Second Assessment

Pratigya Jamakatel

2024-05-31

```r
#Q.no.6Ans:
# Load necessary libraries
library(ggplot2)
```

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

```r
# Set seed for reproducibility
set.seed(27)

# Generate random data
age <- sample(10:99, 200, replace = TRUE)
age
```

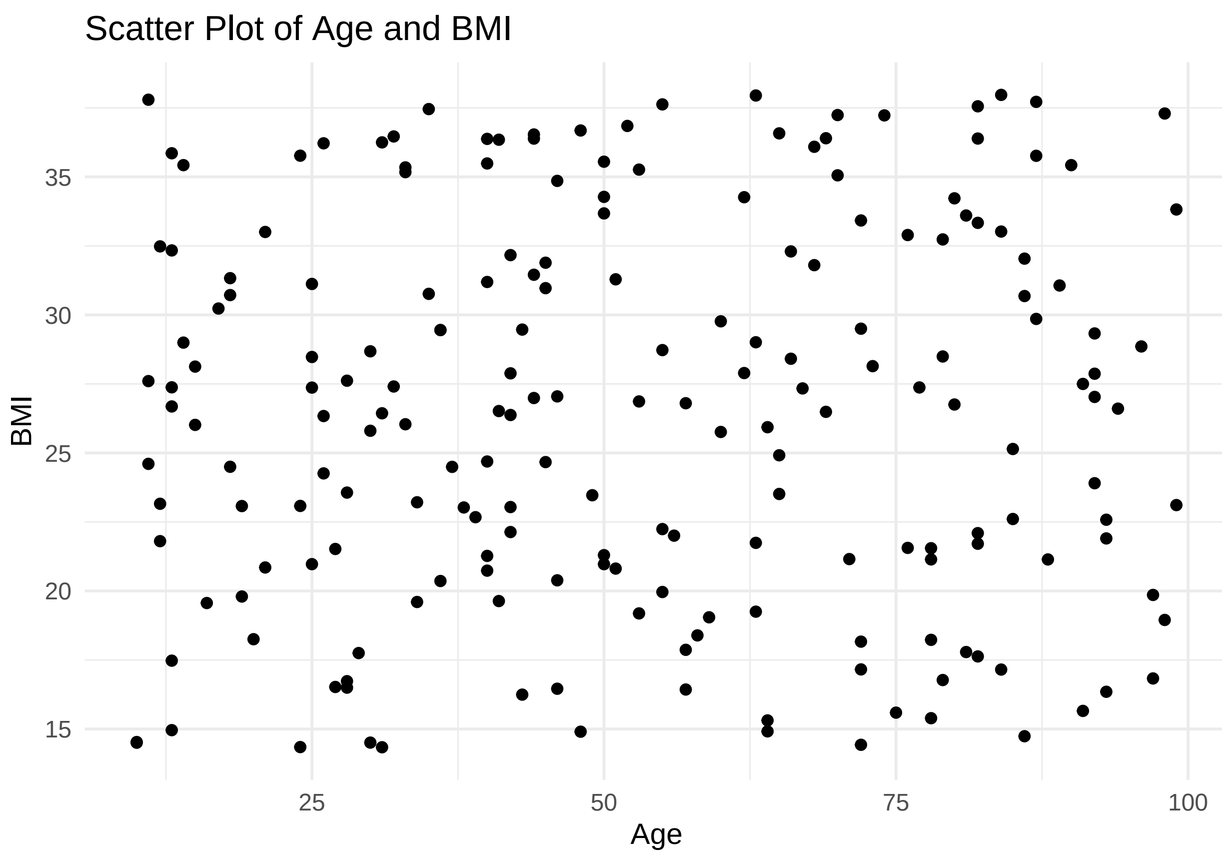
```
##   [1] 78 63 82 92 25 42 34 42 66 10 62 12 84 87 65 26 85 87 24 33 37 28 42 97 52
##  [26] 10 18 78 91 13 40 50 33 93 45 82 99 96 58 98 25 82 86 42 49 77 13 15 60 18
##  [51] 40 64 38 55 76 30 36 91 69 13 46 92 40 48 45 46 63 69 57 66 30 51 92 99 94
##  [76] 43 97 28 51 81 46 78 64 55 33 46 12 24 82 45 26 50 28 79 14 32 18 32 12 86
## [101] 31 53 90 72 84 35 82 28 40 14 50 36 11 93 80 81 72 88 74 43 50 44 86 29 87
## [126] 72 75 27 70 26 85 27 67 65 68 78 17 41 21 25 93 31 63 41 31 55 50 16 53 62
## [151] 73 55 44 65 64 72 36 79 39 35 13 13 40 48 44 13 11 79 34 89 68 57 20 63 19
## [176] 19 84 30 44 76 53 82 42 72 15 71 11 92 80 60 40 70 59 57 56 41 21 98 25 24
```

```r
sex <- sample(c("male", "female"), 200, replace = TRUE)
sex
```

