Phase 3

Project Phase 3

Mira Madi & Rita Rizkallah

1-About Dataset

This dataset can used to predict possibility or risk of getting other diseases.This includes the details of individuals having various disorders.Very useful for corona risk prediction. It has attributes like age, diabetes, cancer, etc. The dataset is from kaggle. Outcome: risk of getting another disease. (the possibility of comorbidities)

2-Data Preprocessing

Data preprocessing is important because it ensures that the data is clean and that we end up with the most accurate results possible.

First, we set the working directory, read the dataset file and view it.

Then, we invoked the str function which provides a summary of the dataset, including its type, the number of observations, and variables (attributes). It also gives the type of each attribute.

We also invoked the summary function which gives the "5-number summary", for our data it is only relevant for the age and bmi attributes, since the rest are qualitative attributes

colnames(df) prints out the names of the columns which represent the attributes.

```
## 'data.frame':    999 obs. of  13 variables:
##  $ Age                          : int  50 31 32 21 33 30 26 29 53 54 ...
##  $ Gender                       : chr  "Male" "Male" "Female" "Female" ...
##  $ BMI                          : num  30.1 26.6 23.3 28.1 43.1 25.6 31 35.3 30.5 23.4 ...
##  $ Diabetes                     : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ Cardio_Vascular_Diseases     : int  0 1 1 1 0 1 1 1 1 1 ...
##  $ Sickle_cell_diesases         : int  0 0 0 0 1 0 0 0 0 0 ...
##  $ Immuno_deficiency_disease    : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Down_syndrome                : int  0 0 0 0 1 0 0 0 0 0 ...
##  $ Cancer                       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Chronic_Respiratory_disease: int  1 1 1 1 1 1 1 1 1 1 ...
##  $ Hypertension                 : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ Vaccinated                   : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ Outcome                      : int  1 1 1 1 1 1 1 1 1 1 ...
```

```
##       Age           Gender               BMI           Diabetes
##  Min.   :18.00   Length:999         Min.   : 0.00   Min.   :0.0000
##  1st Qu.:24.00   Class :character   1st Qu.:27.10   1st Qu.:0.0000
##  Median :30.00   Mode  :character   Median :32.00   Median :1.0000
```

```
##  Mean   :33.08                      Mean   :31.94   Mean   :0.6607
##  3rd Qu.:40.00                      3rd Qu.:36.50   3rd Qu.:1.0000
##  Max.   :90.00                      Max.   :67.10   Max.   :1.0000
##  Cardio_Vascular_Diseases Sickle_cell_diesases Immuno_deficiency_disease
##  Min.   :0.0000           Min.   :0.00000      Min.   :0.00000
##  1st Qu.:0.0000           1st Qu.:0.00000      1st Qu.:0.00000
##  Median :0.0000           Median :0.00000      Median :0.00000
##  Mean   :0.2593           Mean   :0.07508      Mean   :0.01201
##  3rd Qu.:1.0000           3rd Qu.:0.00000      3rd Qu.:0.00000
##  Max.   :1.0000           Max.   :1.00000      Max.   :1.00000
##  Down_syndrome         Cancer        Chronic_Respiratory_disease
##  Min.   :0.00000   Min.   :0.00000   Min.   :0.0000
##  1st Qu.:0.00000   1st Qu.:0.00000   1st Qu.:0.0000
##  Median :0.00000   Median :0.00000   Median :0.0000
##  Mean   :0.02302   Mean   :0.04905   Mean   :0.3854
##  3rd Qu.:0.00000   3rd Qu.:0.00000   3rd Qu.:1.0000
##  Max.   :1.00000   Max.   :1.00000   Max.   :1.0000
##   Hypertension      Vaccinated        Outcome
##  Min.   :0.0000   Min.   :0.0000   Min.   :0.0000
##  1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.0000
##  Median :1.0000   Median :1.0000   Median :0.0000
##  Mean   :0.5666   Mean   :0.5536   Mean   :0.1792
##  3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.:0.0000
##  Max.   :1.0000   Max.   :1.0000   Max.   :1.0000
```

```
##  [1] "Age"                          "Gender"
##  [3] "BMI"                          "Diabetes"
##  [5] "Cardio_Vascular_Diseases"     "Sickle_cell_diesases"
##  [7] "Immuno_deficiency_disease"    "Down_syndrome"
##  [9] "Cancer"                       "Chronic_Respiratory_disease"
## [11] "Hypertension"                 "Vaccinated"
## [13] "Outcome"
```

In order to check for missing values, we first invoke the dim function which gives the number of observations and the number of attributes, then invoke the na.omit function which removes missing values and then we recheck using dim function, the result did not change meaning that we don't have any missing values. This ensures that our data is complete!

```
## [1] 999  13
```

```
## [1] 999  13
```

If duplicates are present, it is always better if they are removed, as they can introduce correlated error terms, which lead to the prediction intervals being narrower than they should be. So we invoked duplicated function which showed that we had 2 rows that are duplicated, to remove them we used !d uplicated function and saved the result to a new dataframe.

```
##     Age Gender  BMI Diabetes Cardio_Vascular_Diseases Sickle_cell_diesases
## 630  21 Female 24.7        1                        0                    0
## 634  22   Male 27.5        1                        0                    0
##     Immuno_deficiency_disease Down_syndrome Cancer Chronic_Respiratory_disease
## 630                         0             0      0                           0
```

```
## 634                          0            0      0                            0
##     Hypertension Vaccinated Outcome
## 630            0          1       0
## 634            0          1       0
```

```
##  [1] Age                          Gender
##  [3] BMI                          Diabetes
##  [5] Cardio_Vascular_Diseases     Sickle_cell_diesases
##  [7] Immuno_deficiency_disease    Down_syndrome
##  [9] Cancer                       Chronic_Respiratory_disease
## [11] Hypertension                 Vaccinated
## [13] Outcome
## <0 rows> (or 0-length row.names)
```

We suspect that there is a correlation between a high BMI and an increased risk of diabetes, thus we created a subset of data taking BMI > 25, and this showed that our doubts might be correct.

To check if we have logical values of age and BMI we ran the function unique. It showed the possible values for age and BMI that are present in our data and it seems that they are all within the normal range of logical values, thus we do not have any unexpected random values.

```
##  [1] 50 31 32 21 33 30 26 29 53 54 34 57 59 51 27 41 43 22 38 60 28 45 35 46 56
## [26] 37 48 40 25 24 58 42 44 39 36 23 61 69 62 55 65 47 52 66 49 63 67 72 81 64
## [51] 70 68 19 20 18 90
```

```
##   [1] 30.1 26.6 23.3 28.1 43.1 25.6 31.0 35.3 30.5 23.4 37.6 38.0 27.1 25.8 30.0
##  [16] 45.8 29.6 43.3 34.6 39.3 35.4 39.8 29.0 36.6 31.1 39.4 23.2 22.2 34.1 36.0
##  [31] 31.6 24.8 19.9 27.6 24.0 33.2 32.9 38.2 37.1 34.0 40.2 22.7 45.4 27.4 42.0
##  [46] 29.7 28.0 39.1 16.5 19.4 24.2 24.4 33.7 34.7 23.0 37.7 46.8 40.5 41.5 18.3
##  [61] 25.0 25.4 32.8 32.5 42.7 19.6 28.9 28.6 43.4 35.1 32.0 24.7 32.6 43.2 22.4
##  [76] 29.3 24.6 48.8 32.4 38.5 26.5 19.1 46.7 23.8 33.9 20.4 28.7 49.7 39.0 26.1
##  [91] 22.5 39.6 29.5 34.3 37.4 33.3 31.2 28.2 53.2 34.2 33.6 26.8 55.0 42.9 34.5
## [106] 27.9 38.3 21.1 33.8 30.8 36.9 39.5 15.4 27.3 21.9 40.6 47.9 50.0 25.2 40.9
## [121] 37.2 44.2 29.9 31.9 28.4 43.5 32.7 67.1 45.0 34.9 27.7 35.9 22.6 33.1 30.4
## [136] 52.3 24.3 22.9 34.8 30.9 40.1 23.9 37.5 35.5 42.8 42.6 41.8 35.8 37.8 28.8
## [151] 23.6 35.7 36.7 45.2 44.0 46.2 35.0 43.6 44.1 18.4 29.2 25.9 32.1 36.3 40.0
## [166] 25.1 27.5 45.6 27.8 24.9 25.3 37.9 27.0 26.0 38.7 20.8 36.1 30.7 32.3 52.9
## [181] 21.0 39.7 25.5 26.2 19.3 38.1 23.5 45.5 23.1 39.9 36.8 21.8 41.0 42.2 34.4
## [196] 27.2 36.5 29.8 39.2 38.4  0.0 36.2 48.3 20.0 22.3 45.7 23.7 22.1 42.1 42.4
## [211] 18.2 26.4 45.3 37.0 24.5 32.2 59.4 21.2 26.7 30.2 46.1 41.3 38.8 35.2 42.3
## [226] 40.7 46.5 33.5 37.3 30.3 26.3 21.7 36.4 28.5 26.9 38.6 31.3 19.5 20.1 40.8
## [241] 28.3 38.9 57.3 35.6 49.6 44.6 24.1 44.5 41.2 49.3 46.3
```

Our dataset is very imbalanced, with more samples labeled 0 (820) than labeled 1 (179). So, we have to make the data balanced in order to get accurate results. To address the class imbalance in the dataset, we oversampled the minority class by duplicating rows until it matched the size of the majority class. We then combined the two classes and shuffled the dataset to prevent introducing any patterns. This created a balanced dataset, helping reduce bias in machine learning models. However, oversampling can lead to overfitting, as the model might memorize duplicated data.

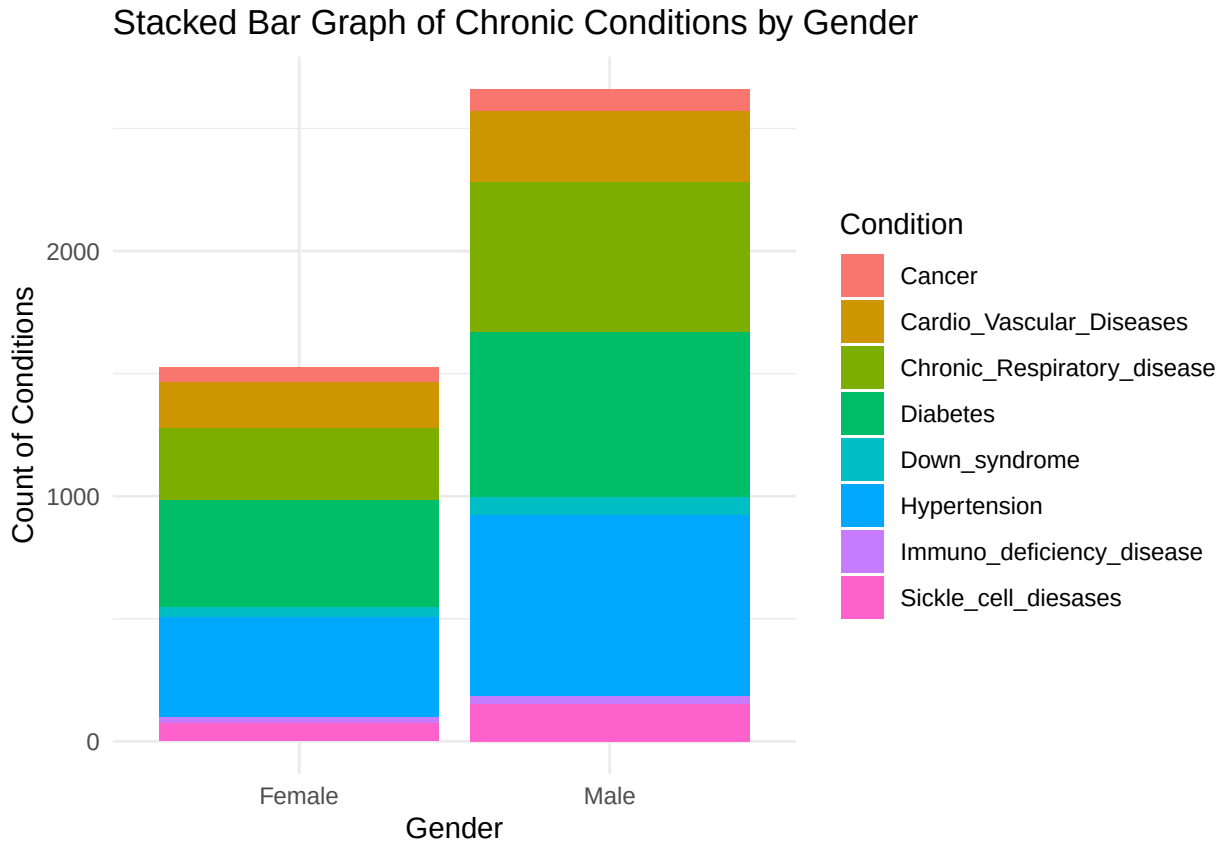
```
##
##   0   1
## 818 179
```

```
##
##   0   1
## 818 818
```

