Statistical Learning II

EAS 509 – Project I

Exasens Dataset

Reference : https://archive.ics.uci.edu/dataset/523/exasens

Team Members:

- **Sai Charitha Konda -** 50539675
- **Deepak Nallapaneni -** 50524851
- **Aishwarya nayak -** 50453772
- **Gowtham kalluri -** 50537244

Introduction:

This report outlines the procedures and outcomes of a data-driven project aimed at analyzing a comprehensive medical dataset, with the primary goal of enhancing diagnostic accuracy for respiratory diseases, including Chronic Obstructive Pulmonary Disease (COPD). The project utilizes a rich dataset containing a range of variables such as patient demographics, smoking habits, and complex medical measurements. The ultimate objective of this project is to leverage the insights derived from the data analysis to support medical professionals in making more accurate and timely diagnoses, thereby improving patient outcomes in respiratory healthcare. This report details each step of the project from data preparation through to model evaluation.

Dataset:

Exasens, a novel dataset for the classification of Chronic Obstructive Pulmonary Disease (COPD) patients and Healthy Controls. This dataset includes demographic information on 4 groups of saliva samples (COPD-HC-Asthma-Infected). It contains information on hundred saliva samples collected from four groups of respiratory patients including: COPD (40 samples), HC (40 samples), asthma (10 samples), and respiratory infected subjects without COPD or asthma (10 samples). The dataset contains 401 entries and 9 columns. Key columns include 'Diagnosis', 'ID', 'Imaginary Part', 'Real Part', 'Gender', 'Age', and 'Smoking', with 'Diagnosis' potentially serving as the target variable for predictive models.

4 sample groups included within the dataset:

(I) Outpatients and hospitalized patients with COPD without acute respiratory infection (COPD).

(II) Outpatients and hospitalized patients with asthma without acute respiratory infections (Asthma).

(III) Patients with respiratory infections, but without COPD or asthma (Infected).

(IV) Healthy controls without COPD, asthma, or any respiratory infection (HC).

The dataset seems to incorporate complex sensor readings, split into imaginary and real components, which are described through minimum and average values in separate columns. 'Gender' is presented as a binary numerical value, likely encoding male and female, while 'Age' provides the patient's age in years. 'Smoking' is a numerical categorical feature, possibly indicating different smoking statuses like non-smoker, current smoker, and ex-smoker.

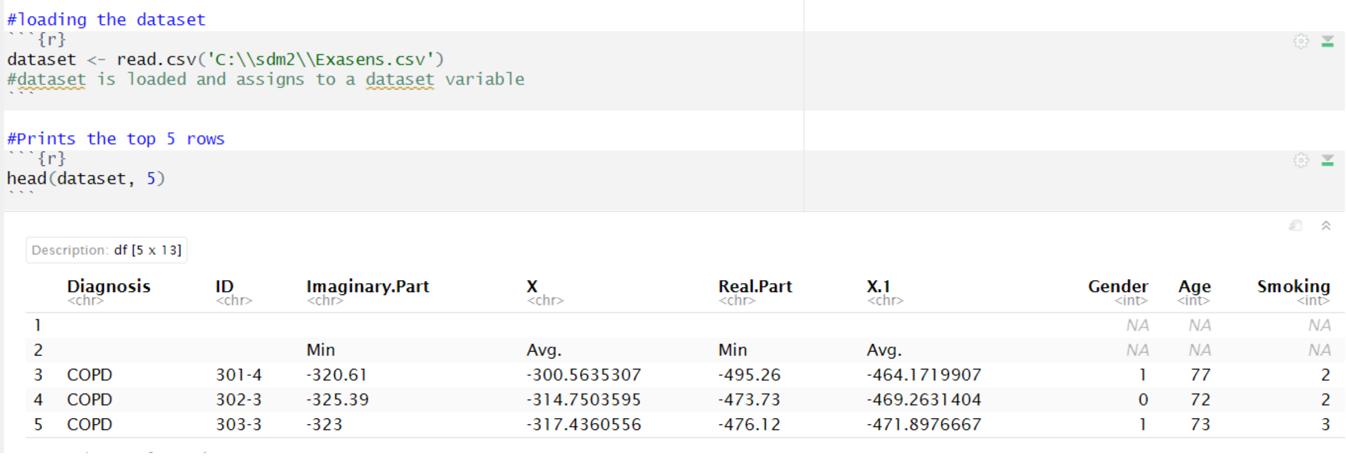
Approach and Methodology:

The approach and methodology used in the project for analyzing the respiratory diseases dataset involved several key stages, each critical for enhancing the understanding and predictive capabilities regarding patient diagnostics. These stages encompass data preparation, exploratory data analysis (EDA), predictive modeling, and clustering analysis.

1. Data Preparation/Data Cleaning:

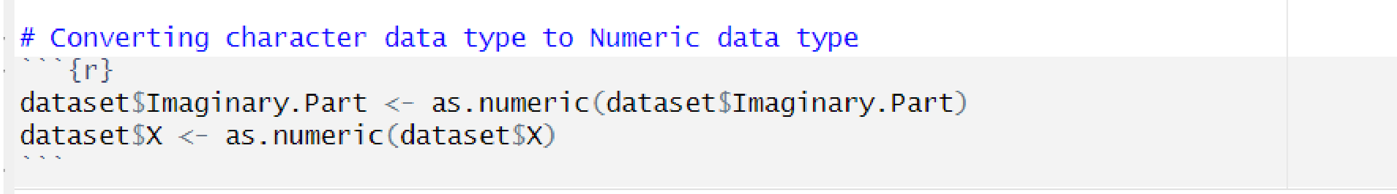
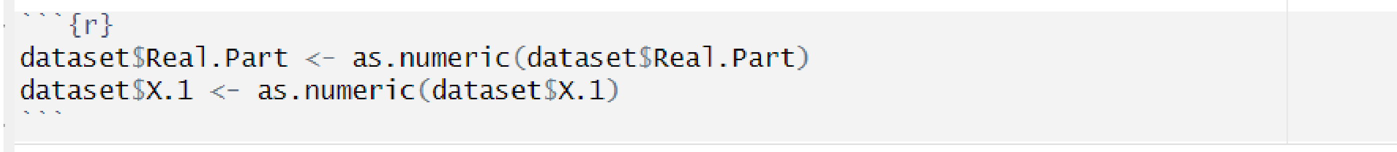
The initial focus was on preparing the raw dataset for analysis, which is pivotal to ensuring the quality and reliability of any subsequent statistical or machine learning analysis.

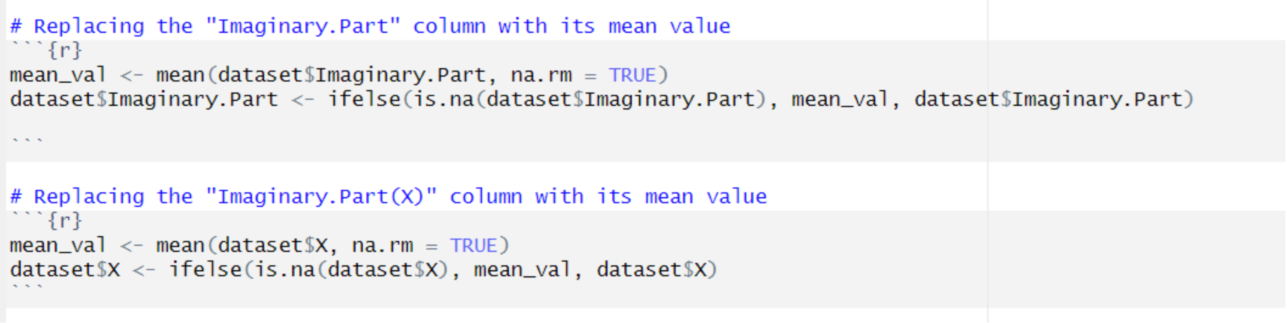
The dataset was first loaded from a CSV file, ensuring the correct parsing of columns and identification of data types.



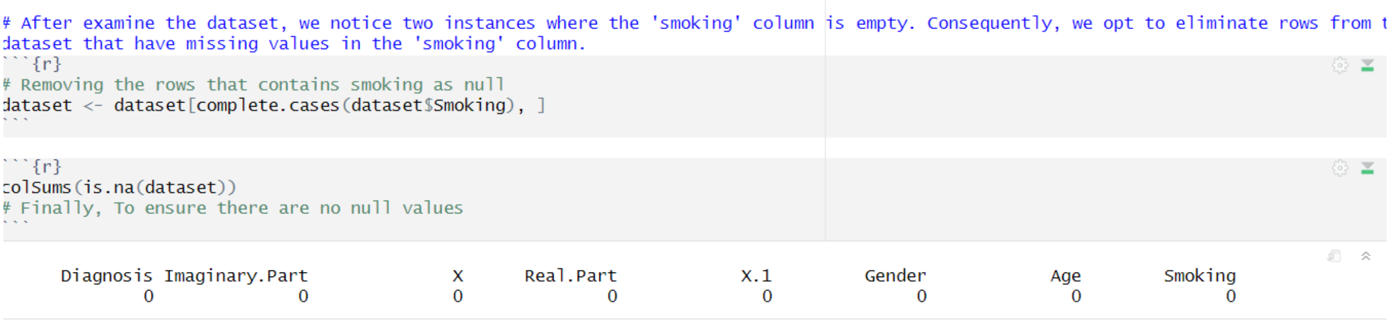
~~~
#loading the dataset
```{r}
dataset <- read.csv('C:\\sdm2\\Exasens.csv')
#dataset is loaded and assigns to a dataset variable
~~~

#Prints the top 5 rows
```{r}
head(dataset, 5)
```
```

```
Description: df [5 x 13]

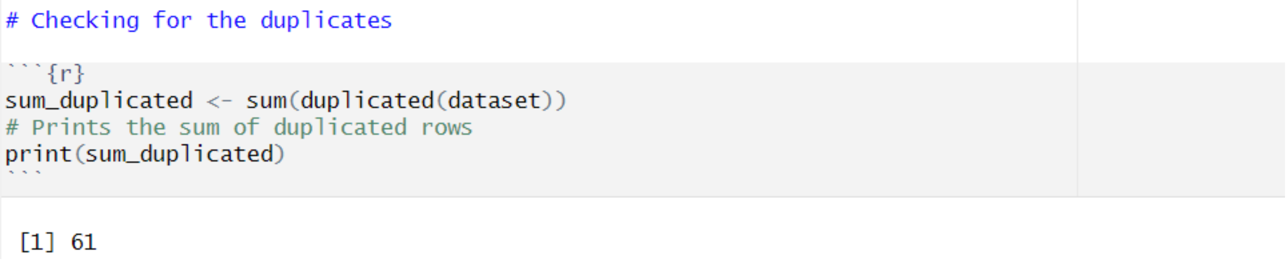
  Diagnosis  ID     Imaginary.Part  X             Real.Part  X.1           Gender  Age  Smoking
  <chr>      <chr>  <chr>           <chr>         <chr>      <chr>          <int> <int>   <int>
1                                                                              NA    NA      NA
2                   Min             Avg.          Min        Avg.              NA    NA      NA
3 COPD       301-4  -320.61         -300.5635307  -495.26    -464.1719907       1    77       2
4 COPD       302-3  -325.39         -314.7503595  -473.73    -469.2631404       0    72       2
5 COPD       303-3  -323            -317.4360556  -476.12    -471.8976667       1    73       3
```

