50822   757.66327
## San Francisco   1645.58380   339.92746
## Seattle          563.12009  1646.43854
## Washington D.C. -932.30945  -138.57175
```

```r
## Interpretation
# These coordinates represent the cities' positions relative to each other based on their pairwise dist
```

```
## D)
## Bi-plot of the model
plot(mds.points, type = "n")
text(mds.points, labels = city_names, cex = 0.7)
```




```
## Interpretation
# - The bi-plot visually represents the cities' positions in the 2D space.
#- The distances between cities in this plot reflect their dissimilarities from the original distance m
#- Cities close together in the plot are more similar (based on the original distances), while those fa
#In summary, the code performs classical MDS on city distances, obtains 2D coordinates, and creates a b
```

#QUESTION NO. 8

```
library(car)
```

```
## Warning: package 'car' was built under R version 4.3.3

## Loading required package: carData

## Warning: package 'carData' was built under R version 4.3.3
```

```
## Loading required package: carData
# Create "crime" dataset
crime_data <- Arrests
head(crime_data)
```

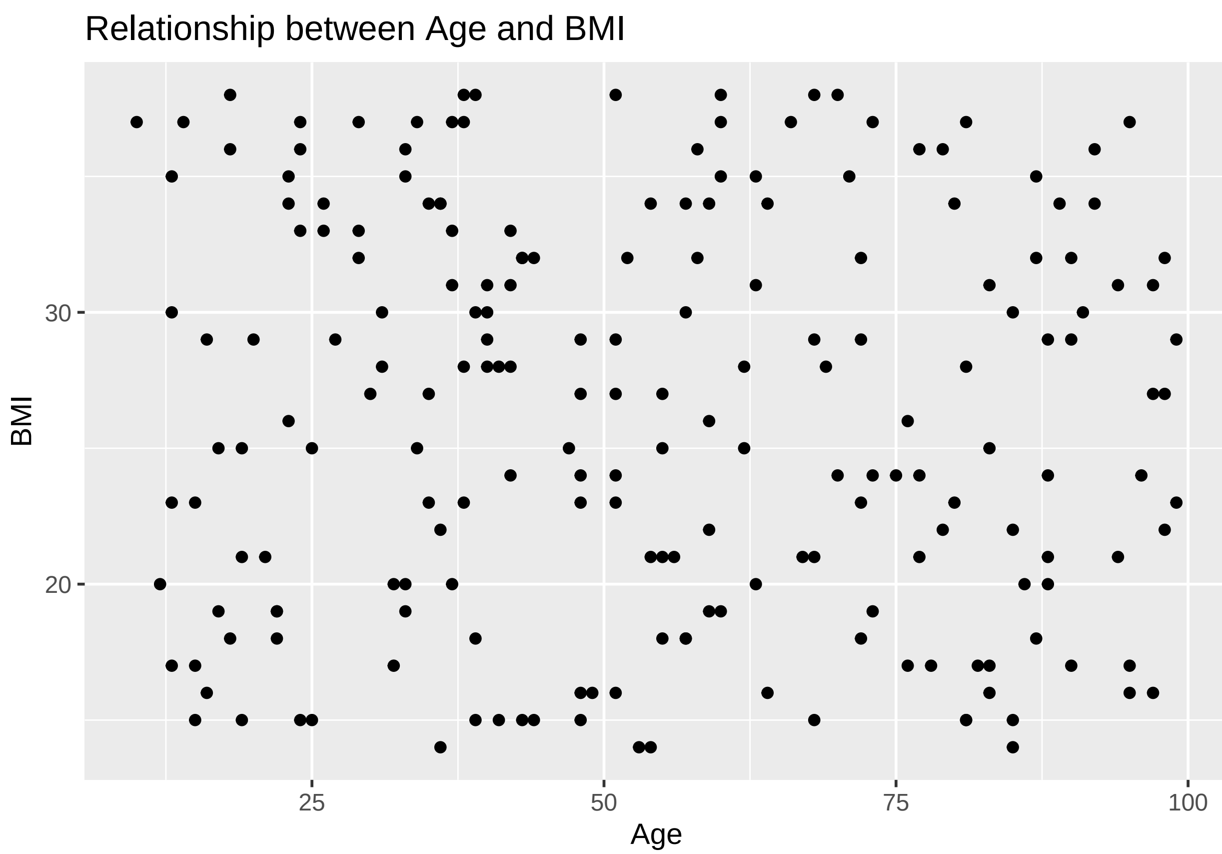
```
##   released colour year age    sex employed citizen checks
## 1      Yes  White 2002  21   Male      Yes     Yes      3
## 2       No  Black 1999  17   Male      Yes     Yes      3
## 3      Yes  White 2000  24   Male      Yes     Yes      3
## 4       No  Black 2000  46   Male      Yes     Yes      1
## 5      Yes  Black 1999  27 Female      Yes     Yes      1
## 6      Yes  Black 1998  16 Female      Yes     Yes      0
```