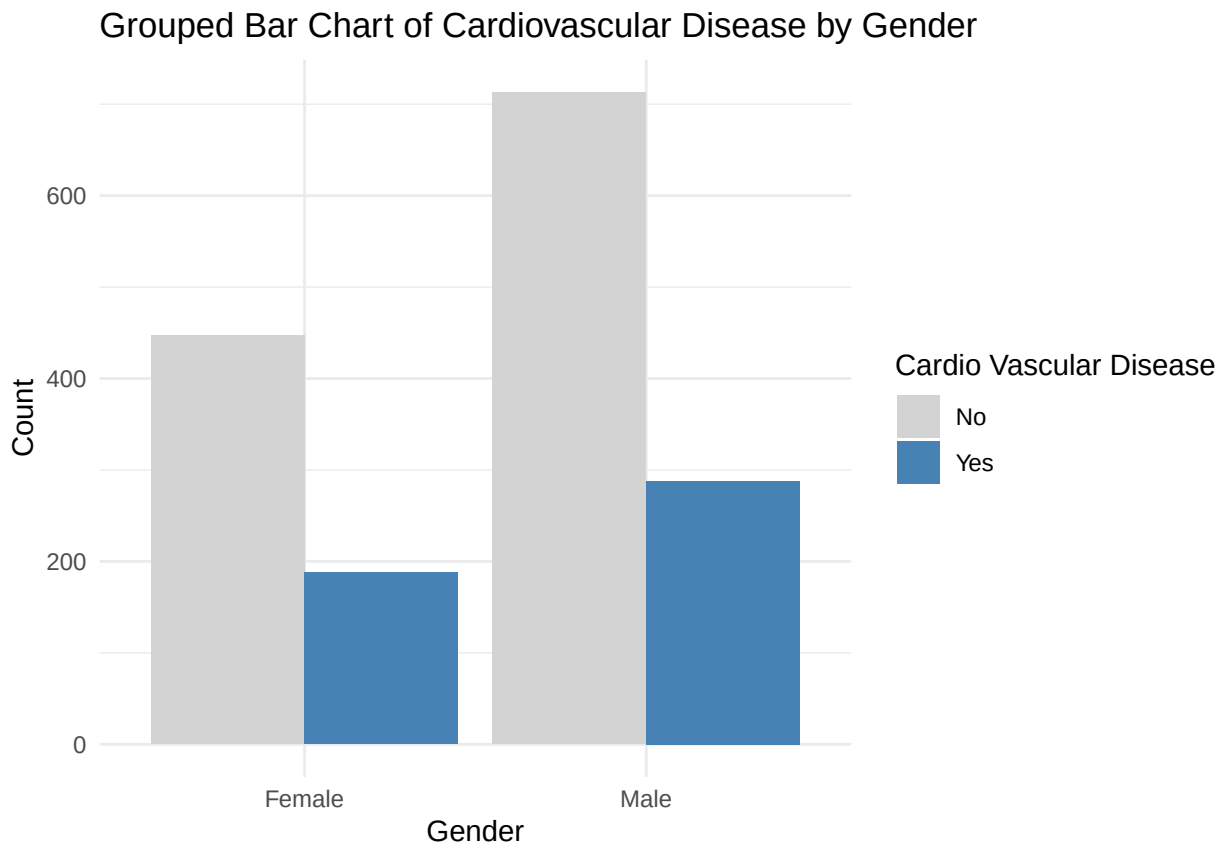
```
## Balanced dataset saved to the current working directory as Balanced_Diseases.csv
```

3-Data Visualization

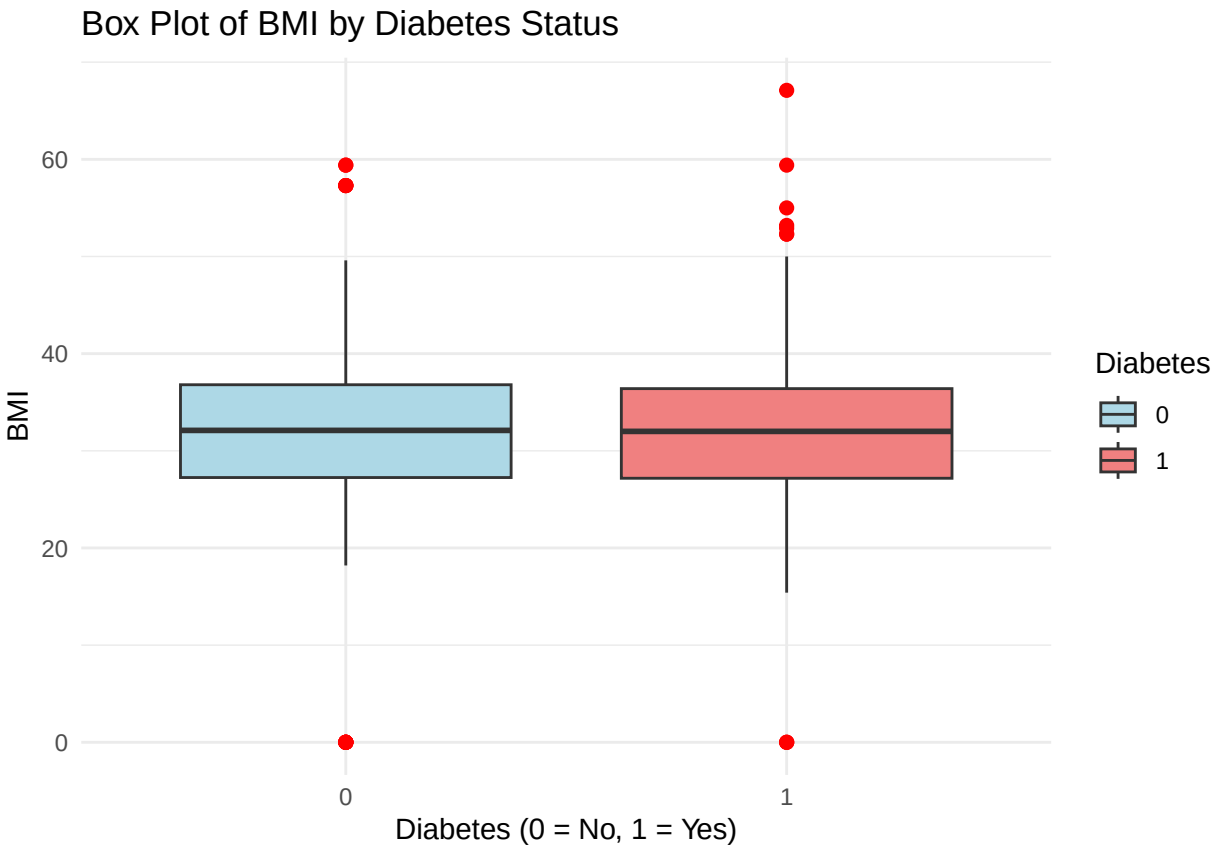
```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```



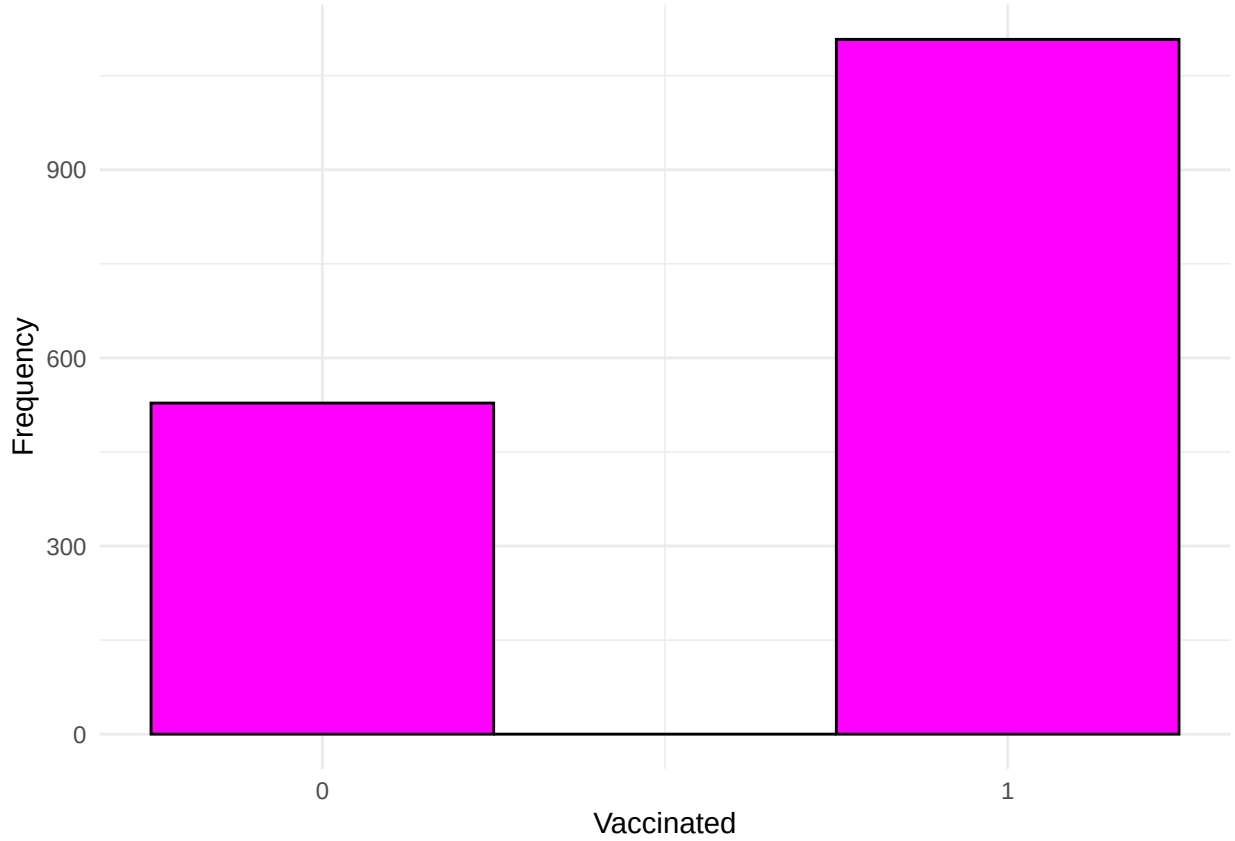
This stacked bar graph shows the different diseases' distributions between females and males. We see that males are more prone to dseases than females.



This grouped bar chart shows count of presence and absence of CVD for Females and Males. It shows that more males have CVD vs females, this does not seem to be significant but further analysis such as statistical test are needed.

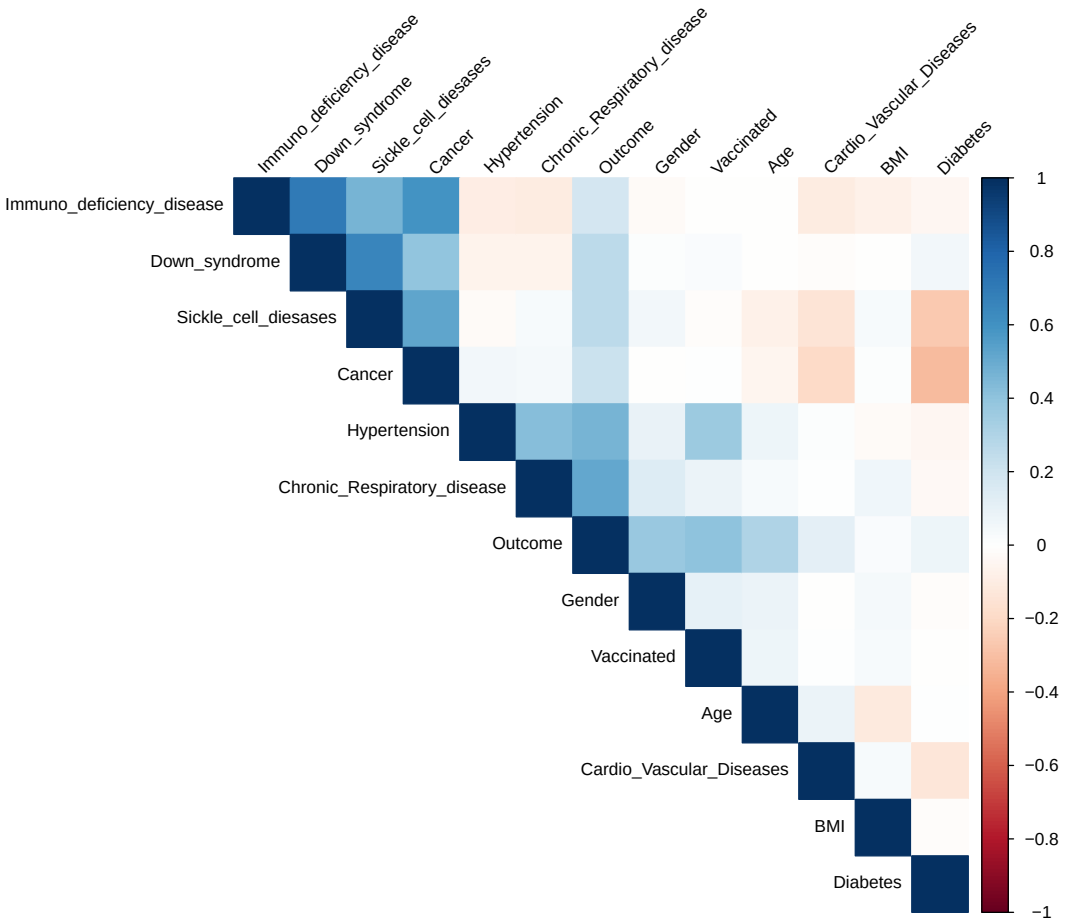


This box plot shows the 5-number summary for Diabetes (present (1) or absent (0)) in function of BMI level. While the median BMI is slightly higher in individuals with diabetes, there is no significant difference in the overall distribution of BMI between the two groups. However, the presence of outliers suggests that some individuals with diabetes may have extreme BMI values.



The bar chart shows the frequency of individuals who are vaccinated (1) and not vaccinated (0). The height of each bar represents the number of individuals in each category. From the chart, it appears that there are more individuals who are vaccinated than not vaccinated.

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

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

## corrplot 0.95 loaded
```