Next several, steps were performed for cleaning the data. Below are some cleaning steps which were performed.

Type Conversion: Character data fields were converted to numeric where necessary, as machine learning models require numerical input for calculations.



```
# Converting character data type to Numeric data type
```{r}
dataset$Imaginary.Part <- as.numeric(dataset$Imaginary.Part)
dataset$X <- as.numeric(dataset$X)
```
```



```
```{r}
dataset$Real.Part <- as.numeric(dataset$Real.Part)
dataset$X.1 <- as.numeric(dataset$X.1)
```
```

Handling Missing Values: Missing data were identified and handled either by imputation (replacing missing values with statistical measures like mean or median) or by removing rows/columns with excessive missing data. **Imputation:** For instance, missing entries in columns representing medical measurements were substituted with the mean of the respective column using **mean()** function, ensuring no loss of data integrity and maintaining the dataset's overall statistical properties. This step is essential to prevent skewing the model's performance.

Below code snippets demonstrates about Imputing missing values:
```



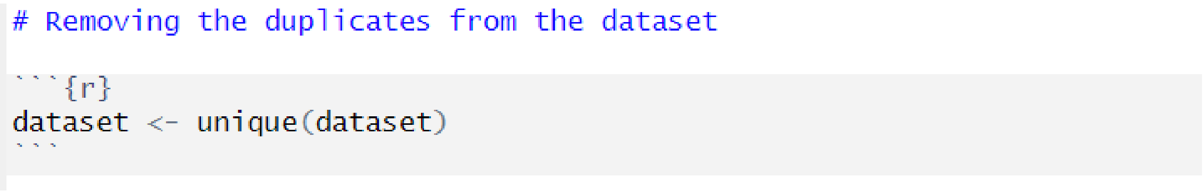
~~~
# Replacing the "Imaginary.Part" column with its mean value
```{r}
mean_val <- mean(dataset$Imaginary.Part, na.rm = TRUE)
dataset$Imaginary.Part <- ifelse(is.na(dataset$Imaginary.Part), mean_val, dataset$Imaginary.Part)

~~~

# Replacing the "Imaginary.Part(X)" column with its mean value
```{r}
mean_val <- mean(dataset$X, na.rm = TRUE)
dataset$X <- ifelse(is.na(dataset$X), mean_val, dataset$X)
```
```

**Deletion:** Entire rows containing missing values, especially in critical columns such as the 'smoking' status, were removed. This approach was chosen to maintain the reliability of analyses, particularly where missing data could significantly skew the results. **complete.cases()** is a function in R that returns a logical vector indicating which cases (i.e., rows) are complete, meaning they have no missing values (NA) in any of the specified columns.



```
After examine the dataset, we notice two instances where the 'smoking' column is empty. Consequently, we opt to eliminate rows from t
dataset that have missing values in the 'smoking' column.
```{r}
# Removing the rows that contains smoking as null
dataset <- dataset[complete.cases(dataset$Smoking), ]
```

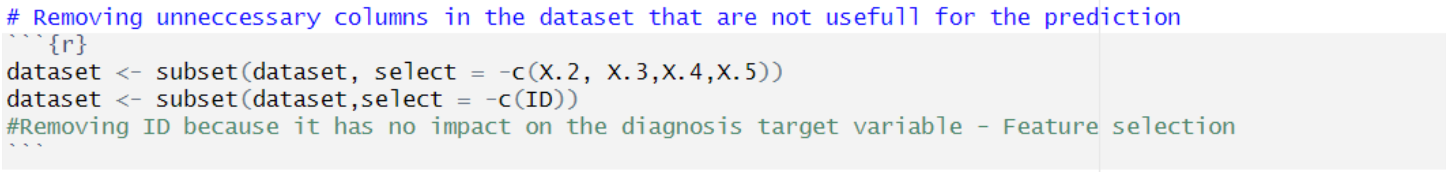
```{r}
colSums(is.na(dataset))
# Finally, To ensure there are no null values
```

     Diagnosis Imaginary.Part   X   Real.Part   X.1   Gender   Age   Smoking
             0   0   0   0   0   0   0   0
```

**Removing Duplicates** The dataset was checked for duplicate entries, and any found duplicates were removed using the **unique()** function in R. This step was crucial as duplicates can artificially inflate the dataset, leading to biased statistical analyses and potentially misleading results. By ensuring each data entry was unique, the project preserved the true variance and structure of the dataset, allowing for more accurate modelling and interpretation of the data.



```
Checking for the duplicates

```{r}
sum_duplicated <- sum(duplicated(dataset))
# Prints the sum of duplicated rows
print(sum_duplicated)
```

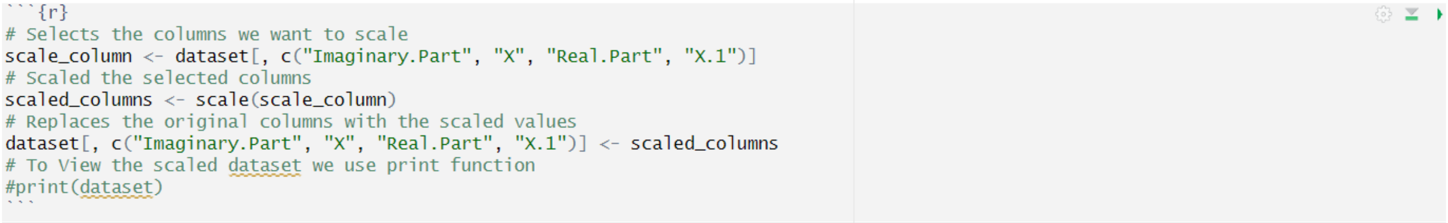
 [1] 61
```



```
Removing the duplicates from the dataset

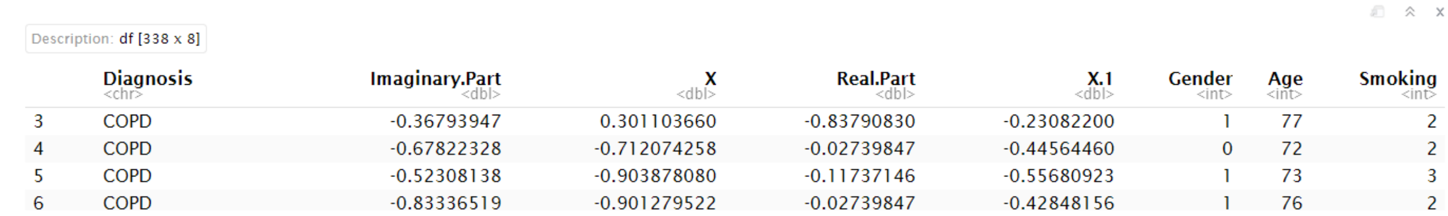
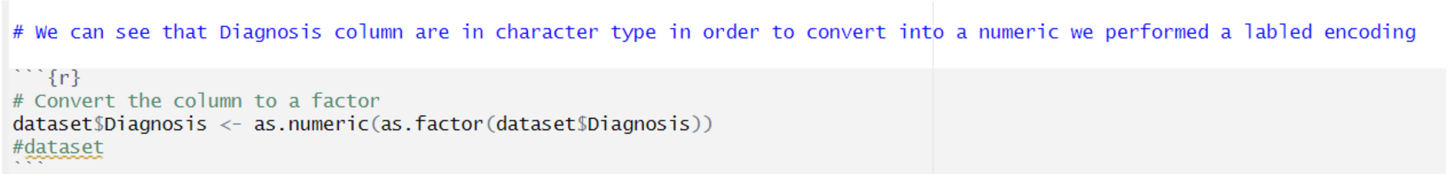
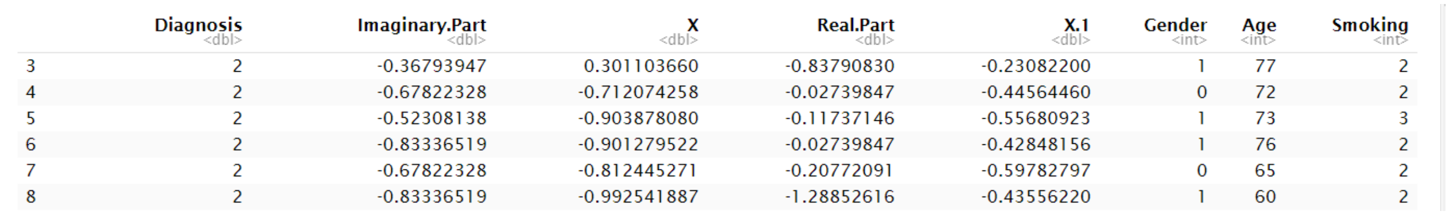
```{r}
dataset <- unique(dataset)
```
```

**Feature Selection:** Irrelevant features, such as IDs that do not contribute to diagnostic outcomes, were removed. This step helps in reducing the dimensionality and focusing the models on the most impactful features.
```



~~~
# Removing unneccessary columns in the dataset that are not usefull for the prediction
```{r}
dataset <- subset(dataset, select = -c(X.2, X.3,X.4,X.5))
dataset <- subset(dataset,select = -c(ID))
#Removing ID because it has no impact on the diagnosis target variable - Feature selection
```
~~~

Scaling: The columns "Imaginary.Part," "X," "Real.Part," and "X.1" were selected for scaling. The **scale()** function in R was applied to these columns to standardize their values, usually by subtracting the mean and dividing by the standard deviation Scaling transforms data to a common scale, enhancing algorithm performance and accuracy by treating all features equally in terms of their potential influence on the predictive outcome, especially important for algorithms sensitive to the magnitude of variables.