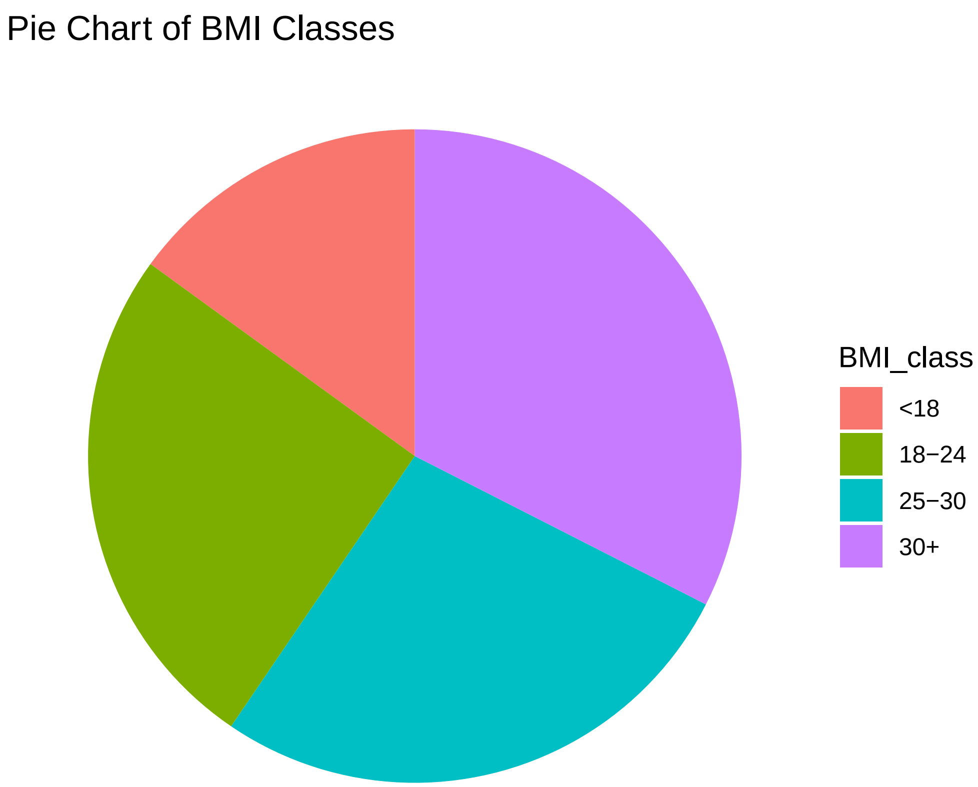
```
##   [1] "female" "female" "male"   "female" "male"   "female" "male"   "male"
##   [9] "male"   "female" "female" "female" "female" "female" "male"   "female"
##  [17] "female" "female" "female" "female" "male"   "female" "female" "female"
##  [25] "female" "female" "male"   "male"   "female" "female" "male"   "female"
##  [33] "female" "female" "male"   "male"   "female" "female" "female" "male"
##  [41] "male"   "female" "female" "male"   "male"   "male"   "male"   "female"
##  [49] "female" "female" "female" "male"   "male"   "female" "female" "male"
##  [57] "male"   "female" "female" "male"   "female" "male"   "male"   "male"
##  [65] "male"   "female" "male"   "male"   "male"   "female" "female" "female"
##  [73] "male"   "male"   "female" "female" "female" "female" "male"   "female"
##  [81] "female" "female" "female" "male"   "male"   "male"   "male"   "female"
##  [89] "female" "male"   "male"   "male"   "female" "male"   "female" "male"
##  [97] "male"   "female" "male"   "female" "female" "female" "female" "male"
## [105] "female" "female" "male"   "female" "female" "male"   "male"   "male"
## [113] "male"   "female" "female" "male"   "male"   "female" "male"   "female"
```

```
## [121] "male"   "male"   "male"   "female" "female" "male"   "male"   "female"
## [129] "female" "male"   "male"   "female" "male"   "female" "male"   "female"
## [137] "female" "male"   "female" "male"   "male"   "female" "male"   "female"
## [145] "female" "female" "female" "male"   "male"   "male"   "female" "female"
## [153] "male"   "male"   "male"   "male"   "female" "female" "female" "male"
## [161] "female" "female" "female" "male"   "male"   "female" "male"   "female"
## [169] "female" "female" "male"   "male"   "male"   "female" "female" "female"
## [177] "male"   "female" "female" "female" "male"   "female" "male"   "male"
## [185] "female" "male"   "female" "female" "male"   "female" "male"   "female"
## [193] "female" "male"   "female" "female" "female" "female" "female" "female"
```