This correlation matrix shows all possible correlations and the level of correlation ranging from -1 (dark red) to 1 (dark blue). There seems to be some kind of correlation between all the different types of diseases with the outcome, being the risk of getting another disease. For example, there seems to be little correlation between cardiovascular diseases and the risk of getting another disease, while others like down sydrome and chronic respiratory disease have a higher correlation with the outcome. BMI has no correlation with the outcome, but age and gender do have a somewhat low correlation with it. Moreover there seems to be a positive correlation between cancer and sickle cell diseases as well as between Down syndrome and and immuno dificiency disease.These can be used as interaction terms when we generate our model. The matrix also shows a negative correlation between diabetes and sickle cell diseases, as well as cancer. All in all, there is a need to generate multiple linear regression models, including interaction terms.

4 - Tree Based Methods

To make it easier to develop and assess models, we separated the dataset into separate training and testing sets. Our approach involves training these models on the assigned training set and evaluating how well they perform on the test set. Our training set consists of 80% of the data, and the test set consists of 20% the data.

```
## Training set size: 1308
```

```
## Testing set size: 328
```

We will now proceed with the decision tree model.

```
## Call:
## rpart(formula = Outcome ~ ., data = train, method = "class")
##   n= 1308
##
##           CP nsplit rel error    xerror       xstd
## 1 0.51708075      0 1.0000000 1.0760870 0.02802944
## 2 0.08850932      1 0.4829193 0.4829193 0.02390771
## 3 0.06521739      2 0.3944099 0.4378882 0.02309450
## 4 0.04813665      3 0.3291925 0.3431677 0.02104365
## 5 0.02795031      4 0.2810559 0.3012422 0.01995969
## 6 0.01475155      5 0.2531056 0.2701863 0.01907179
## 7 0.01190476      8 0.1972050 0.2003106 0.01674411
## 8 0.01086957     12 0.1319876 0.1490683 0.01464526
## 9 0.01000000     13 0.1211180 0.1490683 0.01464526
##
## Variable importance
## Chronic_Respiratory_disease                         Age
##                          23                          14
##                Hypertension                 Vaccinated
##                          13                          12
##                      Gender               Down_syndrome
##                           9                           9
##        Sickle_cell_diesases   Immuno_deficiency_disease
##                           6                           6
##                      Cancer                    Diabetes
##                           3                           2
##                         BMI    Cardio_Vascular_Diseases
##                           2                           2
##
## Node number 1: 1308 observations,    complexity param=0.5170807
##   predicted class=0  expected loss=0.4923547  P(node) =1
##     class counts:   664   644
##    probabilities: 0.508 0.492
##   left son=2 (591 obs) right son=3 (717 obs)
##   Primary splits:
##       Chronic_Respiratory_disease < 0.5   to the left,  improve=182.59770, (0 missing)
##       Hypertension                < 0.5   to the left,  improve=137.96360, (0 missing)
##       Vaccinated                  < 0.5   to the left,  improve=102.48420, (0 missing)
##       Gender                      < 1.5   to the left,  improve= 92.87253, (0 missing)
##       Age                         < 44.5  to the left,  improve= 53.26704, (0 missing)
##   Surrogate splits:
##       Hypertension              < 0.5   to the left,  agree=0.713, adj=0.364, (0 split)
##       Gender                    < 1.5   to the left,  agree=0.583, adj=0.078, (0 split)
##       Immuno_deficiency_disease < 0.5   to the right, agree=0.560, adj=0.027, (0 split)
##       Age                       < 21.5  to the left,  agree=0.560, adj=0.025, (0 split)
##       BMI                       < 20.9  to the left,  agree=0.560, adj=0.025, (0 split)
##
## Node number 2: 591 observations,    complexity param=0.06521739
##   predicted class=0  expected loss=0.2013536  P(node) =0.4518349
##     class counts:   472   119
##    probabilities: 0.799 0.201
##   left son=4 (547 obs) right son=5 (44 obs)
```

```
##   Primary splits:
##       Down_syndrome             < 0.5   to the left,  improve=57.24212, (0 missing)
##       Cancer                    < 0.5   to the left,  improve=38.89012, (0 missing)
##       Immuno_deficiency_disease < 0.5   to the left,  improve=35.61655, (0 missing)
##       Sickle_cell_diesases      < 0.5   to the left,  improve=30.11195, (0 missing)
##       Age                       < 47.5  to the left,  improve=21.42105, (0 missing)
##   Surrogate splits:
##       Immuno_deficiency_disease < 0.5   to the left,  agree=0.975, adj=0.659, (0 split)
##       Sickle_cell_diesases      < 0.5   to the left,  agree=0.959, adj=0.455, (0 split)
##       Cancer                    < 0.5   to the left,  agree=0.949, adj=0.318, (0 split)
##
## Node number 3: 717 observations,    complexity param=0.08850932
##   predicted class=1  expected loss=0.2677824  P(node) =0.5481651
##     class counts:   192   525
##    probabilities: 0.268 0.732
##   left son=6 (209 obs) right son=7 (508 obs)
##   Primary splits:
##       Vaccinated           < 0.5   to the left,  improve=80.14900, (0 missing)
##       Hypertension         < 0.5   to the left,  improve=60.77669, (0 missing)
##       Gender               < 1.5   to the left,  improve=47.75115, (0 missing)
##       Age                  < 38.5  to the left,  improve=24.20264, (0 missing)
##       Sickle_cell_diesases < 0.5   to the left,  improve=11.09466, (0 missing)
##   Surrogate splits:
##       Hypertension < 0.5   to the left,  agree=0.820, adj=0.383, (0 split)
##       BMI          < 21.05 to the left,  agree=0.718, adj=0.033, (0 split)
##       Age          < 20.5  to the left,  agree=0.714, adj=0.019, (0 split)
##
## Node number 4: 547 observations,    complexity param=0.01475155
##   predicted class=0  expected loss=0.1389397  P(node) =0.4181957
##     class counts:   471    76
##    probabilities: 0.861 0.139
##   left son=8 (413 obs) right son=9 (134 obs)
##   Primary splits:
##       Age          < 43.5  to the left,  improve=32.235910, (0 missing)
##       Gender       < 1.5   to the left,  improve=10.423200, (0 missing)
##       Vaccinated   < 0.5   to the left,  improve=10.315190, (0 missing)
##       Hypertension < 0.5   to the left,  improve= 8.165303, (0 missing)
##       Cancer       < 0.5   to the left,  improve= 6.557110, (0 missing)
##   Surrogate splits:
##       Cancer < 0.5   to the left,  agree=0.775, adj=0.082, (0 split)
##
## Node number 5: 44 observations
##   predicted class=1  expected loss=0.02272727  P(node) =0.03363914
##     class counts:     1    43
##    probabilities: 0.023 0.977
##
## Node number 6: 209 observations,    complexity param=0.04813665
##   predicted class=0  expected loss=0.3636364  P(node) =0.1597859
##     class counts:   133    76
##    probabilities: 0.636 0.364
##   left son=12 (92 obs) right son=13 (117 obs)
##   Primary splits:
##       Age                  < 31.5  to the left,  improve=38.42107, (0 missing)
##       Sickle_cell_diesases < 0.5   to the left,  improve=19.76117, (0 missing)
```