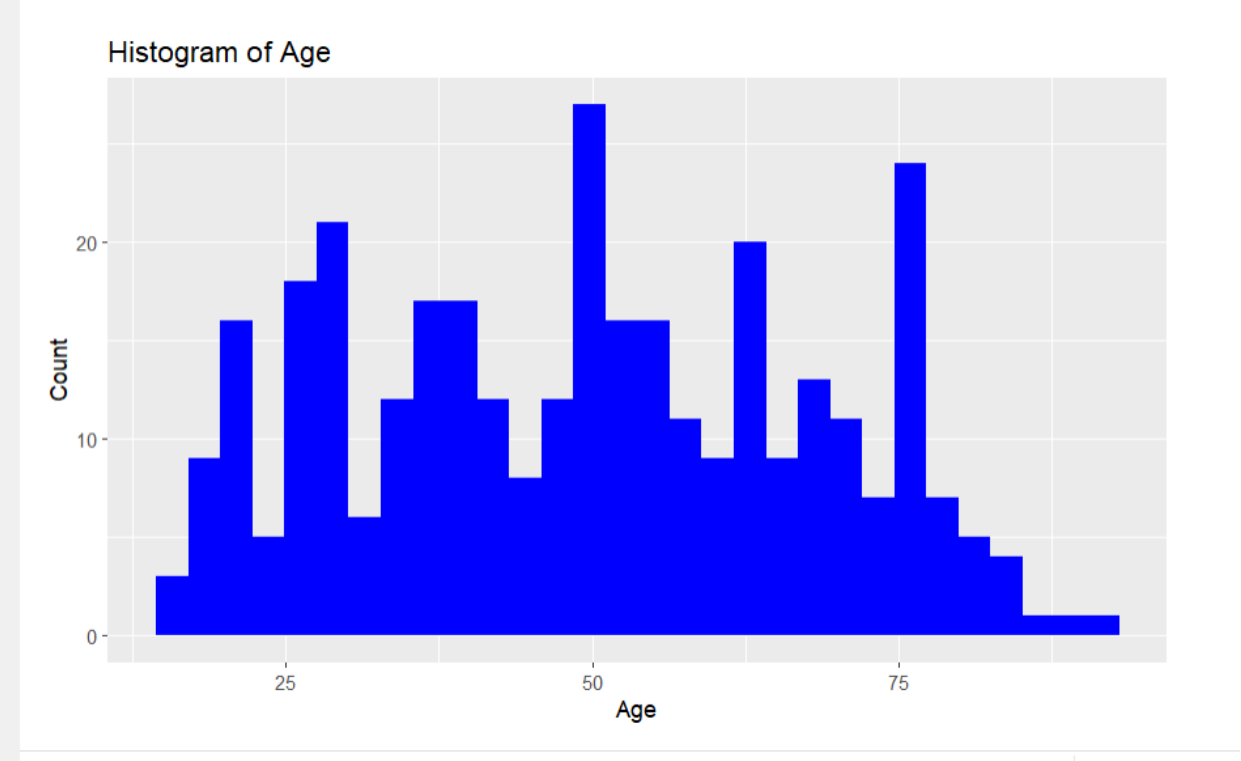
~~~
```{r}
# Selects the columns we want to scale
scale_column <- dataset[, c("Imaginary.Part", "X", "Real.Part", "X.1")]
# Scaled the selected columns
scaled_columns <- scale(scale_column)
# Replaces the original columns with the scaled values
dataset[, c("Imaginary.Part", "X", "Real.Part", "X.1")] <- scaled_columns
# To View the scaled dataset we use print function
#print(dataset)
~~~
```



Description: df [338 x 8]

	Diagnosis <chr>	Imaginary.Part <dbl>	X <dbl>	Real.Part <dbl>	X.1 <dbl>	Gender <int>	Age <int>	Smoking <int>
3	COPD	-0.36793947	0.301103660	-0.83790830	-0.23082200	1	77	2
4	COPD	-0.67822328	-0.712074258	-0.02739847	-0.44564460	0	72	2
5	COPD	-0.52308138	-0.903878080	-0.11737146	-0.55680923	1	73	3
6	COPD	-0.83336519	-0.901279522	-0.02739847	-0.42848156	1	76	2

Label Encoding: label encoding refers to the process of converting the "Diagnosis" column from a character type into a numeric format. Since most machine learning models require numerical input, label encoding is a technique where each unique string value is assigned a numerical identifier. The "Diagnosis" column, which contained character data representing different diagnoses, was transformed into a factor with **as.factor()** function. The factorized "Diagnosis" column was then converted to numeric using **as.numeric()**, which assigns a unique integer to each level of the factor. This process is crucial because it allows for the inclusion of categorical data in the modeling process by providing a numeric representation that can be interpreted by algorithms.



```
# We can see that Diagnosis column are in character type in order to convert into a numeric we performed a labled encoding
```{r}
# Convert the column to a factor
dataset$Diagnosis <- as.numeric(as.factor(dataset$Diagnosis))
#dataset
```
```