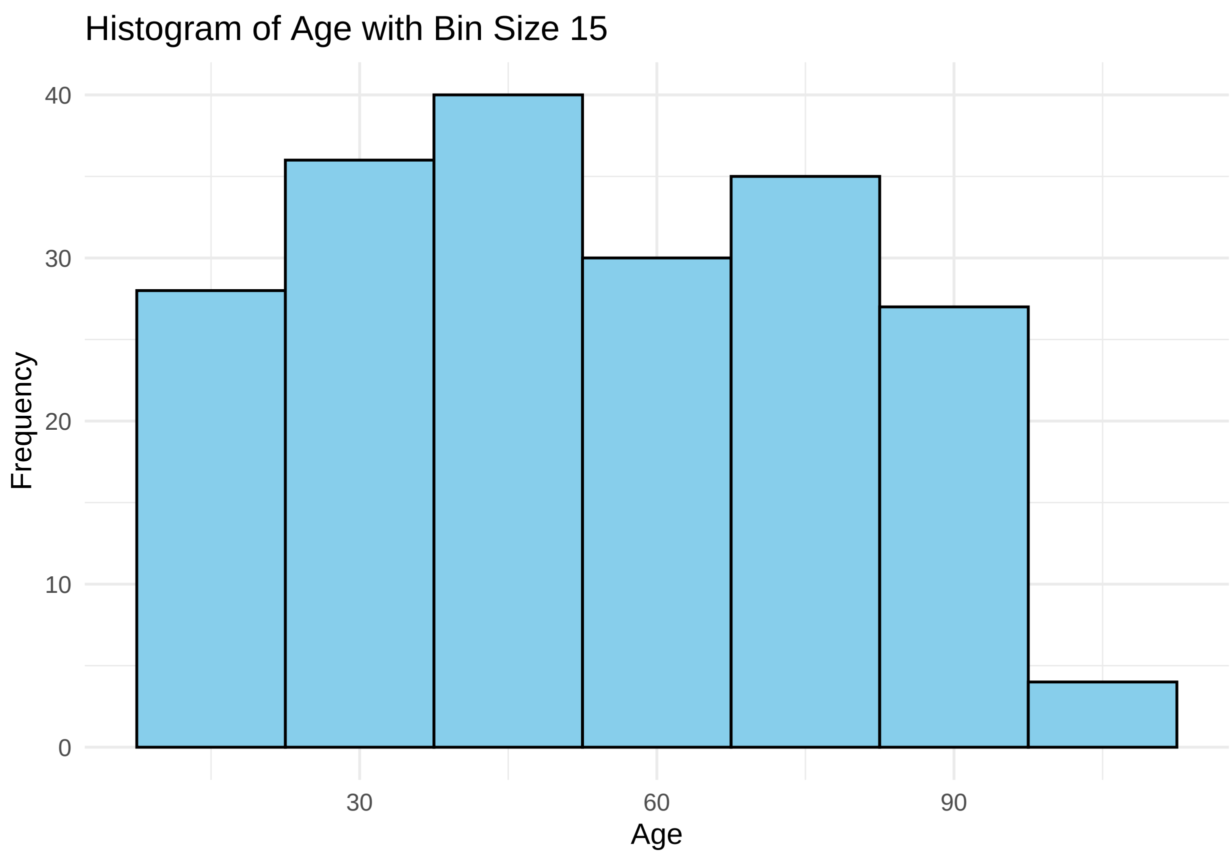
```r
education <- sample(c("No education", "Primary", "Secondary", "Beyond secondary"), 200, replace = TRUE)
socio_economic <- sample(c("Low", "Middle", "High"), 200, replace = TRUE)
bmi <- runif(200, min = 14, max = 38)

#a Create dataset.
data <- data.frame(age, sex, education, socio_economic, bmi)

# b) Create scatter plot of age and BMI
scatter_plot <- ggplot(data, aes(x = age, y = bmi)) +
  geom_point() +
  labs(title = "Scatter Plot of Age and BMI",
       x = "Age",
       y = "BMI") +
  theme_minimal()

print(scatter_plot)
```




```r
# c) Create class of BMI variable and pie chart
# Create BMI class
data$BMI_class <- cut(data$bmi, breaks = c(0, 18, 24, 30, Inf),
                      labels = c("<18", "18-24", "25-30", "30+"),
                      include.lowest = TRUE)

# Create pie chart
pie_chart <- ggplot(data, aes(x = "", fill = BMI_class)) +
  geom_bar(width = 1) +
  coord_polar("y", start = 0) +
  labs(title = "Pie Chart of BMI Classes") +
  theme_void() +
  theme(legend.position = "right")

print(pie_chart)
```




```r
# d) Create histogram of age variable
histogram <- ggplot(data, aes(x = age)) +
  geom_histogram(binwidth = 15, fill = "skyblue", color = "black") +
  labs(title = "Histogram of Age with Bin Size 15",
       x = "Age",
       y = "Frequency") +
  theme_minimal()

print(histogram)
```




```
#Interpretation:
#Scatter Plot (age vs BMI): This plot shows the relationship between age and BMI. By observing the scat

#Pie Chart (BMI Classes): This chart represents the distribution of BMI classes within the dataset. It

#Histogram (Age): The histogram illustrates the distribution of ages within the dataset with a bin size


#Q.no.7 Ans: using airquality dataset of R
#a)perform goodness-of-fit test on Temp variable to check if it follows normal distribution or not.
#b)perform goodness-of-fit test on temp variable by month variable to check if the variances of mpg are
#c)Discuss which independent sample test must be used to compare"Temp"variable by "Month" variable cate
#d)perform the best independent sample statistical test for this data and now interpret result carefull


# Load the airquality dataset
data(airquality)

# a) Perform goodness-of-fit test on Temp variable to check if it follows normal distribution or not.
shapiro.test(airquality$Temp)
```