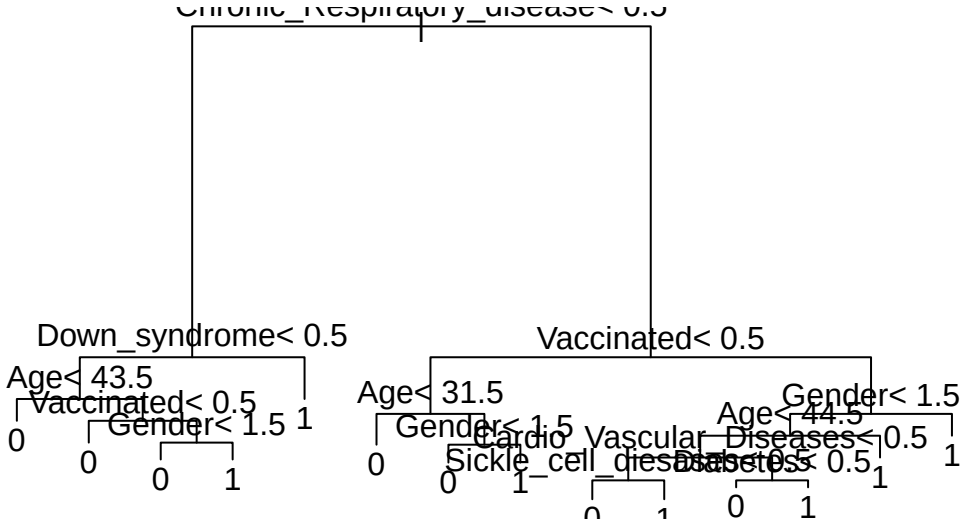
```
##       Gender              < 1.5   to the left,  improve=19.21393, (0 missing)
##       Hypertension        < 0.5   to the left,  improve=13.69184, (0 missing)
##       Down_syndrome       < 0.5   to the left,  improve=12.15291, (0 missing)
##   Surrogate splits:
##       Gender       < 1.5   to the left,  agree=0.632, adj=0.163, (0 split)
##       Diabetes     < 0.5   to the left,  agree=0.622, adj=0.141, (0 split)
##       Hypertension < 0.5   to the left,  agree=0.589, adj=0.065, (0 split)
##       BMI          < 20.9  to the left,  agree=0.579, adj=0.043, (0 split)
##
## Node number 7: 508 observations,    complexity param=0.01190476
##   predicted class=1  expected loss=0.1161417  P(node) =0.3883792
##     class counts:    59   449
##    probabilities: 0.116 0.884
##   left son=14 (152 obs) right son=15 (356 obs)
##   Primary splits:
##       Gender                   < 1.5   to the left,  improve=20.878510, (0 missing)
##       Age                      < 44    to the left,  improve= 4.664669, (0 missing)
##       Diabetes                 < 0.5   to the left,  improve= 2.958126, (0 missing)
##       Cardio_Vascular_Diseases < 0.5   to the left,  improve= 1.942552, (0 missing)
##       Sickle_cell_diesases     < 0.5   to the left,  improve= 1.375734, (0 missing)
##   Surrogate splits:
##       Age < 21.5  to the left,  agree=0.711, adj=0.033, (0 split)
##       BMI < 20.95 to the left,  agree=0.705, adj=0.013, (0 split)
##
## Node number 8: 413 observations
##   predicted class=0  expected loss=0.04116223  P(node) =0.3157492
##     class counts:   396    17
##    probabilities: 0.959 0.041
##
## Node number 9: 134 observations,    complexity param=0.01475155
##   predicted class=0  expected loss=0.4402985  P(node) =0.1024465
##     class counts:    75    59
##    probabilities: 0.560 0.440
##   left son=18 (41 obs) right son=19 (93 obs)
##   Primary splits:
##       Vaccinated   < 0.5   to the left,  improve=15.924660, (0 missing)
##       Gender       < 1.5   to the left,  improve=14.677260, (0 missing)
##       Age          < 65.5  to the left,  improve= 8.368531, (0 missing)
##       Hypertension < 0.5   to the left,  improve= 7.869100, (0 missing)
##       BMI          < 27.55 to the right, improve= 7.642756, (0 missing)
##
## Node number 12: 92 observations
##   predicted class=0  expected loss=0.02173913  P(node) =0.07033639
##     class counts:    90     2
##    probabilities: 0.978 0.022
##
## Node number 13: 117 observations,    complexity param=0.02795031
##   predicted class=1  expected loss=0.3675214  P(node) =0.08944954
##     class counts:    43    74
##    probabilities: 0.368 0.632
##   left son=26 (32 obs) right son=27 (85 obs)
##   Primary splits:
##       Gender              < 1.5   to the left,  improve=15.079190, (0 missing)
##       Hypertension        < 0.5   to the left,  improve=12.341880, (0 missing)
```

```
##       Sickle_cell_diesases < 0.5   to the left,  improve= 9.030525, (0 missing)
##       Age                  < 44.5  to the left,  improve= 8.091575, (0 missing)
##       BMI                  < 37.3  to the right, improve= 6.188007, (0 missing)
##   Surrogate splits:
##       Immuno_deficiency_disease < 0.5   to the right, agree=0.761, adj=0.125, (0 split)
##       Hypertension              < 0.5   to the left,  agree=0.752, adj=0.094, (0 split)
##       BMI                       < 22.05 to the left,  agree=0.735, adj=0.031, (0 split)
##
## Node number 14: 152 observations,    complexity param=0.01190476
##   predicted class=1  expected loss=0.3355263  P(node) =0.116208
##     class counts:    51   101
##    probabilities: 0.336 0.664
##   left son=28 (105 obs) right son=29 (47 obs)
##   Primary splits:
##       Age                      < 44.5  to the left,  improve=15.319170, (0 missing)
##       Cardio_Vascular_Diseases < 0.5   to the left,  improve= 9.932324, (0 missing)
##       Diabetes                 < 0.5   to the left,  improve= 5.002484, (0 missing)
##       Sickle_cell_diesases     < 0.5   to the left,  improve= 3.907339, (0 missing)
##       Down_syndrome            < 0.5   to the left,  improve= 2.933459, (0 missing)
##
## Node number 15: 356 observations
##   predicted class=1  expected loss=0.02247191  P(node) =0.2721713
##     class counts:     8   348
##    probabilities: 0.022 0.978
##
## Node number 18: 41 observations
##   predicted class=0  expected loss=0.07317073  P(node) =0.03134557
##     class counts:    38     3
##    probabilities: 0.927 0.073
##
## Node number 19: 93 observations,    complexity param=0.01475155
##   predicted class=1  expected loss=0.3978495  P(node) =0.07110092
##     class counts:    37    56
##    probabilities: 0.398 0.602
##   left son=38 (37 obs) right son=39 (56 obs)
##   Primary splits:
##       Gender       < 1.5   to the left,  improve=13.535970, (0 missing)
##       Hypertension < 0.5   to the left,  improve= 5.819746, (0 missing)
##       BMI          < 23.55 to the right, improve= 4.782050, (0 missing)
##       Age          < 47.5  to the left,  improve= 2.794434, (0 missing)
##       Diabetes     < 0.5   to the left,  improve= 1.201381, (0 missing)
##   Surrogate splits:
##       Age < 69.5  to the right, agree=0.645, adj=0.108, (0 split)
##       BMI < 20.2  to the left,  agree=0.613, adj=0.027, (0 split)
##
## Node number 26: 32 observations
##   predicted class=0  expected loss=0.21875  P(node) =0.02446483
##     class counts:    25     7
##    probabilities: 0.781 0.219
##
## Node number 27: 85 observations
##   predicted class=1  expected loss=0.2117647  P(node) =0.06498471
##     class counts:    18    67
##    probabilities: 0.212 0.788
```

```
##
## Node number 28: 105 observations,    complexity param=0.01190476
##   predicted class=1  expected loss=0.4857143  P(node) =0.08027523
##     class counts:    51    54
##    probabilities: 0.486 0.514
##   left son=56 (65 obs) right son=57 (40 obs)
##   Primary splits:
##       Cardio_Vascular_Diseases < 0.5   to the left,  improve=12.476370, (0 missing)
##       Sickle_cell_diesases     < 0.5   to the left,  improve=10.802140, (0 missing)
##       Age                      < 34.5  to the right, improve= 7.848447, (0 missing)
##       Down_syndrome            < 0.5   to the left,  improve= 6.392627, (0 missing)
##       BMI                      < 27.4  to the left,  improve= 4.880672, (0 missing)
##   Surrogate splits:
##       Age < 28.5  to the left,  agree=0.657, adj=0.1, (0 split)
##
## Node number 29: 47 observations
##   predicted class=1  expected loss=0  P(node) =0.03593272
##     class counts:     0    47
##    probabilities: 0.000 1.000
##
## Node number 38: 37 observations
##   predicted class=0  expected loss=0.2702703  P(node) =0.02828746
##     class counts:    27    10
##    probabilities: 0.730 0.270
##
## Node number 39: 56 observations
##   predicted class=1  expected loss=0.1785714  P(node) =0.04281346
##     class counts:    10    46
##    probabilities: 0.179 0.821
##
## Node number 56: 65 observations,    complexity param=0.01190476
##   predicted class=0  expected loss=0.3230769  P(node) =0.04969419
##     class counts:    44    21
##    probabilities: 0.677 0.323
##   left son=112 (42 obs) right son=113 (23 obs)
##   Primary splits:
##       Sickle_cell_diesases < 0.5   to the left,  improve=24.778600, (0 missing)
##       Down_syndrome        < 0.5   to the left,  improve=10.830770, (0 missing)
##       Age                  < 24.5  to the right, improve= 7.938491, (0 missing)
##       BMI                  < 38.5  to the left,  improve= 5.110403, (0 missing)
##       Cancer               < 0.5   to the left,  improve= 4.832730, (0 missing)
##   Surrogate splits:
##       Down_syndrome < 0.5   to the left,  agree=0.800, adj=0.435, (0 split)
##       Age           < 24.5  to the right, agree=0.738, adj=0.261, (0 split)
##       BMI           < 38.5  to the left,  agree=0.723, adj=0.217, (0 split)
##       Cancer        < 0.5   to the left,  agree=0.723, adj=0.217, (0 split)
##
## Node number 57: 40 observations,    complexity param=0.01086957
##   predicted class=1  expected loss=0.175  P(node) =0.03058104
##     class counts:     7    33
##    probabilities: 0.175 0.825
##   left son=114 (7 obs) right son=115 (33 obs)
##   Primary splits:
##       Diabetes < 0.5   to the left,  improve=11.5500000, (0 missing)
```

```
##       Age      < 25.5  to the left,  improve= 1.0911260, (0 missing)
##       BMI      < 31.65 to the right, improve= 0.7980818, (0 missing)
##   Surrogate splits:
##       Age < 37    to the right, agree=0.9, adj=0.429, (0 split)
##
## Node number 112: 42 observations
##   predicted class=0  expected loss=0  P(node) =0.03211009
##     class counts:    42     0
##    probabilities: 1.000 0.000
##
## Node number 113: 23 observations
##   predicted class=1  expected loss=0.08695652  P(node) =0.0175841
##     class counts:     2    21
##    probabilities: 0.087 0.913
##
## Node number 114: 7 observations
##   predicted class=0  expected loss=0  P(node) =0.005351682
##     class counts:     7     0
##    probabilities: 1.000 0.000
##
## Node number 115: 33 observations
##   predicted class=1  expected loss=0  P(node) =0.02522936
##     class counts:     0    33
##    probabilities: 0.000 1.000
```