	Diagnosis <dbl>	Imaginary.Part <dbl>	X <dbl>	Real.Part <dbl>	X.1 <dbl>	Gender <int>	Age <int>	Smoking <int>
3	2	-0.36793947	0.301103660	-0.83790830	-0.23082200	1	77	2
4	2	-0.67822328	-0.712074258	-0.02739847	-0.44564460	0	72	2
5	2	-0.52308138	-0.903878080	-0.11737146	-0.55680923	1	73	3
6	2	-0.83336519	-0.901279522	-0.02739847	-0.42848156	1	76	2
7	2	-0.67822328	-0.812445271	-0.20772091	-0.59782797	0	65	2
8	2	-0.83336519	-0.992541887	-1.28852616	-0.43556220	1	60	2
```

## 2. Exploratory Data Analysis:

Exploratory Data Analysis (EDA) was conducted to uncover the underlying structure of the data, identify any anomalies, and test underlying assumptions through summary statistics and visual representations. This process included analyzing the distribution of key variables like age and smoking habits through histograms, pie charts, and bar plots.

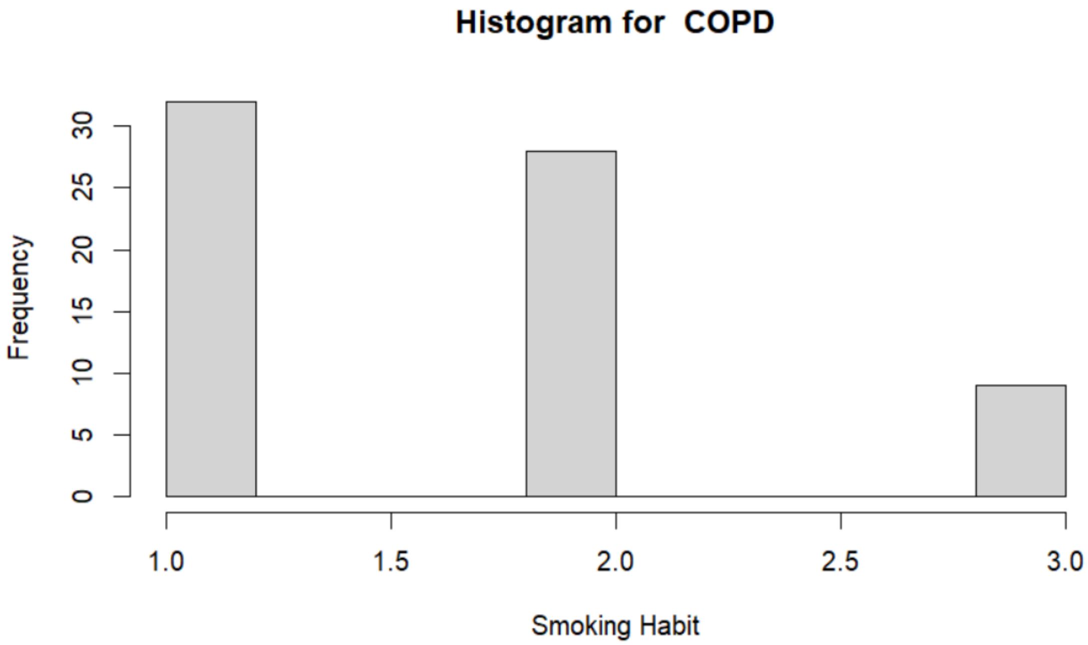
**Histogram:**

The histogram from the dataset displays the age distribution of patients, with distinct age intervals revealing the frequency of patients in each group. The visualization shows an uneven distribution across ages, highlighting a higher concentration of patients in certain age brackets, especially among older adults. This pattern suggests that older age groups might be more represented or more susceptible to the respiratory conditions covered by the dataset. Lower frequencies in other age ranges may indicate a reduced occurrence or reporting of respiratory issues among those ages. The observed peaks in the histogram could be indicative of increased health risks or diagnostic rates associated with advancing age. For healthcare providers, such data is instrumental in tailoring healthcare services to the demographics most affected by respiratory conditions.



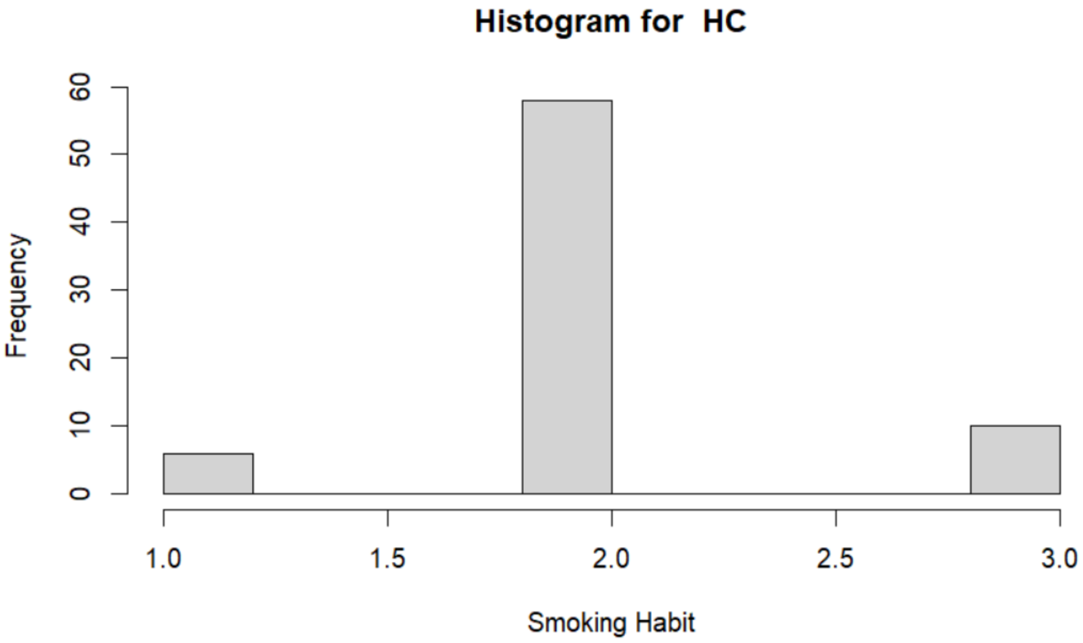
**Histogram for COPD:**

The histogram for COPD patients indicates a predominance of ex-smokers (denoted by the peak at 2.0), followed by non-smokers (1.0), and the least number of active smokers (3.0). This suggests a potential link between smoking cessation and the prevalence of COPD, highlighting ex-smokers as the most significant category in this condition.



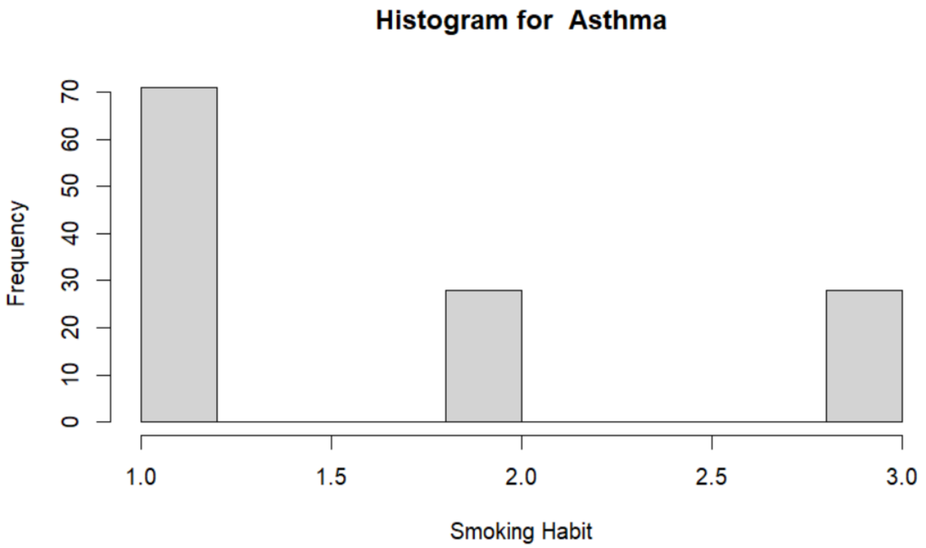
**Histogram for HC:**

In contrast, the histogram for Healthy Controls shows a stark difference, with non-smokers (1.0) being the most significant group, dwarfing the other categories. This could imply a correlation between non-smoking and the absence of respiratory conditions, supporting the notion that non-smoking is associated with better respiratory health.



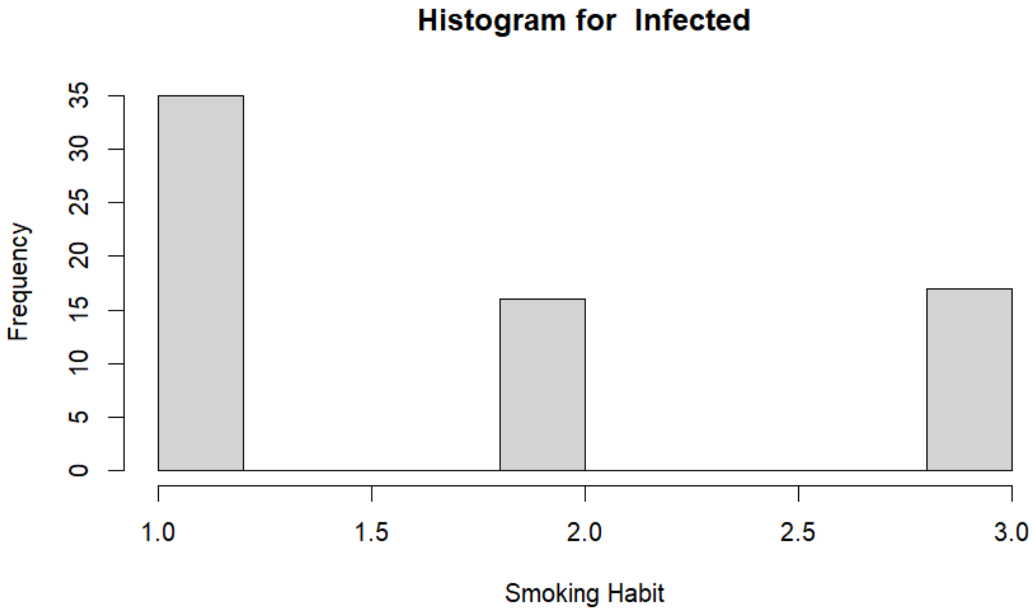
**Histogram for Asthma:**

For Asthma patients, non-smokers (1.0) again form the largest group, indicating that asthma occurs frequently in individuals with no history of smoking. However, the presence of both ex-smokers and active smokers in smaller numbers suggests that smoking is less of a defining characteristic for asthma compared to COPD.



**Histogram for Infected:**

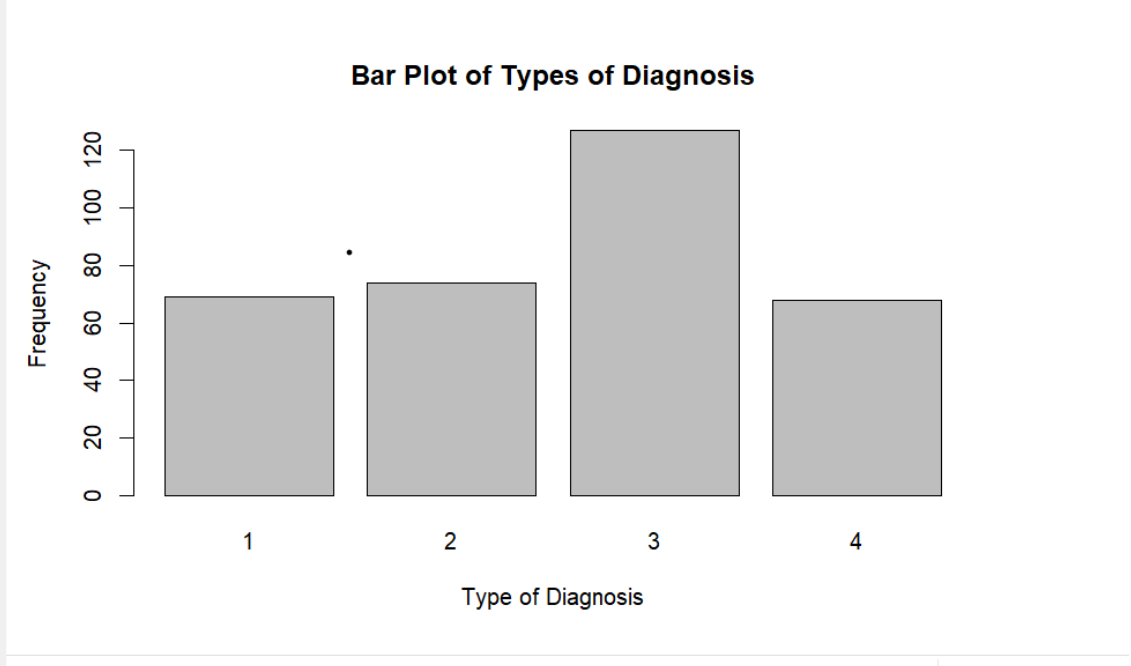
Finally, the histogram for the Infected category displays a more evenly spread distribution between non-smokers and active smokers, with ex-smokers being the least represented. This pattern could point towards a more varied interaction between smoking habits and infections, potentially reflecting the complex nature of infections which can be influenced by factors other than smoking.



**Bar Plot:**

The bar plot illustrates the frequency distribution of various types of diagnoses in the dataset. Each bar represents a different diagnostic category, labeled numerically on the x-axis from 1 to 4. The height of each bar corresponds to the frequency of each diagnosis type within the dataset. From the plot, we can observe that the Asthama (Type 3) has the highest frequency, indicating that it is the most common diagnosis among the patients in this dataset. The HC (Type 2) appears to have a marginally lower frequency but is still significantly represented. In contrast, the COPD (Type 1) and Infected (Type 4) show comparatively lower frequencies, suggesting these diagnoses are less common in the dataset.

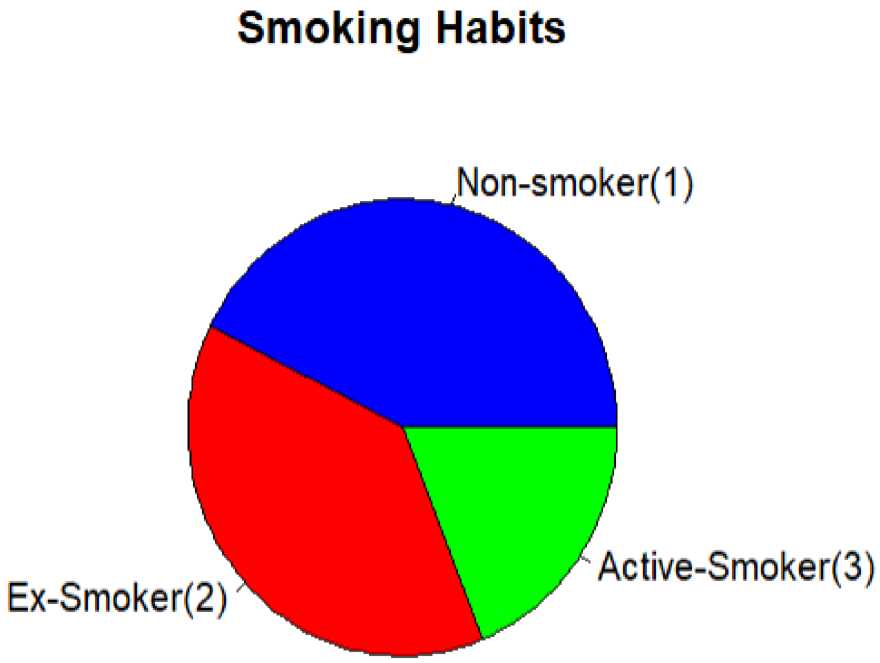