```
set.seed(10)
index <- sample(2, size = nrow(crime_data),replace = TRUE, prob = c(0.7, 0.3))
train_full <- crime_data[index ==1,]
test_full <- crime_data[index ==2,]
crime_data_full <- crime_data
```

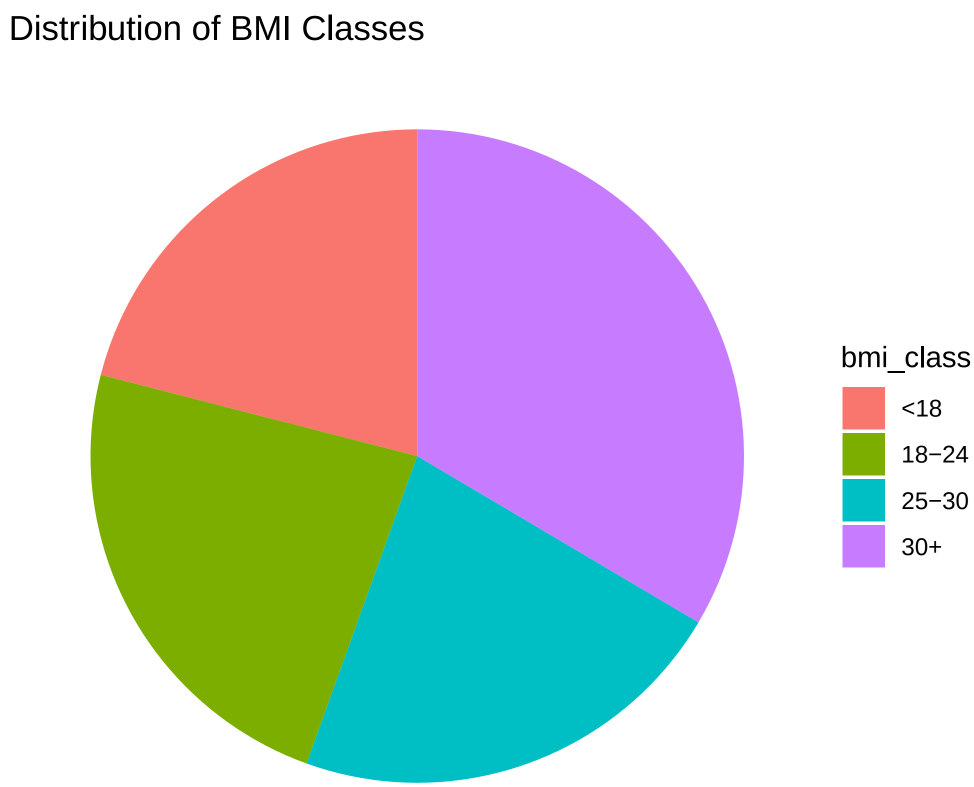
```
#QUESTION NO 6
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.3.3
```

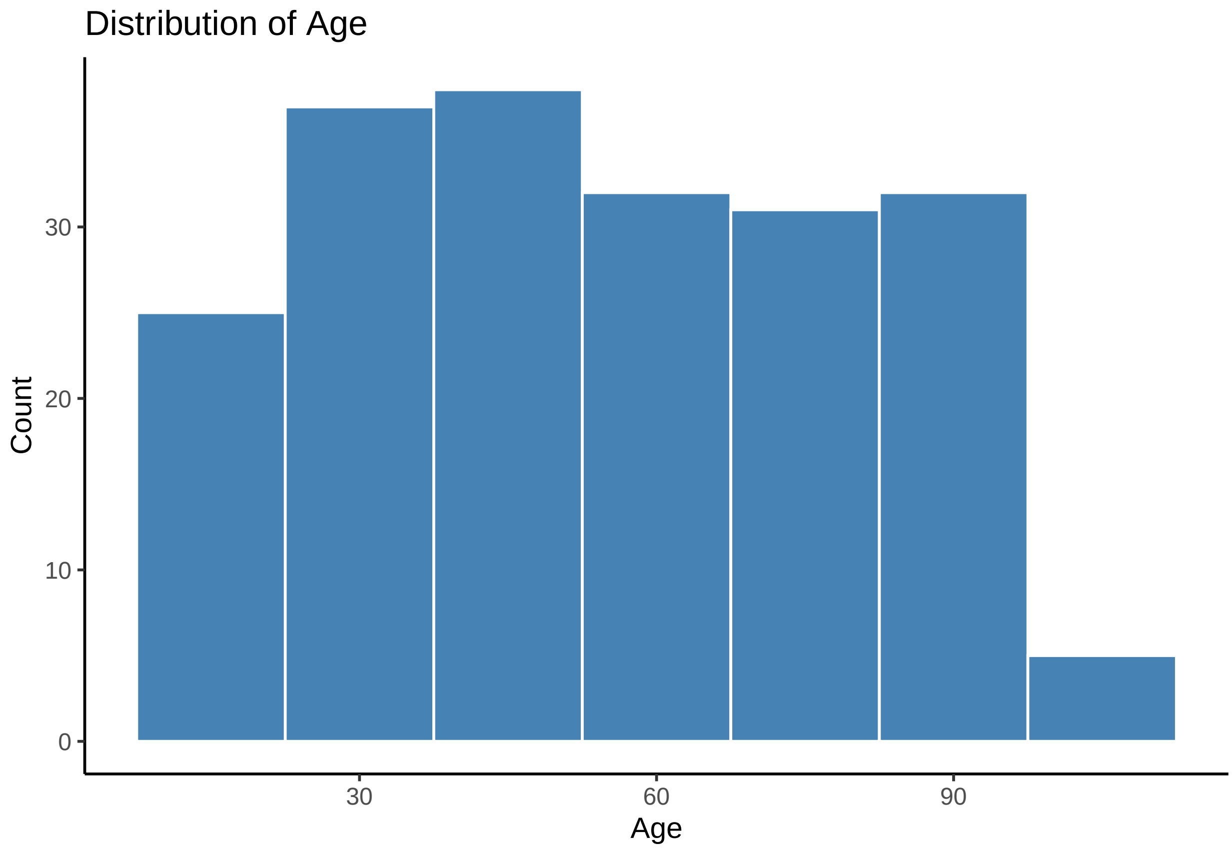
```
set.seed(10)