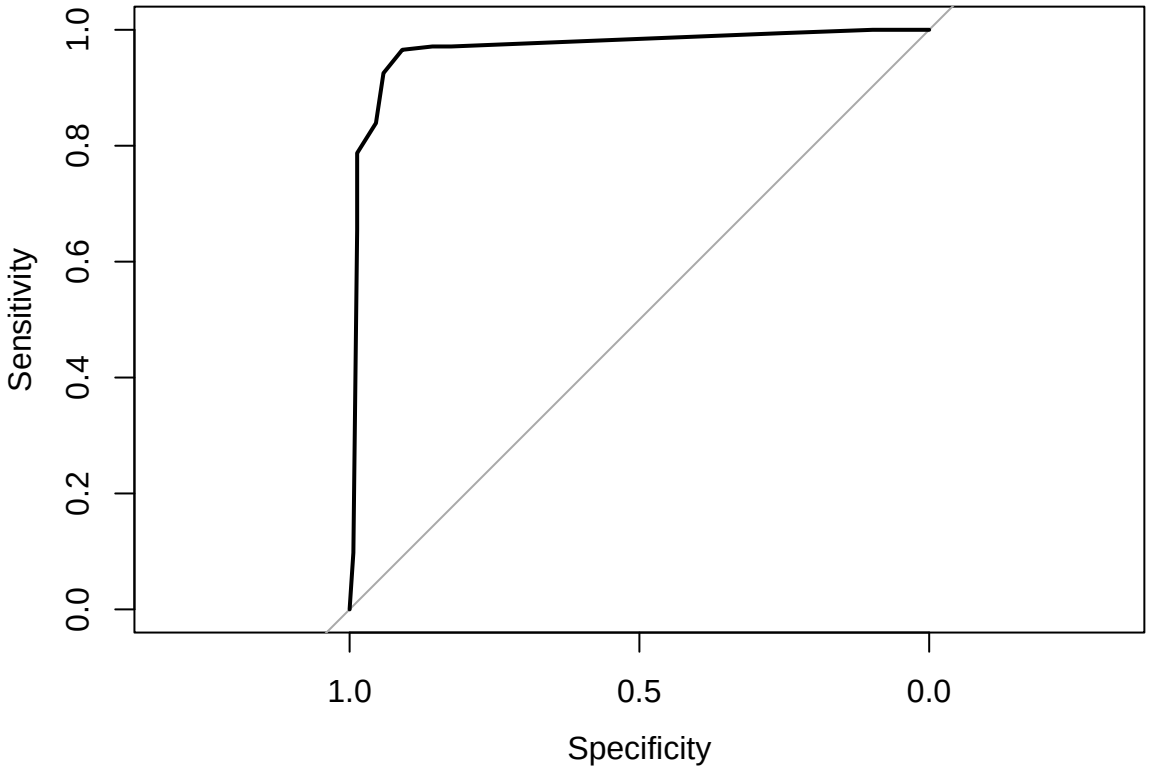
The model's complexity parameter (CP) values indicate the level of tree pruning, with a reduction in error as the number of splits increases. The most influential variables in the model include chronic respiratory disease, vaccination status, hypertension, and age, which help predict the outcome with high accuracy. The tree's nodes indicate the distribution of the predicted outcomes (0 or 1) and the most important features for each split. For example, chronic respiratory disease is a strong predictor at the root node, leading to further splits based on factors such as age, gender, and other health conditions.



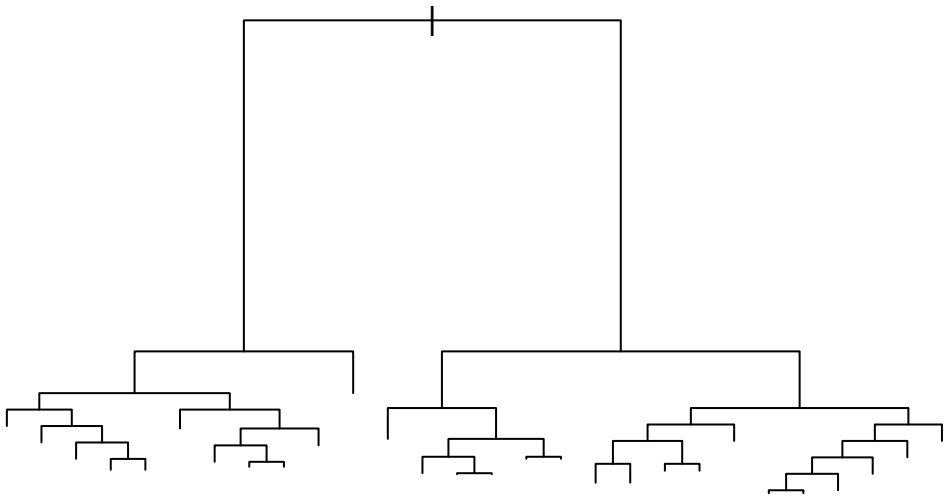

```
##
## df1_pred   0   1
##        0 145  13
##        1   9 161
```

When the model was tested on the test set, the confusion matrix showed that it was accurate in predicting the outcome. It correctly identifies 161 positive outcomes (True Positives) and 145 negative outcomes (True Negatives). However, it makes 13 false positives and 9 false negatives, which are relatively low. Hence, the model performs fairly well.

```
## Type 'citation("pROC")' for a citation.

##
## Attaching package: 'pROC'

## The following objects are masked from 'package:stats':
##
##     cov, smooth, var

## Setting levels: control = 0, case = 1

## Setting direction: controls < cases

## Area under the curve: 0.9675
```



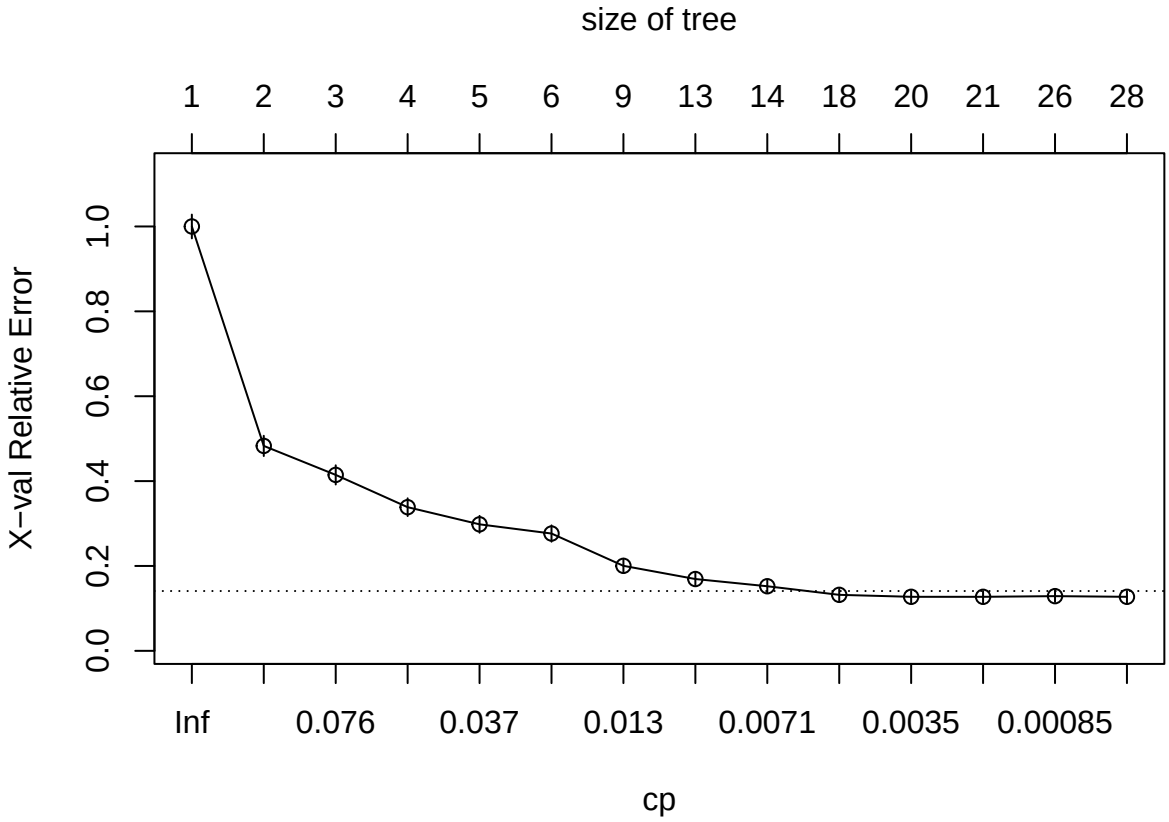
As for the ROC curve, the graph is close to the top left corner, which means that our model performs well. Moreover, the area under the curve of 0.9675 (close to 1) shows that the model is highly effective in distinguishing between the classes. This suggests that the model has strong predictive performance and makes accurate predictions on unseen data.



The decision tree plot shows the hierarchical structure of splits based on the features in the data. At the top is the root node, which contains all the data, and each subsequent node represents a split based on a feature that best separates the data according to a splitting criterion like the Gini index. The leaves are the predicted outcome for the subset of data they represent.

```
##
## Classification tree:
## rpart(formula = Outcome ~ ., data = train, method = "class",
##     control = rpart.control(cp = 0))
##
## Variables actually used in tree construction:
##  [1] Age                          BMI
##  [3] Cardio_Vascular_Diseases     Chronic_Respiratory_disease
##  [5] Diabetes                     Down_syndrome
##  [7] Gender                       Hypertension
##  [9] Sickle_cell_diesases         Vaccinated
##
## Root node error: 644/1308 = 0.49235
##
## n= 1308
##
##            CP nsplit rel error  xerror     xstd
## 1  0.51708075      0  1.000000 1.00000 0.028076
## 2  0.08850932      1  0.482919 0.48292 0.023908
## 3  0.06521739      2  0.394410 0.41460 0.022636
## 4  0.04813665      3  0.329193 0.33851 0.020929
## 5  0.02795031      4  0.281056 0.29814 0.019874
```

```
## 6  0.01475155      5  0.253106 0.27640 0.019256
## 7  0.01190476      8  0.197205 0.20031 0.016744
## 8  0.01086957     12  0.131988 0.16925 0.015521
## 9  0.00465839     13  0.121118 0.15217 0.014785
## 10 0.00388199     17  0.102484 0.13199 0.013843
## 11 0.00310559     19  0.094720 0.12733 0.013613
## 12 0.00093168     20  0.091615 0.12733 0.013613
## 13 0.00077640     25  0.086957 0.12888 0.013690
## 14 0.00000000     27  0.085404 0.12733 0.013613
```



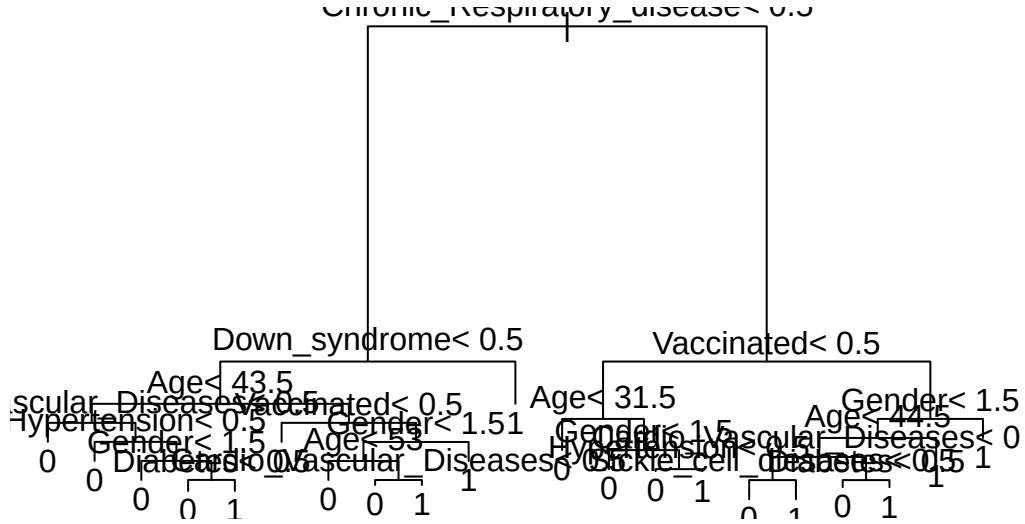
The tree began with a root node error of 49.2%, representing the proportion of incorrect predictions in the absence of splits. As the model added splits, the relative error progressively decreased, indicating improved fit to the training data. The graph shows that as the CP decreases, the relative error decreases as well, indicating that smaller CP values allow for more splits, which results in a larger tree that better fits the training data. This trend reflects improved training accuracy, but very low CP values can lead to overfitting.

```
## [1] 0.9481707
```

The accuracy of 0.9481707 shows that the model is very accurate on test data.