This distribution can provide insights into the prevalence of specific conditions within the patient population and might be reflective of the dataset's demographic or the nature of the respiratory conditions being studied. Healthcare providers and researchers can use this information to prioritize resources and research efforts towards the most common diagnoses observed in the data.



**Pie Plot:**

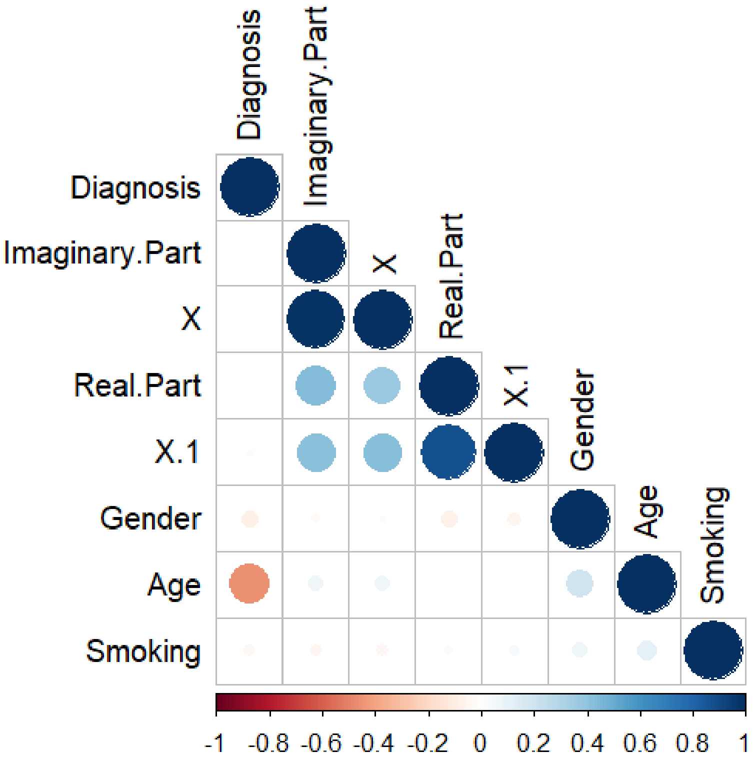
The pie chart illustrates the smoking habits of individuals within the dataset, categorized into three groups: Non-smokers (1), Ex-smokers (2), and Active-smokers (3). Each slice of the pie chart is proportional to the frequency of the corresponding category, with different colors representing each group: blue for Non-smokers, red for Ex-smokers, and green for Active-smokers.

From the visualization, Non-smokers constitute the largest segment, indicated by the blue section, suggesting that they are the most prevalent group within this population. The Ex-smoker category, shown in red, also represents a significant proportion, while the Active-smoker group (in green) is the smallest. This distribution can be critical for public health analysis and planning, as it offers insights into the smoking patterns that could potentially influence respiratory health outcomes.



**Correlation Plot:**

The provided image appears to be a correlation matrix visualized through a correlogram, which displays the correlation coefficients between pairs of variables in the dataset. Each circle's size and color intensity represent the strength and direction of the correlation: larger, darker blue circles indicate stronger positive correlations, while larger, darker red circles suggest strong negative correlations. Variables such as "Real.Part" and "X" show a high positive correlation, indicating that these measurements tend to increase and decrease together. The smaller and lighter colored circles imply weaker relationships, like between "Age" and "Imaginary.Part". Notably, the diagonal shows perfect correlations, as it represents each variable's correlation with itself. This type of visualization is instrumental in quickly identifying relationships and dependencies between variables, which is essential for feature selection and model building in predictive analytics.



## 3. Model Predictions:

Several models were used for making predictions on the Exasens dataset. These models include classification and clustering algorithms. These models provide a good starting point for predictive analysis in medical datasets due to their varied strengths in classification, interpretability, and ease of implementation. The Exasens dataset's predictive models were chosen based on their suitability for binary or multiclass problems, their capacity for interpretable outputs essential in medical diagnostics, and their ability to serve as computationally efficient baselines for initial exploration to identify influential features and assess the dataset's predictive power.

**Classification Algorithms:**

The classification algorithms which are used include:

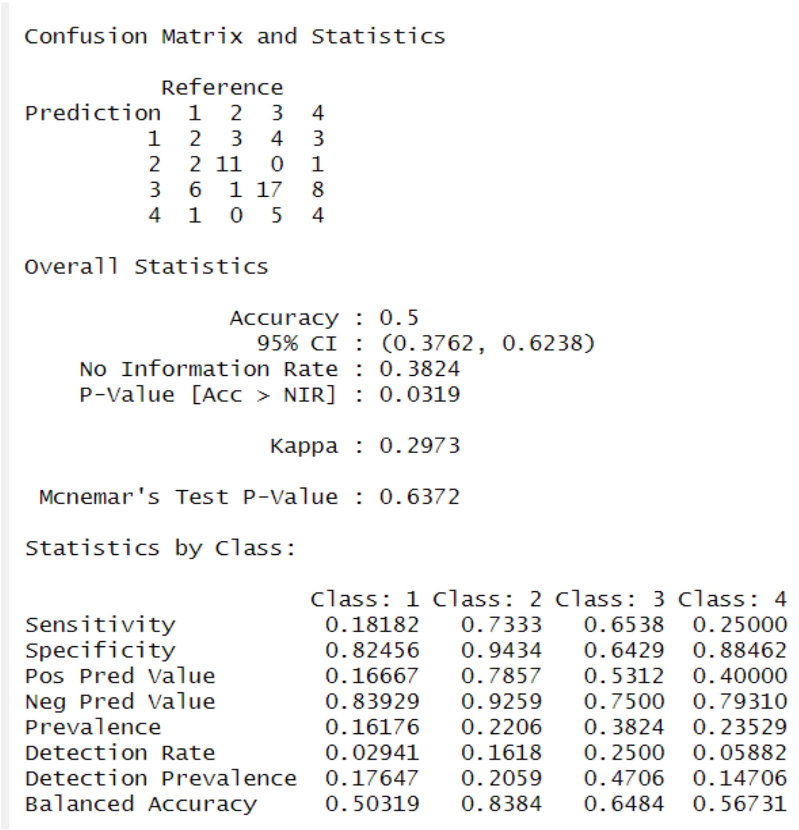
- K-Nearest Neighbors (KNN):
- Decision Tree
- Logistic Regression

**K-Nearest Neighbors (KNN):**

This model was chosen due to its simplicity and effectiveness for classification problems. KNN is a non-parametric, lazy learning algorithm that classifies new cases based on a similarity measure (e.g., distance functions). It was likely selected to leverage the labeled data in the dataset and make predictions based on the closest data points in the feature space.

In the Exasens dataset, The KNN algorithm would classify a new observation based on the majority vote of its 'k' nearest neighbors, considering a set of predictors within the dataset (like age, smoking status, and other clinical measurements). KNN is often chosen for medical datasets because similar patient profiles are likely to lead to similar diagnoses, and it does not make any assumptions about the underlying data distribution.

**Performance Metrics:** The accuracy of the KNN model was calculated, providing a direct indication of how well the algorithm performed on the dataset. Confusion matrix was calculated for this classification algorithm. This matrix presents the performance of the model in terms of actual versus predicted categories across four different classes.