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

```r
# b) Perform goodness-of-fit test on Temp variable by Month variable to check if the variances of Temp
bartlett.test(Temp ~ Month, data = airquality)
```

```
##
##  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
#the Bartlett test indicates whether the variances across different groups are equal or not, it helps d
#If the variances are equal, a parametric test like ANOVA can be used. If not, a non-parametric test li

# d) Perform the best independent sample statistical test for this data and now interpret the result ca
# Since Bartlett test indicates unequal variances, use the Kruskal-Wallis test.
kruskal.test(Temp ~ Month, data = airquality)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  Temp by Month
## Kruskal-Wallis chi-squared = 73.328, df = 4, p-value = 4.496e-15
```

```r
#Q.no.10 Ans:
# Load the iris dataset
data("iris")

# Take the first four variables (features) of the iris dataset
iris_features <- iris[, 1:4]
iris_features
```

```
##     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
## 7            4.6         3.4          1.4         0.3
## 8            5.0         3.4          1.5         0.2
## 9            4.4         2.9          1.4         0.2
## 10           4.9         3.1          1.5         0.1
## 11           5.4         3.7          1.5         0.2
## 12           4.8         3.4          1.6         0.2
## 13           4.8         3.0          1.4         0.1
## 14           4.3         3.0          1.1         0.1
## 15           5.8         4.0          1.2         0.2
## 16           5.7         4.4          1.5         0.4
## 17           5.4         3.9          1.3         0.4
## 18           5.1         3.5          1.4         0.3
## 19           5.7         3.8          1.7         0.3
## 20           5.1         3.8          1.5         0.3
```

```
## 21          5.4         3.4          1.7         0.2
## 22          5.1         3.7          1.5         0.4
## 23          4.6         3.6          1.0         0.2
## 24          5.1         3.3          1.7         0.5
## 25          4.8         3.4          1.9         0.2
## 26          5.0         3.0          1.6         0.2
## 27          5.0         3.4          1.6         0.4
## 28          5.2         3.5          1.5         0.2
## 29          5.2         3.4          1.4         0.2
## 30          4.7         3.2          1.6         0.2
## 31          4.8         3.1          1.6         0.2
## 32          5.4         3.4          1.5         0.4
## 33          5.2         4.1          1.5         0.1
## 34          5.5         4.2          1.4         0.2
## 35          4.9         3.1          1.5         0.2
## 36          5.0         3.2          1.2         0.2
## 37          5.5         3.5          1.3         0.2
## 38          4.9         3.6          1.4         0.1
## 39          4.4         3.0          1.3         0.2
## 40          5.1         3.4          1.5         0.2
## 41          5.0         3.5          1.3         0.3
## 42          4.5         2.3          1.3         0.3
## 43          4.4         3.2          1.3         0.2
## 44          5.0         3.5          1.6         0.6
## 45          5.1         3.8          1.9         0.4
## 46          4.8         3.0          1.4         0.3
## 47          5.1         3.8          1.6         0.2
## 48          4.6         3.2          1.4         0.2
## 49          5.3         3.7          1.5         0.2
## 50          5.0         3.3          1.4         0.2
## 51          7.0         3.2          4.7         1.4
## 52          6.4         3.2          4.5         1.5
## 53          6.9         3.1          4.9         1.5
## 54          5.5         2.3          4.0         1.3
## 55          6.5         2.8          4.6         1.5
## 56          5.7         2.8          4.5         1.3
## 57          6.3         3.3          4.7         1.6
## 58          4.9         2.4          3.3         1.0
## 59          6.6         2.9          4.6         1.3
## 60          5.2         2.7          3.9         1.4
## 61          5.0         2.0          3.5         1.0
## 62          5.9         3.0          4.2         1.5
## 63          6.0         2.2          4.0         1.0
## 64          6.1         2.9          4.7         1.4
## 65          5.6         2.9          3.6         1.3
## 66          6.7         3.1          4.4         1.4
## 67          5.6         3.0          4.5         1.5
## 68          5.8         2.7          4.1         1.0
## 69          6.2         2.2          4.5         1.5
## 70          5.6         2.5          3.9         1.1
## 71          5.9         3.2          4.8         1.8
## 72          6.1         2.8          4.0         1.3
## 73          6.3         2.5          4.9         1.5
## 74          6.1         2.8          4.7         1.2
```

```
## 75           6.4         2.9          4.3         1.3
## 76           6.6         3.0          4.4         1.4
## 77           6.8         2.8          4.8         1.4
## 78           6.7         3.0          5.0         1.7
## 79           6.0         2.9          4.5         1.5
## 80           5.7         2.6          3.5         1.0
## 81           5.5         2.4          3.8         1.1
## 82           5.5         2.4          3.7         1.0
## 83           5.8         2.7          3.9         1.2
## 84           6.0         2.7          5.1         1.6
## 85           5.4         3.0          4.5         1.5
## 86           6.0         3.4          4.5         1.6
## 87           6.7         3.1          4.7         1.5
## 88           6.3         2.3          4.4         1.3
## 89           5.6         3.0          4.1         1.3
## 90           5.5         2.5          4.0         1.3
## 91           5.5         2.6          4.4         1.2
## 92           6.1         3.0          4.6         1.4
## 93           5.8         2.6          4.0         1.2
## 94           5.0         2.3          3.3         1.0
## 95           5.6         2.7          4.2         1.3
## 96           5.7         3.0          4.2         1.2
## 97           5.7         2.9          4.2         1.3
## 98           6.2         2.9          4.3         1.3
## 99           5.1         2.5          3.0         1.1
## 100          5.7         2.8          4.1         1.3
## 101          6.3         3.3          6.0         2.5
## 102          5.8         2.7          5.1         1.9
## 103          7.1         3.0          5.9         2.1
## 104          6.3         2.9          5.6         1.8
## 105          6.5         3.0          5.8         2.2
## 106          7.6         3.0          6.6         2.1
## 107          4.9         2.5          4.5         1.7
## 108          7.3         2.9          6.3         1.8
## 109          6.7         2.5          5.8         1.8
## 110          7.2         3.6          6.1         2.5
## 111          6.5         3.2          5.1         2.0
## 112          6.4         2.7          5.3         1.9
## 113          6.8         3.0          5.5         2.1
## 114          5.7         2.5          5.0         2.0
## 115          5.8         2.8          5.1         2.4
## 116          6.4         3.2          5.3         2.3
## 117          6.5         3.0          5.5         1.8
## 118          7.7         3.8          6.7         2.2
## 119          7.7         2.6          6.9         2.3
## 120          6.0         2.2          5.0         1.5
## 121          6.9         3.2          5.7         2.3
## 122          5.6         2.8          4.9         2.0
## 123          7.7         2.8          6.7         2.0
## 124          6.3         2.7          4.9         1.8
## 125          6.7         3.3          5.7         2.1
## 126          7.2         3.2          6.0         1.8
## 127          6.2         2.8          4.8         1.8
## 128          6.1         3.0          4.9         1.8
```

```
## 129          6.4         2.8          5.6         2.1
## 130          7.2         3.0          5.8         1.6
## 131          7.4         2.8          6.1         1.9
## 132          7.9         3.8          6.4         2.0
## 133          6.4         2.8          5.6         2.2
## 134          6.3         2.8          5.1         1.5
## 135          6.1         2.6          5.6         1.4
## 136          7.7         3.0          6.1         2.3
## 137          6.3         3.4          5.6         2.4
## 138          6.4         3.1          5.5         1.8
## 139          6.0         3.0          4.8         1.8
## 140          6.9         3.1          5.4         2.1
## 141          6.7         3.1          5.6         2.4
## 142          6.9         3.1          5.1         2.3
## 143          5.8         2.7          5.1         1.9
## 144          6.8         3.2          5.9         2.3
## 145          6.7         3.3          5.7         2.5
## 146          6.7         3.0          5.2         2.3
## 147          6.3         2.5          5.0         1.9
## 148          6.5         3.0          5.2         2.0
## 149          6.2         3.4          5.4         2.3
## 150          5.9         3.0          5.1         1.8
```