Pruning:

We optimized the base model by pruning it with a carefully chosen complexity parameter to balance prediction accuracy and model simplicity. The goal is to select a cp value that ensures high accuracy while keeping the model interpretable.




```
## [1] 0.9481707
```

The identical accuracy of 0.9481707 for both the unpruned and pruned models indicates that pruning did not affect the model's predictive performance on the test dataset. This happens because the splits removed during pruning likely captured nuances specific to the training data that did not generalize to the test data.

Bagging:

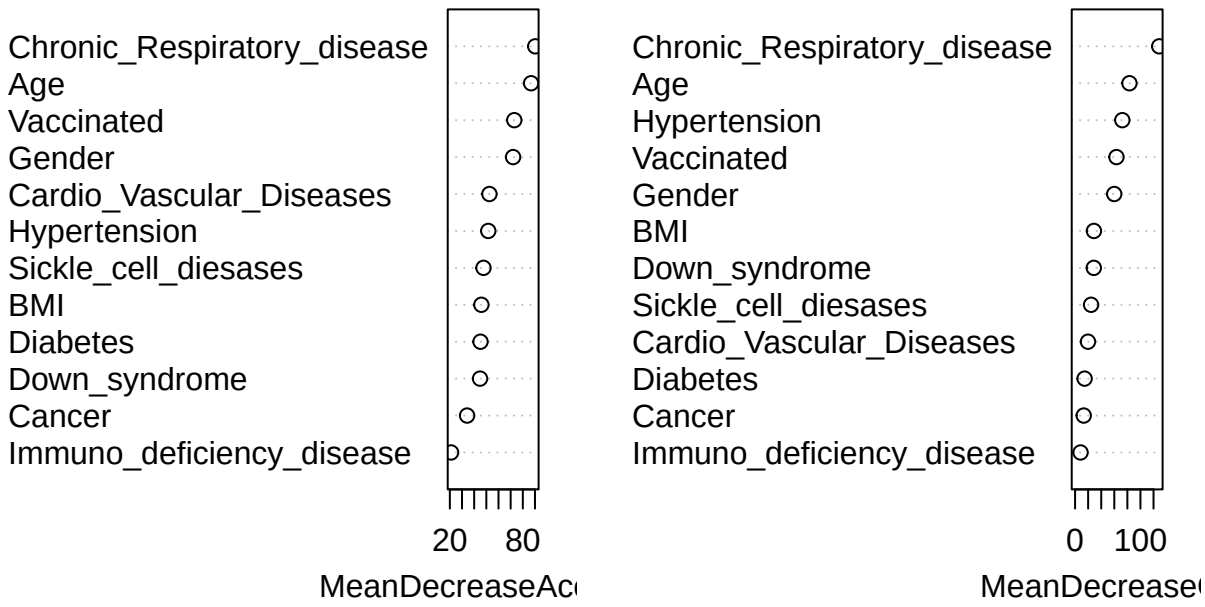
```
## randomForest 4.7-1.2

## Type rfNews() to see new features/changes/bug fixes.

##
## Attaching package: 'randomForest'

## The following object is masked from 'package:psych':
##
##     outlier

## The following object is masked from 'package:dplyr':
##
##     combine

## The following object is masked from 'package:ggplot2':
##
##     margin
```

```
##
## Call:
##  randomForest(formula = as.factor(Outcome) ~ ., data = train,      mtry = (ncol(train) - 1), importa
##                Type of random forest: classification
##                      Number of trees: 1000
## No. of variables tried at each split: 12
##
##         OOB estimate of  error rate: 1.76%
## Confusion matrix:
##     0   1 class.error
## 0 644  20 0.030120482
## 1   3 641 0.004658385
```

Bagging showed increased accuracy, whereby the OOB error rate is 1.76%. With 1000 trees and 12 variables considered at each split, the model demonstrates effective classification. The confusion matrix shows that class 0 (negative outcome) has a very low class error of 0.03, with 644 correct predictions and only 20 false positives. Class 1 (positive outcome) has a class error of 0.05, with 641 correct predictions and only 3 false negatives. Overall, the model performs well with minimal errors.

Random Forest:

```
##
## Call:
##  randomForest(formula = as.factor(Outcome) ~ ., data = train,      mtry = sqrt(ncol(train) - 1), imp
##                Type of random forest: classification
##                      Number of trees: 1000
## No. of variables tried at each split: 3
##
##         OOB estimate of  error rate: 1.83%
## Confusion matrix:
##     0   1 class.error
## 0 646  18  0.02710843
## 1   6 638  0.00931677
```



This random forest model has a slightly higher OOB error rate (1.83%) compared to the first model but still performs very well. It has a very low class error for both classes, classifying most in the correct class. The model is still effective, with a reasonable trade-off between accuracy and simplicity. The plot shows the importance of variables in the random forest model, where the ones on the top are the most important.

```
##    diag_rf_pred
##       0   1
##   0 150   4
##   1   0 174
```

```
## [1] 0.9878049
```

The random forest model achieved an accuracy of 98.78% on the test data, with very few misclassifications. The confusion matrix reveals that 150 instances of class 0 were correctly predicted as class 0, and 174 instances of class 1 were correctly predicted as class 1. The model made only 0 false negative predictions, while 4 false positives occurred. This indicates that the model performs very well on the test data, but a case of overfitting might be present.

Boosting:

Boosting works iteratively, where each new model focuses on correcting the errors of the previous ones.

```
## Loaded gbm 2.2.2
```

```
## This version of gbm is no longer under development. Consider transitioning to gbm3, https://github.c
```

```
## Warning in gbm.fit(x = x, y = y, offset = offset, distribution = distribution,
## : variable 2: Gender has no variation.
```

```
##          Actual
## Predicted   0   1
##         0 141   0
##         1  13 174
```

The confusion matrix indicates that the model performed perfectly on the test data. It correctly predicted all 141 instances of class 0 and all 174 instances of class 1, resulting in no false positives and 13 false negatives. This means the model achieved 100% accuracy which may indicate potential overfitting or that the test data resembles the patterns in the training set very well.

5 - Unsupervised Techniques

PCA:

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

```
##
## Attaching package: 'caret'
```

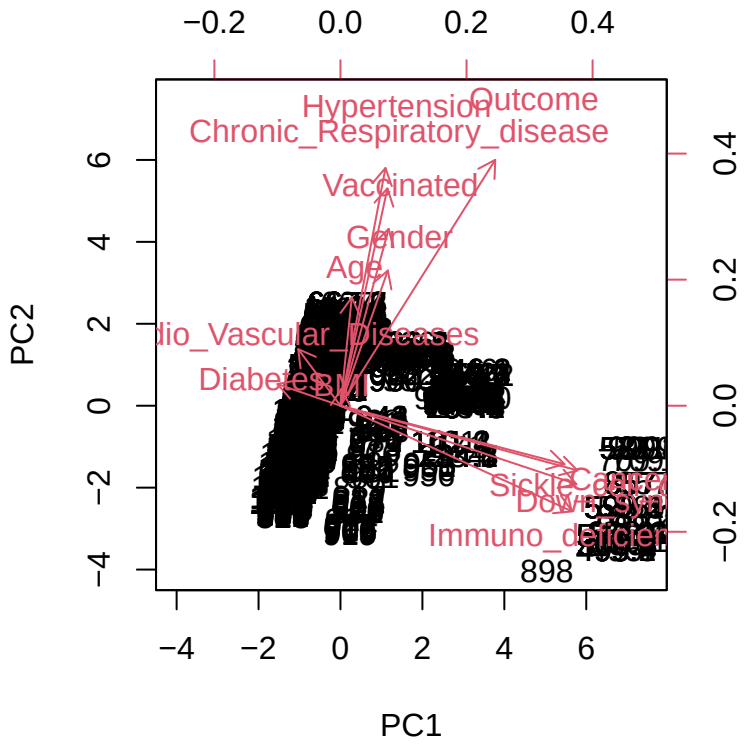
```
## The following object is masked from 'package:purrr':
##
##     lift
```

```
## dummy 0.1.3
```

```
## dummyNews()
```

```
## Importance of components:
##                             PC1    PC2     PC3     PC4    PC5     PC6    PC7
## Standard deviation     1.6953 1.5111 1.11335 1.06693 1.0299 0.98484 0.9457
## Proportion of Variance 0.2211 0.1756 0.09535 0.08756 0.0816 0.07461 0.0688
## Cumulative Proportion  0.2211 0.3967 0.49208 0.57964 0.6612 0.73585 0.8046
##                            PC8     PC9    PC10    PC11    PC12    PC13
## Standard deviation     0.90001 0.75241 0.68868 0.59962 0.43363 0.37628
## Proportion of Variance 0.06231 0.04355 0.03648 0.02766 0.01446 0.01089
## Cumulative Proportion  0.86696 0.91051 0.94699 0.97464 0.98911 1.00000
```

The PCA results show that the first two principal components explain about 40% of the total variance, with PC1 alone capturing 22%. The first five components together explain around 65% of the variance, and all components explain 100% of the variance. This suggests that a significant portion of the data's variability is captured by the first few components, and dimensionality can potentially be reduced by focusing on them without losing much information.