```
Confusion Matrix and Statistics

          Reference
Prediction  1  2  3  4
         1  2  3  4  3
         2  2 11  0  1
         3  6  1 17  8
         4  1  0  5  4

Overall Statistics

               Accuracy : 0.5
                 95% CI : (0.3762, 0.6238)
    No Information Rate : 0.3824
    P-Value [Acc > NIR] : 0.0319

                  Kappa : 0.2973

 Mcnemar's Test P-Value : 0.6372

Statistics by Class:

                     Class: 1 Class: 2 Class: 3 Class: 4
Sensitivity           0.18182   0.7333   0.6538  0.25000
Specificity           0.82456   0.9434   0.6429  0.88462
Pos Pred Value        0.16667   0.7857   0.5312  0.40000
Neg Pred Value        0.83929   0.9259   0.7500  0.79310
Prevalence            0.16176   0.2206   0.3824  0.23529
Detection Rate        0.02941   0.1618   0.2500  0.05882
Detection Prevalence  0.17647   0.2059   0.4706  0.14706
Balanced Accuracy     0.50319   0.8384   0.6484  0.56731
```

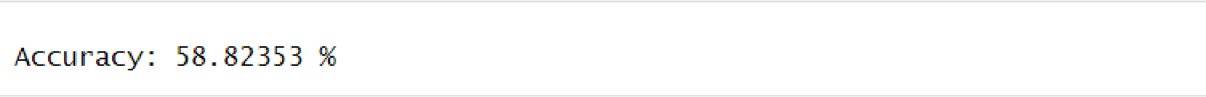
The accuracy of the model is 50%, which is quite low, only slightly better than a random guess in a binary classification. The confidence interval for accuracy (95% CI) is between 0.3762 and 0.6238, indicating a high level of uncertainty about the accuracy. The No Information Rate (NIR), which shows the accuracy that could be achieved by always predicting the most frequent class, is 0.3824, and the fact that the P-Value (Acc > NIR) is 0.0319 indicates that the model is doing better than the NIR.

The confusion matrix shows the KNN model's accuracy at 50%, with a Kappa statistic of 0.2973, hinting at fair consistency in predictions. Class 2 has the highest sensitivity at 73.33%, indicating it is most accurately identified, while Class 4's specificity of 88.46% suggests it is reliably distinguished from other classes. However, the overall predictive performance is relatively low, as evidenced by the P-value of 0.0319, which though indicates statistical significance, is close to the threshold.

Decision Tree:

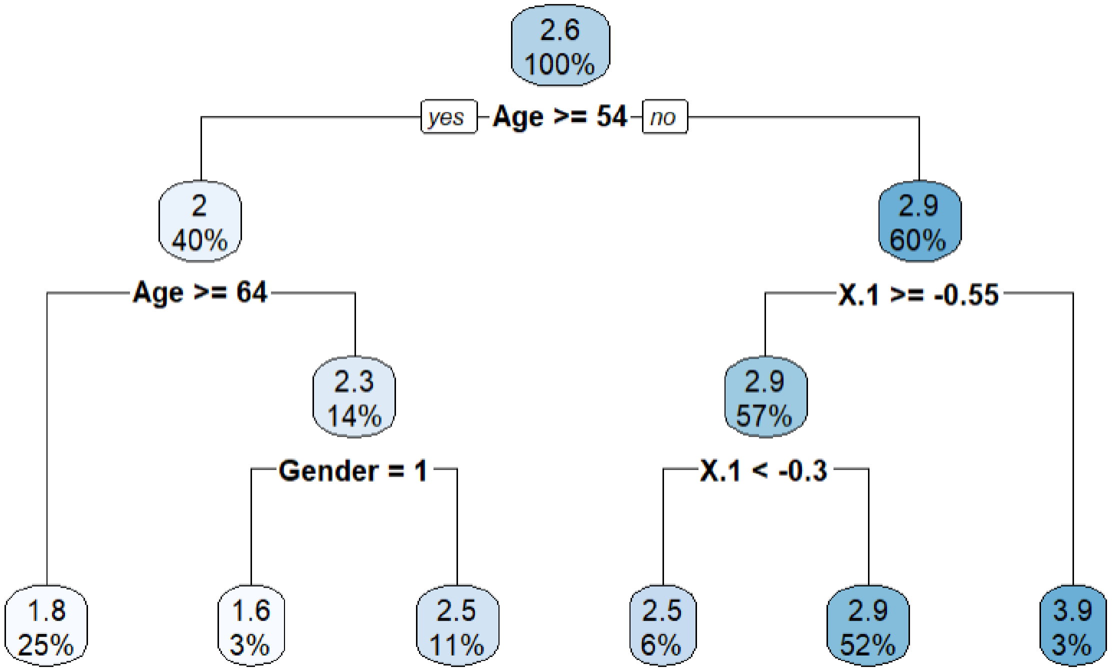
Decision Trees are a flowchart-like tree structure where an internal node represents a feature (or attribute), the branch represents a decision rule, and each leaf node represents the outcome. In the Exasens dataset, decision trees have been used to model the decision rules that lead to specific diagnoses .They are particularly advantageous in a clinical setting because they mimic human decision-making more closely than other techniques and their outcomes are easy to interpret.

Performance Metrics: The accuracy of the decision tree model was calculated, providing a direct indication of how well the algorithm performed on the dataset. The reported accuracy of around 58.82% indicates that the decision tree could capture the patterns in the data reasonably well.



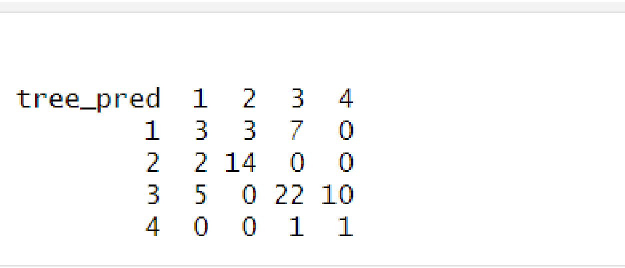
```
Accuracy: 58.82353 %
```

Below is the plot for the decision tree. This particular tree splits on features such as age and a variable labeled 'X.1', indicating their importance in predicting the diagnosis.



Analysis of the plot shows that age is the primary decision rule at the top-level: patients 54 years or older follow one branch, while younger patients follow another. Further splits are made on age (64 years and above), and gender for older patients, while the 'X.1' variable differentiates among the younger patients. Each leaf node of the tree represents the predicted class, with the proportion of samples in that class shown as a percentage. This decision tree helps to visualize the hierarchy of decision-making based on the patient data and indicates that age is a significant factor in the diagnosis process.

Below is the confusion matrix for decision tree:



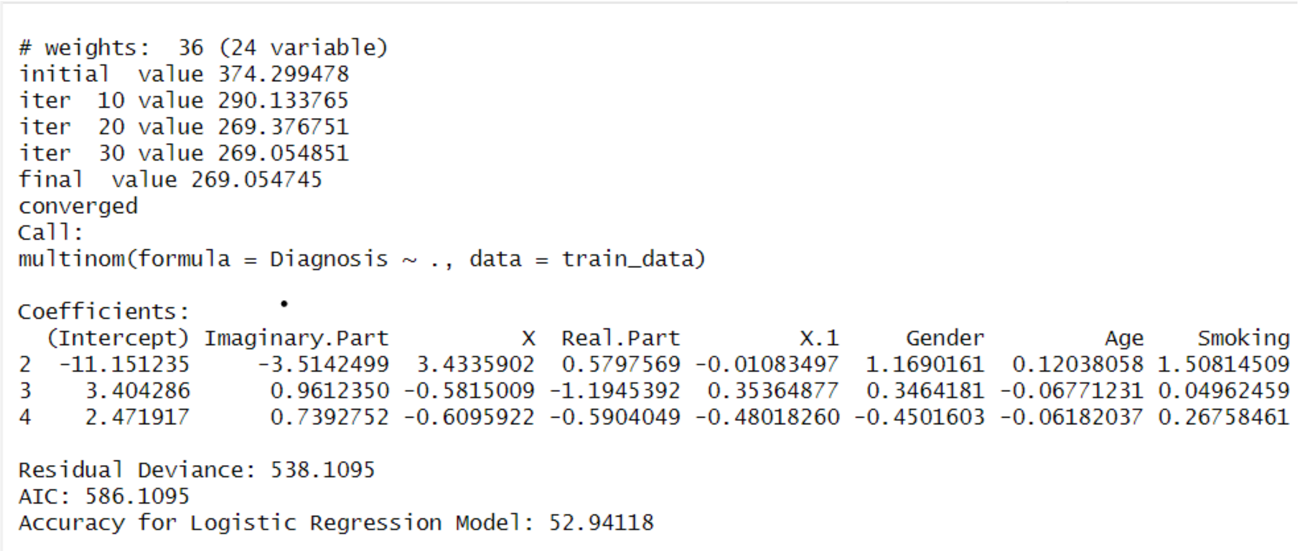
```
tree_pred  1  2  3  4
        1  3  3  7  0
        2  2 14  0  0
        3  5  0 22 10
        4  0  0  1  1
```

The confusion matrix for the decision tree model shows the predictions versus the actual classifications of the test data. It indicates that the model predicts Class 3 with high accuracy (22 true positives out of 32 predictions), but struggles with Class 4, often confusing it with Class 3 (10 false positives). Class 2 is well-predicted with 14 true positives , while Class 1 has a mixed result with more false negatives (7) than true positives (3). This matrix helps in identifying which classes are being predicted correctly and which ones are being confused by the model.

Logistic Regression:

Logistic Regression is used for binary classification problems. In the context of the Exasens dataset, it likely refers to multinomial logistic regression, which is an extension of logistic regression that allows for more than two categories of the dependent or target variable. It was probably utilized to predict the probability of each type of diagnosis category based on the input variables. It's a robust classifier that can also provide the probability of the target classes, which is very informative in a clinical diagnosis scenario.

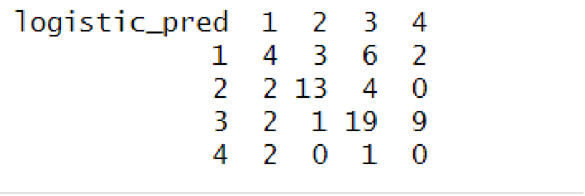
Performance Metrics: The logistic regression model had an accuracy of 52.94%, meaning it correctly predicted outcomes a little more than half the time, providing a reasonable but not high level of predictive performance.