## A)
age <- sample(10:99, 200, replace = TRUE)
sex <- sample(c("Male", "Female"), 200, replace = TRUE)
education <- sample(c("No education", "Primary", "Secondary", "Beyond secondary"), 200, replace = TRUE)
socioeconomic_status <- sample(c("Low", "Middle", "High"), 200, replace = TRUE)
bmi <- sample(14:38, 200, replace = TRUE)

## B)
ggplot(data = data.frame(age, bmi), aes(x = age, y = bmi)) +
  geom_point() +
  labs(x = "Age", y = "BMI", title = "Relationship between Age and BMI")
```




```
##  the data is well spread as No trend is seen from the data.

## C)
bmi_class <- cut(bmi, breaks = c(0, 18, 24, 30, Inf), labels = c("<18", "18-24", "25-30", "30+"))

ggplot(data.frame(bmi_class), aes(x = "", fill = bmi_class)) +
  geom_bar(width = 1) +
  coord_polar("y", start = 0) +
  labs(title = "Distribution of BMI Classes") +
  theme_void() +
  theme(legend.position = "right")
```




```
##the maximum part of the data is covered by group 25-30 and 30+
#and minimum part of the data is from <18

## D)

ggplot(data.frame(age), aes(x = age)) +
  geom_histogram(binwidth = 15, fill = "steelblue", color = "white") +
  labs(x = "Age", y = "Count", title = "Distribution of Age") +
  theme_classic()
```




```
#From above plot we can see thet all the data has simmilar frequency except highest one
```