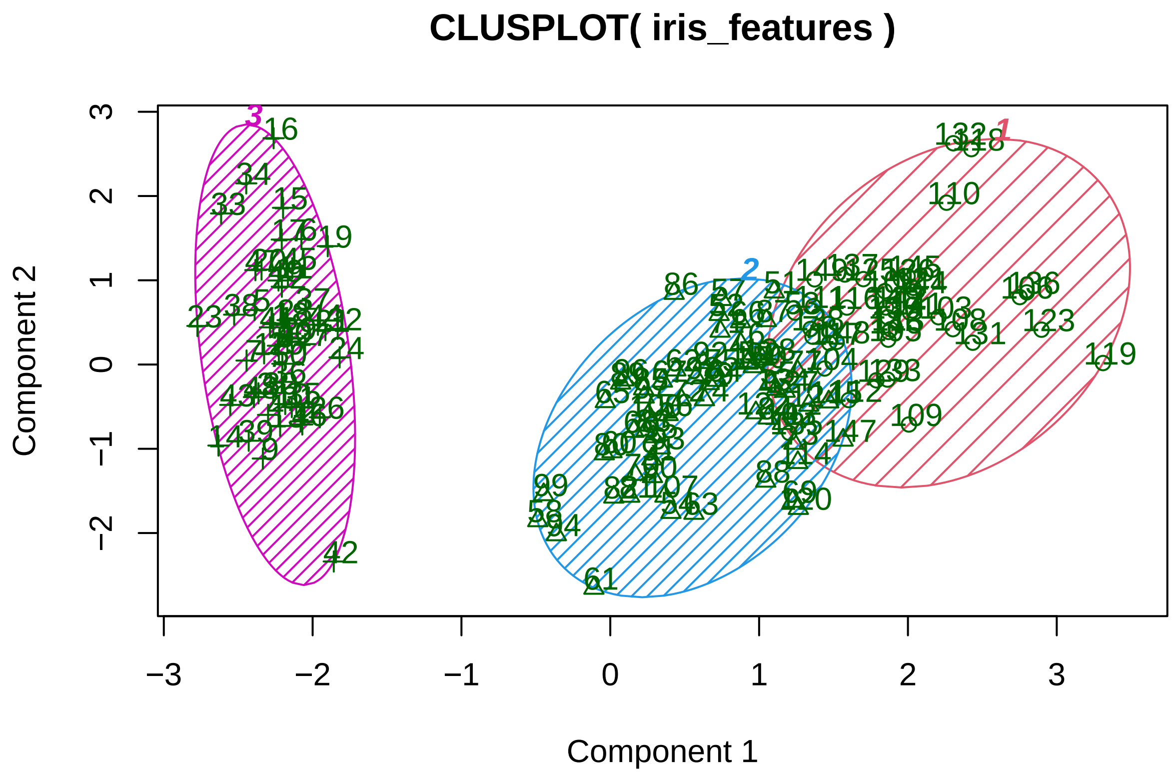
```r
# a) Fit a k-means clustering model in the data with k=2 and k=3
set.seed(27) # for reproducibility
kmeans_model_k2 <- kmeans(iris_features, centers = 2)
kmeans_model_k2
```

```
## K-means clustering with 2 clusters of sizes 97, 53
##
## Cluster means:
##   Sepal.Length Sepal.Width Petal.Length Petal.Width
## 1     6.301031    2.886598     4.958763    1.695876
## 2     5.005660    3.369811     1.560377    0.290566
##
## Clustering vector:
##   [1] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
##  [38] 2 2 2 2 2 2 2 2 2 2 2 2 2 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
##  [75] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1
## [112] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
## [149] 1 1
##
## Within cluster sum of squares by cluster:
## [1] 123.79588  28.55208
##  (between_SS / total_SS =  77.6 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
## [6] "betweenss"    "size"         "iter"         "ifault"
```

```r
kmeans_model_k3 <- kmeans(iris_features, centers = 3)
kmeans_model_k3
```