```
# weights:  36 (24 variable)
initial  value 374.299478
iter  10 value 290.133765
iter  20 value 269.376751
iter  30 value 269.054851
final  value 269.054745
converged
Call:
multinom(formula = Diagnosis ~ ., data = train_data)

Coefficients:
  (Intercept) Imaginary.Part          X  Real.Part         X.1    Gender         Age    Smoking
2  -11.151235     -3.5142499  3.4335902  0.5797569 -0.01083497 1.1690161  0.12038058 1.50814509
3    3.404286      0.9612350 -0.5815009 -1.1945392  0.35364877 0.3464181 -0.06771231 0.04962459
4    2.471917      0.7392752 -0.6095922 -0.5904049 -0.48018260 -0.4501603 -0.06182037 0.26758461

Residual Deviance: 538.1095
AIC: 586.1095
Accuracy for Logistic Regression Model: 52.94118
```

Above snippet indicates that the model, which is designed to handle multiple classes for the dependent variable 'Diagnosis', has converged, indicating that the fitting process was successful. The output displays the coefficients for the model, which quantify the relationship between each predictor variable (like 'Imaginary.Part', 'X', 'Real.Part', 'X.1', 'Gender', 'Age', 'Smoking') and the log odds of the outcomes. The coefficient values can be interpreted in terms of their impact on the likelihood of each diagnosis. For instance, the negative coefficient for 'Imaginary.Part' in the second class suggests that higher values for this predictor are associated with lower odds of the second diagnosis. The model achieved an accuracy of approximately 52.9418%, suggesting that it correctly predicted the diagnosis over half of the time. While this indicates a level of predictive ability, it also points to room for improvement, as the accuracy is relatively modest.

Below is the confusion matrix for the logistic regression:



```
logistic_pred  1  2  3  4
            1  4  3  6  2
            2  2 13  4  0
            3  2  1 19  9
            4  2  0  1  0
```

The confusion matrix for the logistic regression model displays the counts of correct and incorrect predictions across four different classes. The model shows a tendency to predict Class 2 and Class 3 with higher accuracy, indicated by higher counts of 13 and 19 on the diagonal for these classes, respectively. However, it struggles with Class 4, often misclassifying it as Class 3, as shown by the 9 off-diagonal entries. The confusion across classes 1, 2, and 3 suggests the model has difficulty differentiating between these categories, while Class 4 is more likely to be underrepresented or more challenging to predict accurately.

Clustering Algorithms:

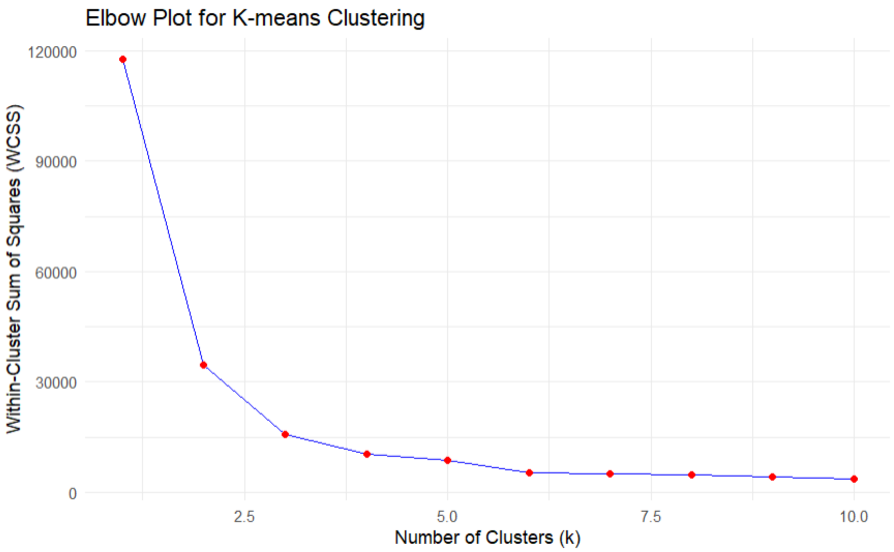
clustering algorithms are used to uncover natural groupings or patterns in the Exasens dataset without the use of predefined labels. The clustering techniques used are:

- K-means clustering
- Hierarchical clustering

K-Means Clustering:

K-Means is a centroid-based clustering algorithm that partitions the dataset into K distinct, non-overlapping subgroups or clusters. It would have been used to classify data points into clusters based on their similarity. The algorithm requires the number of clusters (K) to be specified in advance. It then randomly initializes K centroids and iteratively assigns each data point to the nearest centroid while minimizing the within-cluster sum of squares (WCSS).

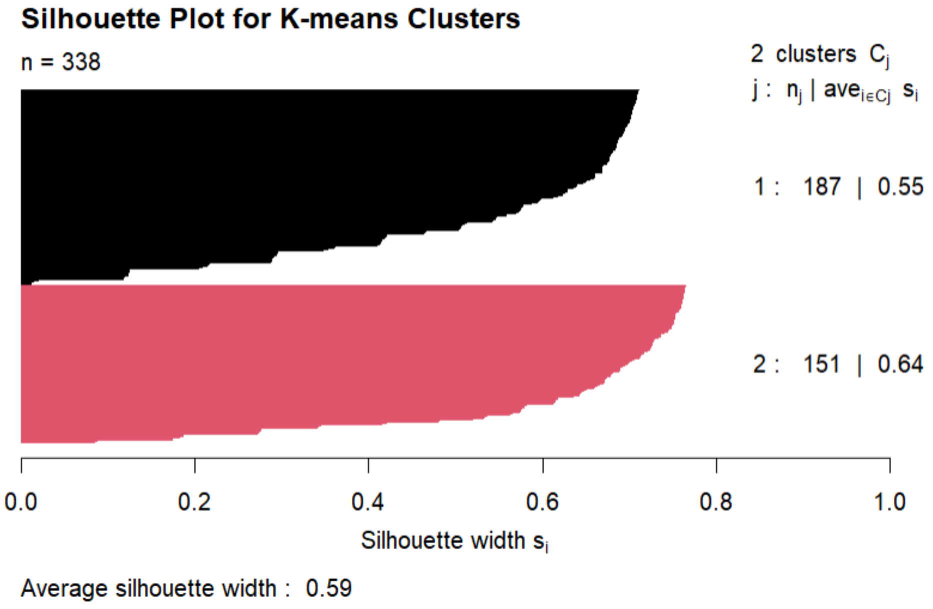
Selection of K: An 'elbow plot' was likely used to determine the optimal number of clusters by plotting the WCSS against different K values. The elbow point, where the rate of decrease sharply changes, signifies the most appropriate number of clusters.



The elbow plot is used to determine the optimal number of clusters (k) for K-means clustering by displaying the within-cluster sum of squares (WCSS) against the number of clusters. If the elbow plot is to indicate k=2 as the optimal number of clusters, then the "elbow" or the point of inflection would be where the plot starts to flatten out after k=2, instead of k=3. This would mean that the reduction in within-cluster sum of squares (WCSS) from 1 to 2 is significant, while the reduction from 2 to 3 is not as pronounced, thus k=2 would be the most efficient choice for clustering this particular dataset.

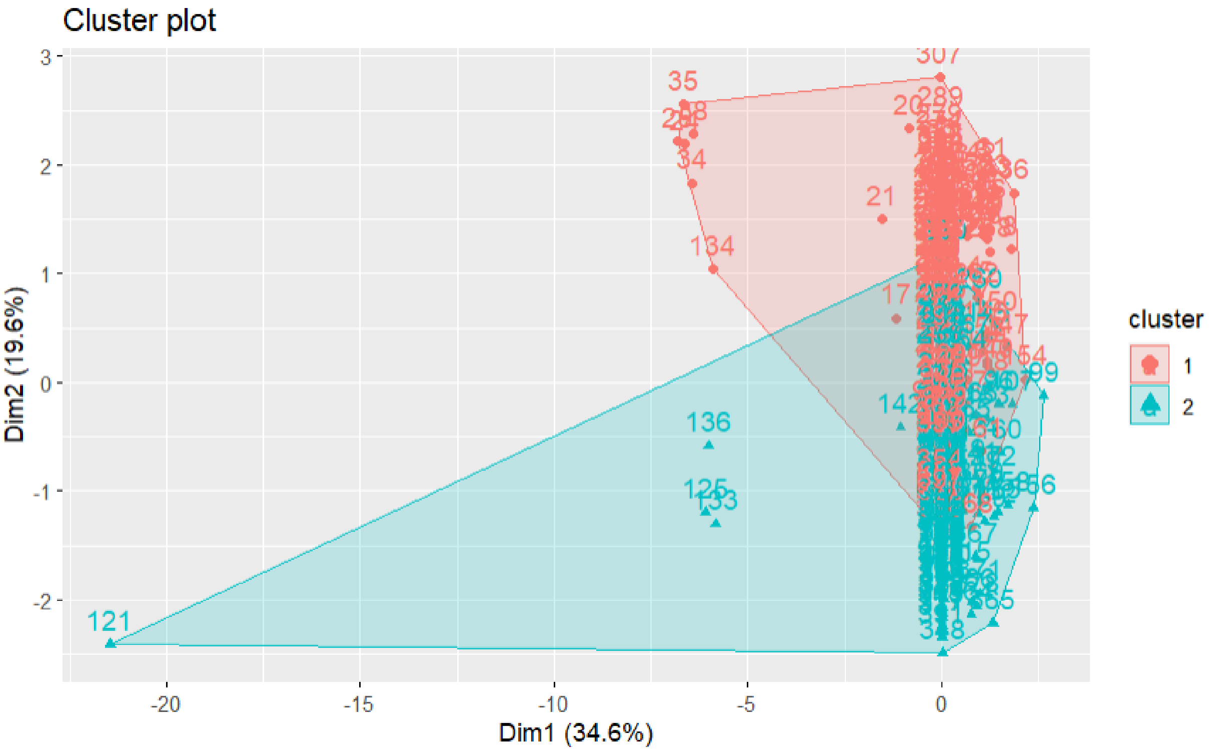
For k = 2 we are getting the high silhouette value than k = 3

Silhoutte Analysis: After choosing the number of clusters, silhouette analysis was performed. It measures how similar a data point is to its own cluster (cohesion) compared to other clusters (separation). A high silhouette score indicates that the object is well matched to its own cluster and poorly matched to neighboring clusters.



The silhouette plot for K-means clustering indicates two clusters with an average silhouette width of 0.59, suggesting a moderately strong structure within the data. Each cluster has its own silhouette width, with Cluster 1 having a width of 0.55 and Cluster 2 a width of 0.64. These values indicate that most points in both clusters are well matched to their own cluster and relatively poorly matched to neighboring clusters, with Cluster 2 being slightly more compact and separated than Cluster 1.

Cluster Plot: The cluster plot visualizes data points grouped into two clusters, indicated by different colors and shapes: red circles for Cluster 1 and blue triangles for Cluster 2. The plot demonstrates a clear separation between the two clusters along the first dimension (Dim1), which explains 34.6% of the variance, suggesting that this dimension is a strong differentiator between the clusters. The second dimension (Dim2) accounts for 19.6% of the variance and shows some overlap between clusters, indicating that it is a less distinctive but still significant feature for clustering. The relatively tight grouping within each cluster and the distance between the two clusters indicate that the K-means algorithm effectively partitioned the data into meaningful subgroups.

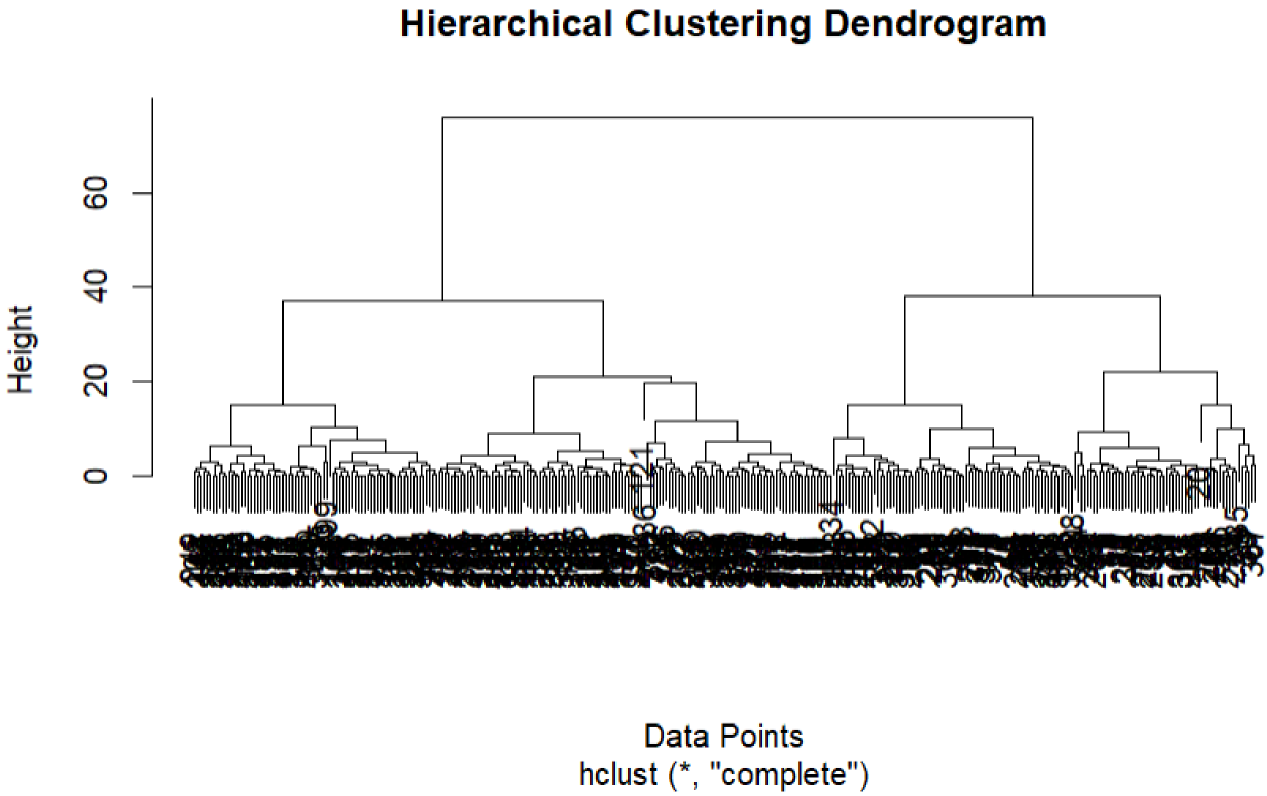


Hierarchical Clustering:

Hierarchical Clustering is a connectivity-based clustering method that builds a tree of clusters by progressively merging or splitting existing groups. The algorithm starts by treating each data point as a single cluster and then iteratively merges the closest pairs of clusters until all points are merged into a single cluster, creating a dendrogram (tree diagram).

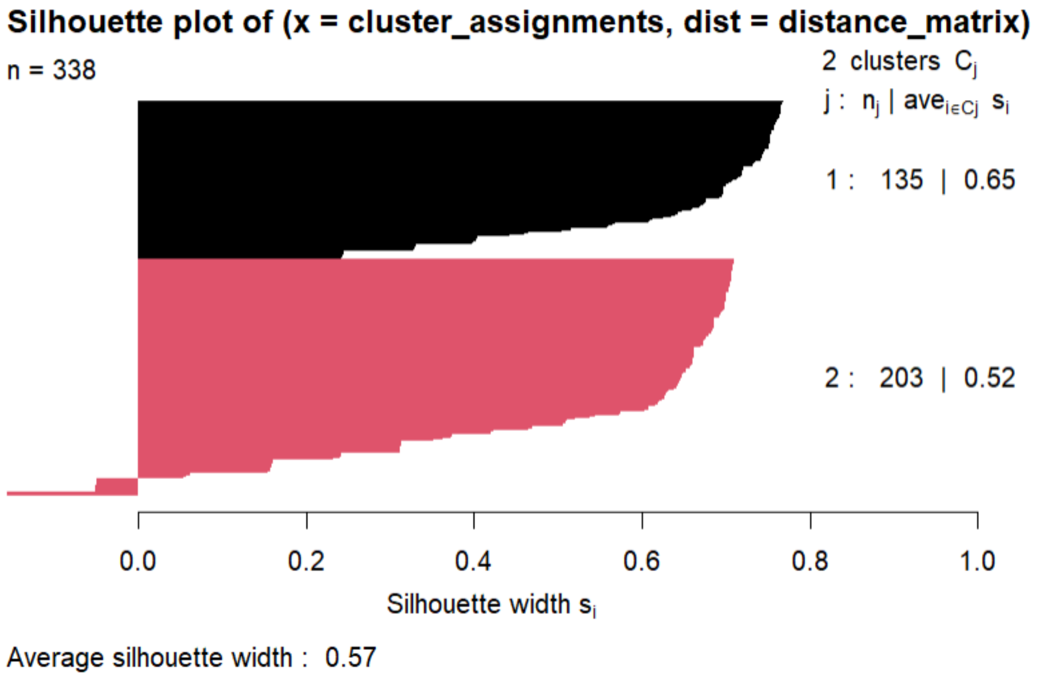
Dendrogram Analysis:

The dendrogram represents the result of a hierarchical clustering analysis using complete linkage, depicting how data points are grouped together at various levels of similarity. The 'height' in the dendrogram indicates the distance at which clusters are merged, with larger heights reflecting less similarity between clusters. In this dendrogram, there are several small clusters that merge at low heights, indicating high similarity within those groups. As we move up the height, clusters combine to form larger clusters, with the largest mergers happening at the greatest heights, suggesting these groups are less similar to each other. This visualization helps identify the natural groupings in the data and, by cutting the dendrogram at a specific height, can determine the number of clusters that best represents the data structure. The distance metric used is the Euclidean distance.



Silhouette Analysis:

The silhouette plot provides a graphical representation of how well each object lies within its cluster, a metric known as the silhouette width, for hierarchical clustering. With an average silhouette width of 0.57, the clusters can be considered reasonably well-defined. Cluster 1, with 135 data points, has a higher silhouette width of 0.65, suggesting that its data points are, on average, more appropriately placed within the cluster compared to Cluster 2, which has 203 data points and a silhouette width of 0.52. These widths indicate that while there's a good degree of separation between the clusters, Cluster 1 is more cohesive than Cluster 2.



Conclusion:

The clustering algorithms, K-means with **k=2** and hierarchical clustering, have yielded relatively close silhouette scores of 0.58 and 0.57, respectively, indicating that both methods provided a similar quality of cluster separation and coherence. However, the marginally higher silhouette score of K-means suggests it may have performed slightly better at grouping the data into two distinct clusters. For the classification algorithms, the decision tree outperformed both KNN and logistic regression with an accuracy of 58.3%, suggesting it was more effective at predicting the correct class labels in this particular dataset. KNN had the lowest performance at 50% accuracy, which is only as good as random guessing in a two-class problem, indicating that the algorithm may not be capturing the complexities of the dataset. Logistic regression was slightly better than KNN but still underperformed compared to the decision tree with an accuracy of 52.9%.

Finally, for clustering, K-means might be the preferred method for this dataset, while for classification purposes, the decision tree algorithm should be considered over KNN and logistic regression, given its superior performance in accuracy.