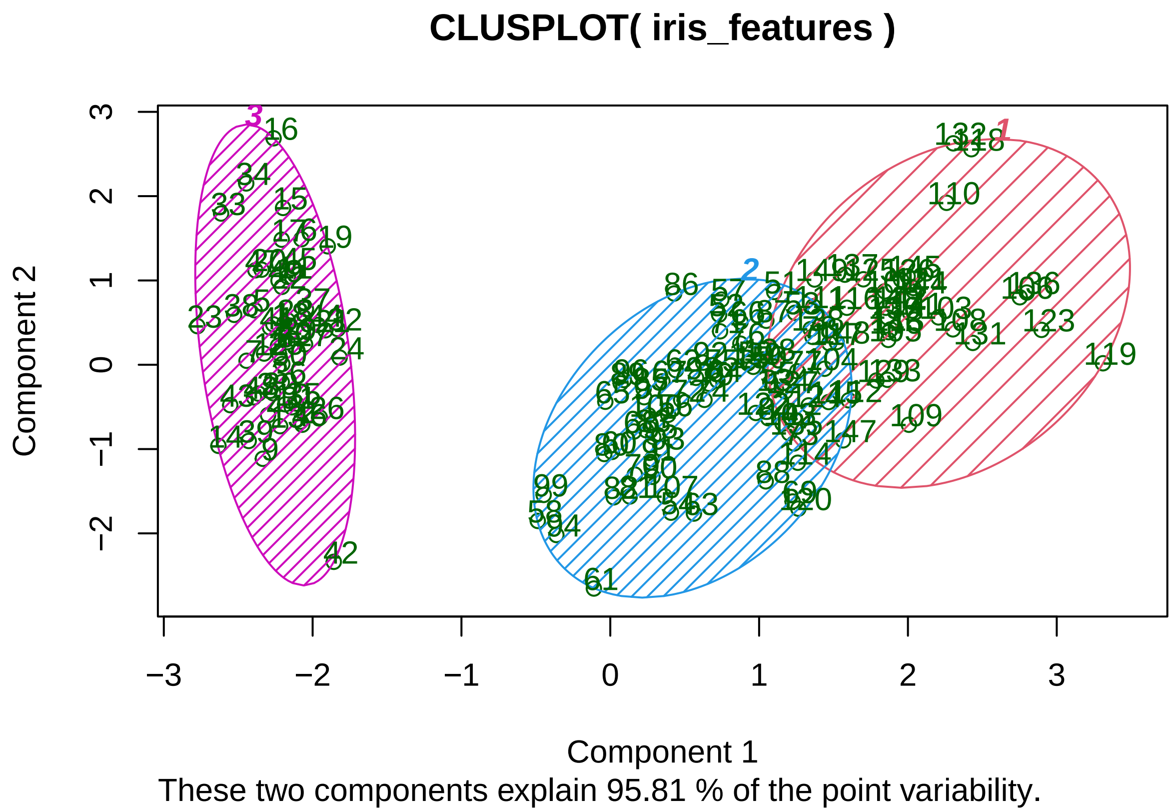
```
## K-means clustering with 3 clusters of sizes 38, 62, 50
##
## Cluster means:
##   Sepal.Length Sepal.Width Petal.Length Petal.Width
## 1     6.850000    3.073684     5.742105    2.071053
## 2     5.901613    2.748387     4.393548    1.433871
## 3     5.006000    3.428000     1.462000    0.246000
##
## 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"
```

```r
# b) Plot the clusters formed with k=3 in a single graph and interpret them carefully
library(cluster)
clusplot(iris_features, kmeans_model_k3$cluster, color = TRUE, shade = TRUE, labels = 2, lines = 0)
```



These two components explain 95.81 % of the point variability.

```
# Interpretation:
# The clusterplot visualizes the clusters formed by k-means with k=3.
# Each point represents an observation (flower) colored according to its assigned cluster.
# The plot provides insights into the separation of clusters based on the first two principal component

# c) Add cluster centers for the plot of cluster formed with k=3 above and interpret it carefully
clusplot(iris_features, kmeans_model_k3$cluster, color = TRUE, shade = TRUE, labels = 2, lines = 0, plo
points(kmeans_model_k3$centers, col = 1:3, pch = 8, cex = 2)
```




```
# Interpretation:
# In addition to the previous plot, this plot adds cluster centers represented by large triangles.
# Each triangle represents the centroid of a cluster.
# The plot allows for a clearer understanding of the location of the cluster centers relative to the da

# d) Compare the k=3 cluster variable with species variable of iris data using confusion matrix and int
table(iris$Species, kmeans_model_k3$cluster)
```

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

```
# Interpretation:
# The confusion matrix compares the species variable of the original iris dataset with the clusters for
# Each row represents the true species, while each column represents the assigned cluster.
# The numbers in the cells represent the counts of observations falling into each category.
# By comparing the clusters with the true species, we can assess how well the clustering algorithm perf


#Q.no.7 Ans: using airquality dataset of R

# Load the airquality dataset
data(airquality)
```

```r
# a) Perform goodness-of-fit test on Temp variable to check if it follows normal distribution or not.
shapiro.test(airquality$Temp)
```

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

```r
# b) Perform goodness-of-fit test on Temp variable by Month variable to check if the variances of Temp
bartlett.test(Temp ~ Month, data = airquality)
```

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

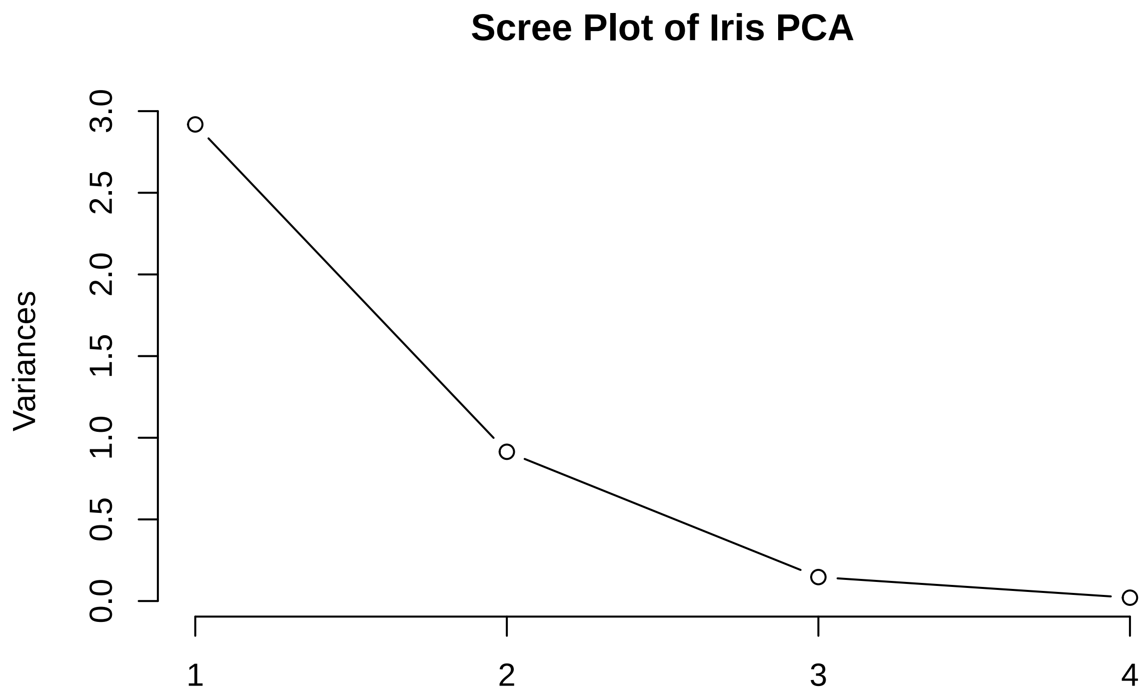
```r
# c) Discuss which independent sample test must be used to compare "Temp" variable by "Month" variable
#the Bartlett test indicates whether the variances across different groups are equal or not, it helps d
#If the variances are equal, a parametric test like ANOVA can be used. If not, a non-parametric test li

# d) Perform the best independent sample statistical test for this data and now interpret the result ca
# Since Bartlett test indicates unequal variances, use the Kruskal-Wallis test.
kruskal.test(Temp ~ Month, data = airquality)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  Temp by Month
## Kruskal-Wallis chi-squared = 73.328, df = 4, p-value = 4.496e-15
```

```r
#Q.no.9 Ans:using iris dataset

# Load the iris dataset
data(iris)

# a) Create a "flower scale" of the first four variables of iris dataset using PCA.
iris_pca <- prcomp(iris[,1:4], scale. = TRUE)

# b) Compute the eigenvalues and interpret the PCA result carefully using Kaiser's criteria.
eigenvalues <- iris_pca$sdev^2
kaisers_criteria <- sum(eigenvalues >= 1)
print(kaisers_criteria)
```

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

```r
# c) Show the scree plot and decide on the number of components to retain with careful interpretation.
screeplot(iris_pca, type = "line", main = "Scree Plot of Iris PCA")
```




```r
# d) Revise the flower scale with 3 components using VARIMAX rotation and interpret the result carefully
library(psych)
```

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

```
##
## Attaching package: 'psych'
```

```
## The following objects are masked from 'package:ggplot2':
##
##     %+%, alpha
```

```r
iris_pca_varimax <- principal(iris[,1:4], nfactors = 3, rotate = "varimax")
print(iris_pca_varimax)
```

```
## Principal Components Analysis
## Call: principal(r = iris[, 1:4], nfactors = 3, rotate = "varimax")
## Standardized loadings (pattern matrix) based upon correlation matrix
##                RC1   RC3   RC2   h2      u2 com
## Sepal.Length  0.55  0.84  0.01 1.00 0.00141 1.7
## Sepal.Width  -0.18 -0.03  0.98 1.00 0.00032 1.1
## Petal.Length  0.79  0.53 -0.28 0.99 0.01331 2.0
## Petal.Width   0.90  0.39 -0.20 0.99 0.00568 1.5
##
```

```
##                        RC1  RC3  RC2
## SS loadings           1.76 1.14 1.08
## Proportion Var        0.44 0.28 0.27
## Cumulative Var        0.44 0.72 0.99
## Proportion Explained  0.44 0.29 0.27
## Cumulative Proportion 0.44 0.73 1.00
##
## Mean item complexity =  1.6
## Test of the hypothesis that 3 components are sufficient.
##
## The root mean square of the residuals (RMSR) is  0
##  with the empirical chi square  0.03  with prob <  NA
##
## Fit based upon off diagonal values = 1
```

```
#Interpretation:
# a)PCA was performed on the first four variables of the iris dataset.
#b) The eigenvalues represent the amount of variance explained by each principal component. Kaiser's cr

#c) The scree plot displays the eigenvalues for each principal component. By observing the scree plot, 

#d) VARIMAX rotation is applied to the PCA to improve interpret ability by maximizing the variance of t


#Q.no.8 Ans: Using "Arrests" dataset of car package.
#a)Divide the Arrests data into the train and test the datasets with 80:20 random splits with 27.
```