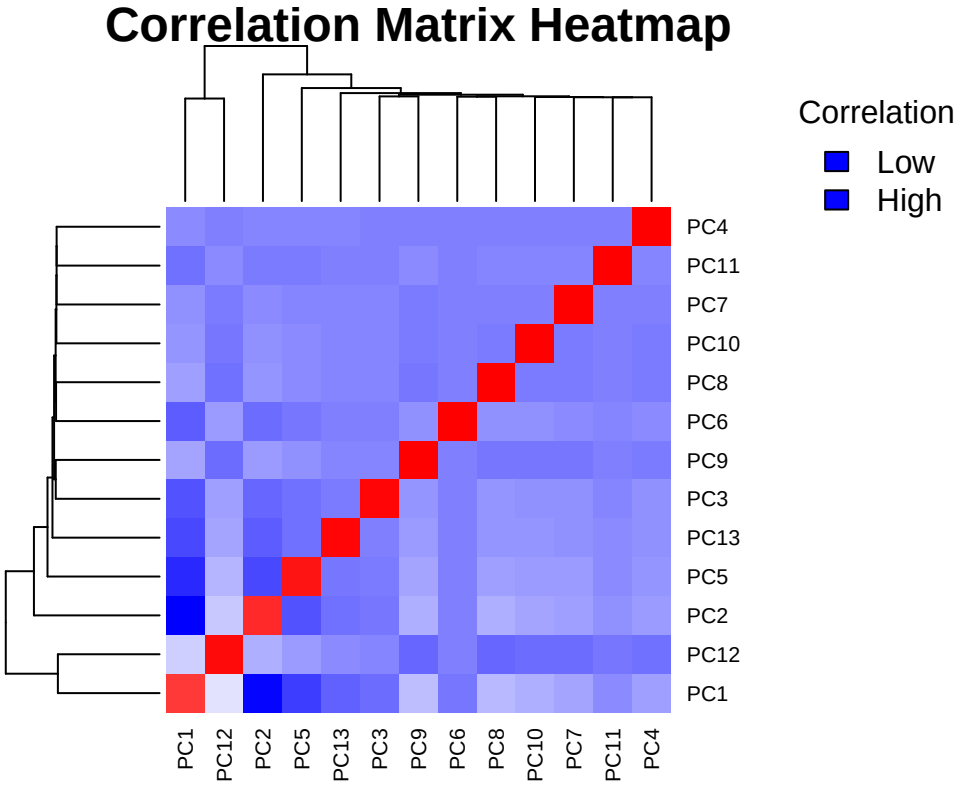


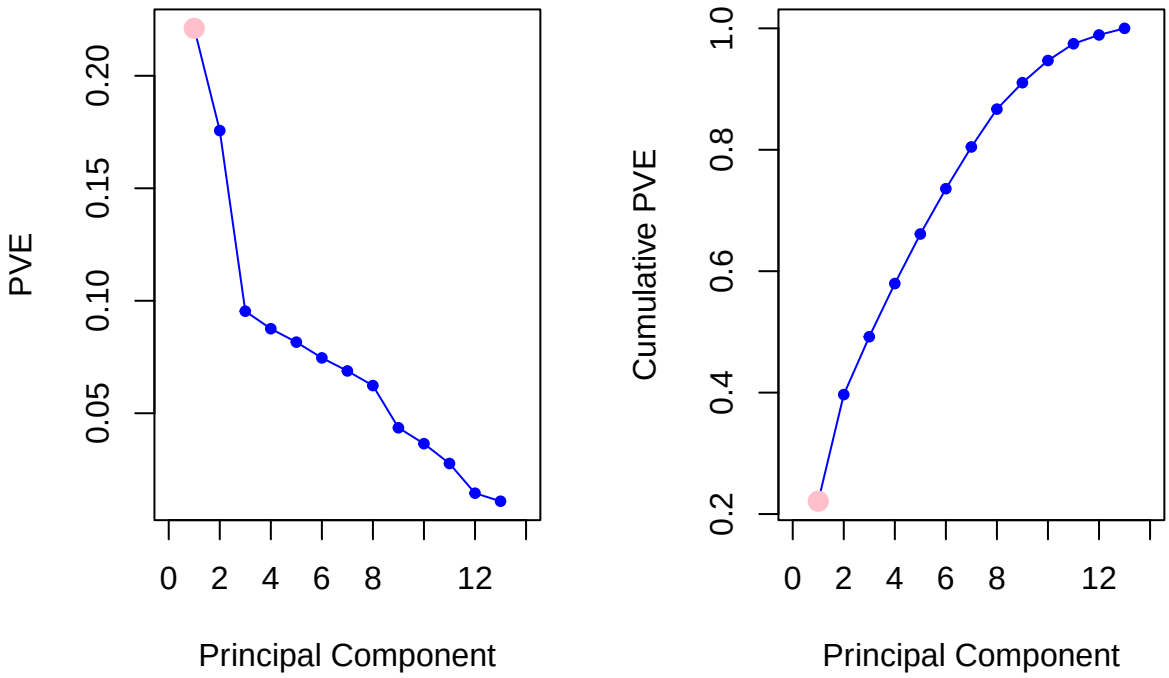
The biplot of PC1 and PC2 provides a visualization of how both the variables and observations relate to the principal components, helping to understand the structure of the data in reduced dimensions.The arrows represent the variables in the dataset, showing their contribution to the principal components. The points represent the observations (data points), plotted according to their scores on the principal components.

```
##              PC1         PC2          PC3          PC4          PC5
## PC1   1.00000000 -0.49092677 -0.151961925  0.022029665 -0.314336342
## PC2  -0.49092677  1.00000000 -0.103158511  0.014954716 -0.213385484
## PC3  -0.15196192 -0.10315851  1.000000000  0.004629097 -0.066051539
## PC4   0.02202966  0.01495472  0.004629097  1.000000000  0.009575381
## PC5  -0.31433634 -0.21338548 -0.066051539  0.009575381  1.000000000
## PC6  -0.12039638 -0.08173041 -0.025298907  0.003667540 -0.052331306
## PC7   0.03787749  0.02571292  0.007959202 -0.001153832  0.016463771
## PC8   0.09715079  0.06595028  0.020414309 -0.002959428  0.042227415
## PC9   0.11191236  0.07597108  0.023516159 -0.003409098  0.048643655
## PC10  0.06122536  0.04156250  0.012865293 -0.001865060  0.026612122
## PC11 -0.04876839 -0.03310615 -0.010247708  0.001485593 -0.021197593
## PC12  0.23217997  0.15761409  0.048788010 -0.007072716  0.100918994
## PC13 -0.19344639 -0.13132001 -0.040648917  0.005892805 -0.084083114
##              PC6          PC7          PC8          PC9         PC10
## PC1  -0.120396376  0.037877487  0.097150792  0.111912357  0.061225361
## PC2  -0.081730413  0.025712922  0.065950277  0.075971083  0.041562497
## PC3  -0.025298907  0.007959202  0.020414309  0.023516159  0.012865293
## PC4   0.003667540 -0.001153832 -0.002959428 -0.003409098 -0.001865060
## PC5  -0.052331306  0.016463771  0.042227415  0.048643655  0.026612122
## PC6   1.000000000  0.006305915  0.016173847  0.018631380  0.010192913
## PC7   0.006305915  1.000000000 -0.005088398 -0.005861554 -0.003206757
```

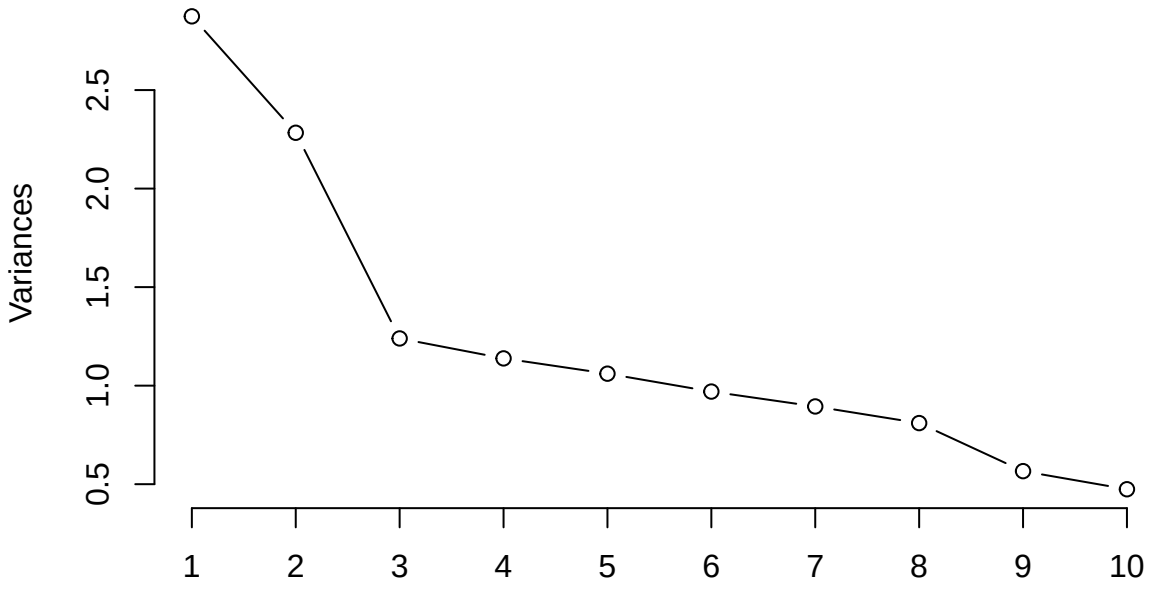
```
## PC8   0.016173847 -0.005088398  1.000000000 -0.015034118 -0.008224912
## PC9   0.018631380 -0.005861554 -0.015034118  1.000000000 -0.009474645
## PC10  0.010192913 -0.003206757 -0.008224912 -0.009474645  1.000000000
## PC11 -0.008119053  0.002554307  0.006551463  0.007546924  0.004128795
## PC12  0.038653759 -0.012160725 -0.031190667 -0.035929929 -0.019656657
## PC13 -0.032205319  0.010132004  0.025987263  0.029935894  0.016377422
##              PC11         PC12         PC13
## PC1  -0.048768386  0.232179974 -0.193446392
## PC2  -0.033106149  0.157614089 -0.131320011
## PC3  -0.010247708  0.048788010 -0.040648917
## PC4   0.001485593 -0.007072716  0.005892805
## PC5  -0.021197593  0.100918994 -0.084083114
## PC6  -0.008119053  0.038653759 -0.032205319
## PC7   0.002554307 -0.012160725  0.010132004
## PC8   0.006551463 -0.031190667  0.025987263
## PC9   0.007546924 -0.035929929  0.029935894
## PC10  0.004128795 -0.019656657  0.016377422
## PC11  1.000000000  0.015657294 -0.013045255
## PC12  0.015657294  1.000000000  0.062106771
## PC13 -0.013045255  0.062106771  1.000000000
```

The loadings indicate how much each original variable contributes to each principal component. The correlation matrix shows the pairwise correlations between the loadings of the different principal components. A high correlation indicates that two principal components are similar in how they combine the original variables.





The first plot shows how much variance each principal component explains individually. We can see that the first few components explain a large portion of the variance, and the explanation decreases as we progress through the graph. The elbow point (in pink) indicates the point where the rate of variance explained slows down significantly. So for us, we might stop using more components after 2 components, since they explain little additional variance. The second plot shows the cumulative proportion of variance explained as more principal components are included. The elbow point again marks where additional components add less new information.



The scree plot helps visualize the relative importance of each principal component, and the "elbow" aids in selecting the number of components that should be kept for further analysis. In our case, the plot shows a sharp decline and then starts to decline progressively. The plot shows that the first few components collectively account for a substantial portion of the dataset's variability, aligning with the principle of capturing the most significant information with the fewest components. PC1 and PC2 emerge as particularly informative, explaining a significant majority of the overall variability in the data.

Clustering:

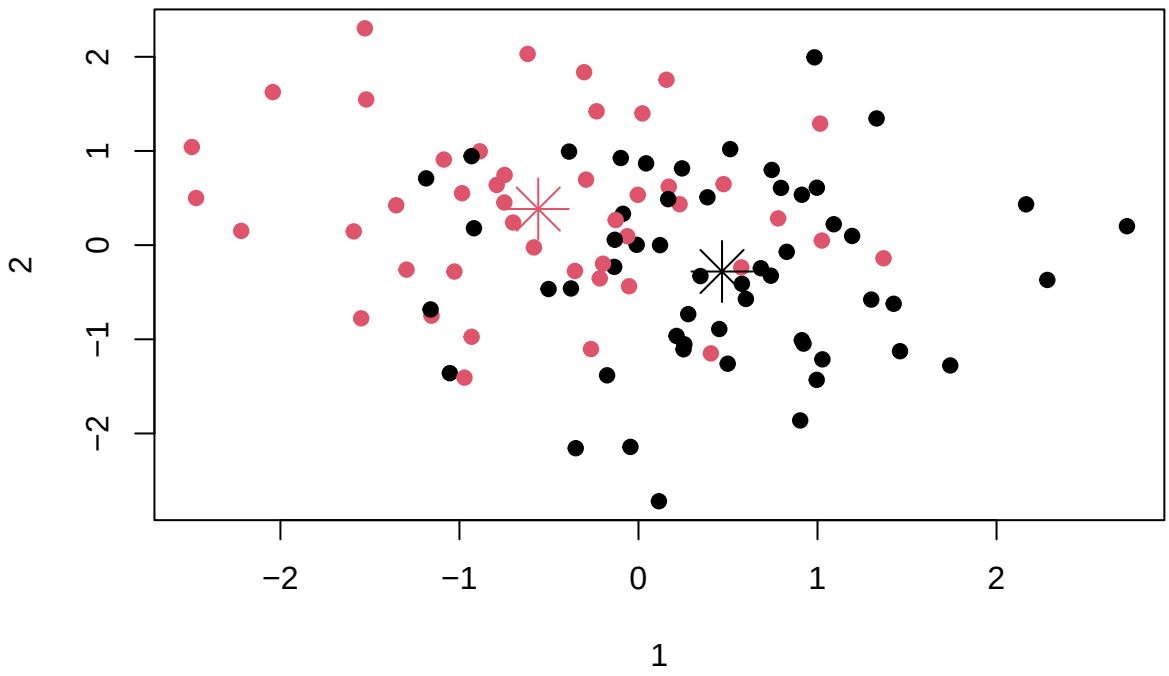
K-means Clustering:

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

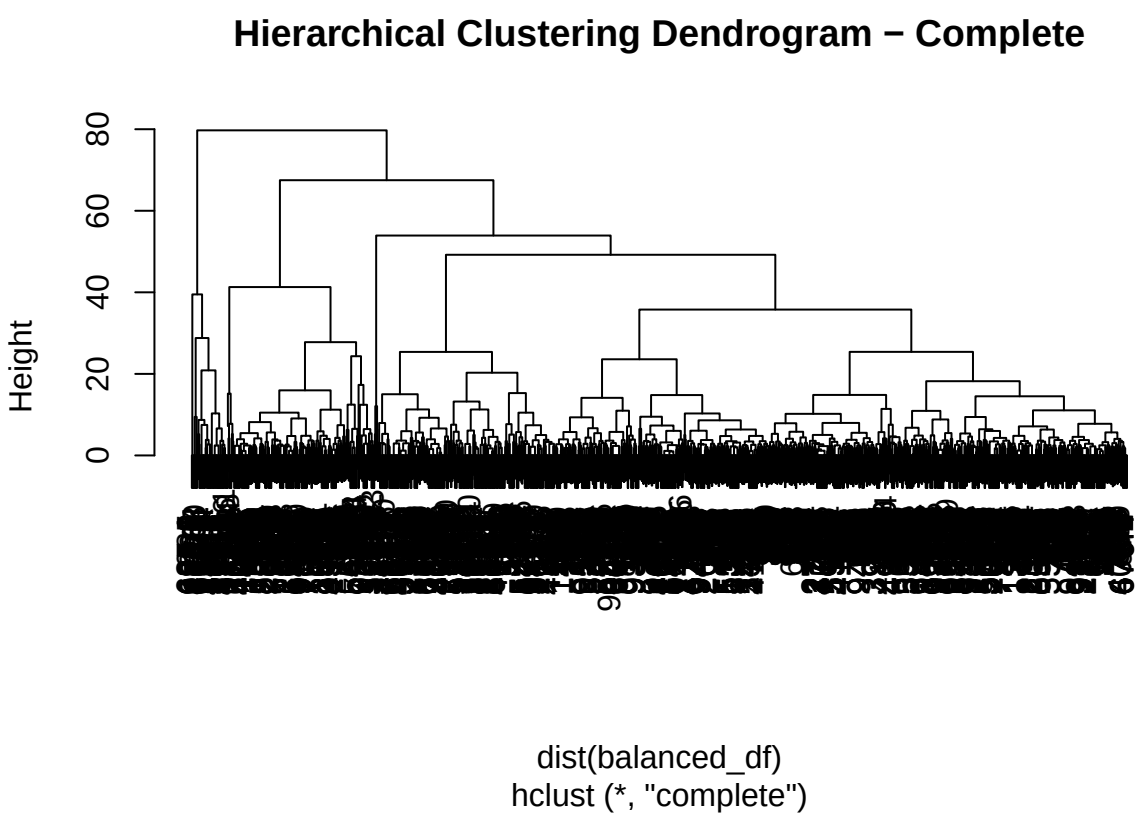
```
## [1] TRUE
```

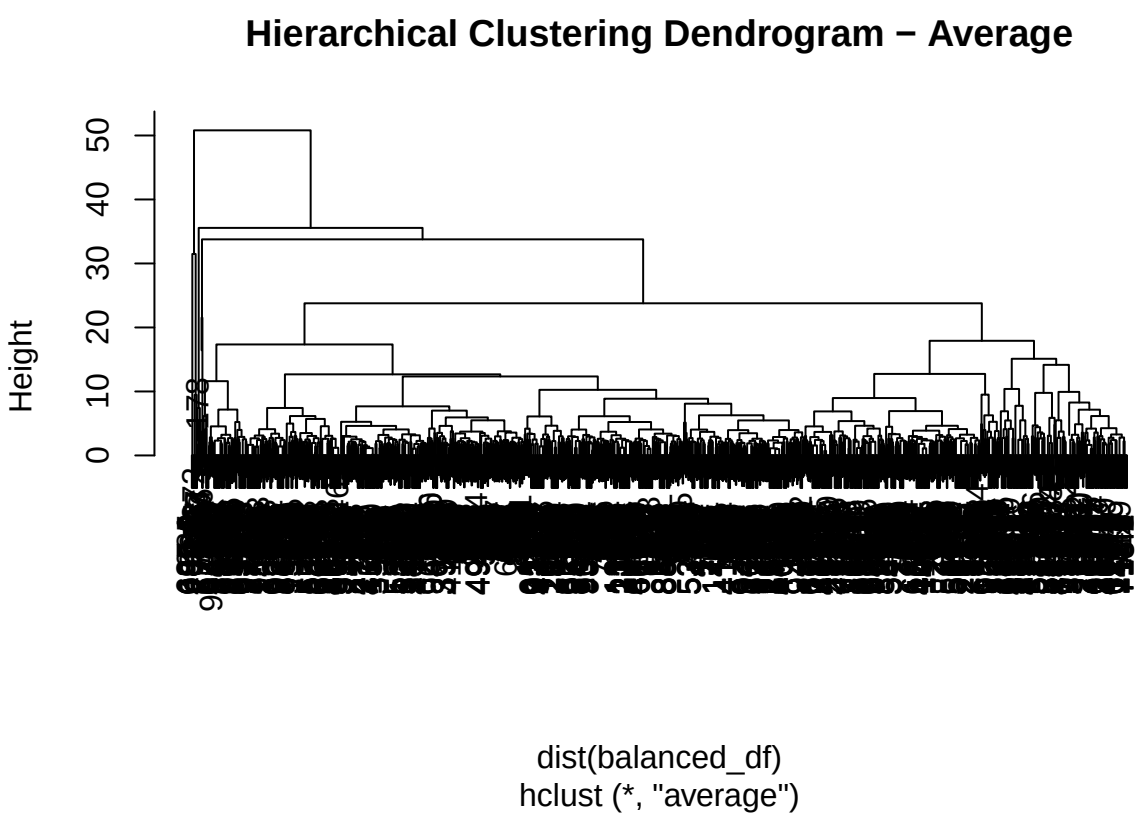
```
##   [1] 1 1 2 2 1 1 1 2 2 2 1 1 1 2 1 1 1 1 1 1 1 2 1 1 2 2 1 2 2 1 1 1 2 2 2 1 2
##  [38] 2 2 1 1 2 1 2 2 1 2 2 1 2 2 1 1 2 2 1 2 1 1 1 1 2 2 2 1 2 2 2 1 2 1 1 2 2
##  [75] 1 1 2 1 2 2 1 1 1 2 2 1 2 1 2 2 2 1 1 1 1 1 1 1 1 1
```

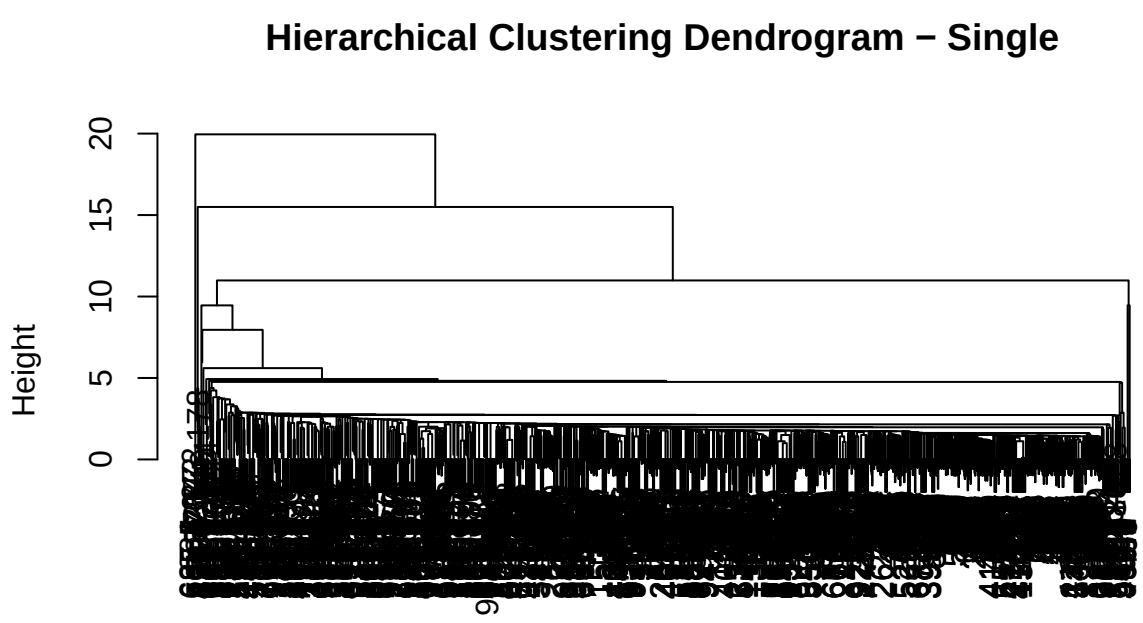
We obtained the k means clusters (where k=2, we have 2 clusters), and now we want to visualize the results of k-means clustering by plotting the clusters along with their centroids.



Hierarchichal Clustering:


Hierarchical Clustering Dendrogram − Complete
Height
0
20
40
60
80
dist(balanced_df)
hclust (*, "complete")


Hierarchical Clustering Dendrogram − Average
Height
0
10
20
30
40
50
dist(balanced_df)
hclust (*, "average")




```
##  925.7    816   23.2    134   11.6     52    805    226  537.1   21.4  926.5
##      1      1      1      1      1      1      1      1      1      1      1
##    6.1    812    999    604    457     79  283.3    959     87    635    517
##      1      1      2      1      1      1      1      1      1      1      1
##    4.1    825   22.1    992  670.4  678.1   43.4    210    148    355    875
##      1      1      1      1      1      1      1      1      1      1      1
##    831    477    649  980.2    123    601  405.4    840    735    188  141.6
##      1      1      1      1      1      1      1      1      1      1      1
##  550.2    878  671.4    750    610    949    390    671    953  407.3  958.1
##      1      1      1      1      1      1      1      1      1      1      1
##    160  930.1    757    244  945.7    687    871    292    695     24    860
##      1      1      1      1      1      1      3      1      1      1      1
##    444    122    662  401.2  284.2  671.1     54    4.2     78   19.1  402.2
##      1      1      1      1      1      1      1      1      1      1      1
##    266    352    542  925.4    327  553.2  143.5  543.3    532    743  141.1
##      1      1      1      1      1      1      1      1      1      1      1
##    508     32     35   43.3    218  101.1    297  668.4    372  419.1  286.1
##      1      1      1      1      1      1      1      1      3      1      1
##    399    554  676.2    910    806    874    914    491  952.1  417.2    224
##      1      1      1      1      1      1      1      1      1      1      1
##   10.2  137.5    793    436   62.3    238    814     16    559    767  982.3
##      1      1      1      1      1      1      1      1      1      1      1
##  270.1    984  674.4     68    647    682     96    930     45    314  570.3
##      1      1      1      1      1      1      1      1      1      1      1
##  133.4    765  267.1    801  276.6    328    412    527    574    752    205
##      1      1      1      1      1      1      1      1      1      1      1
```

```
##   10.6    225     20    709    289   23.3    882  793.2  940.1    552  670.6
##      1      1      1      1      1      1      1      1      1      1      1
##    617     66    692     93  320.2  506.7    214  543.4    567    7.5  414.2
##      1      1      1      1      1      1      1      1      1      1      1
##    299   67.4    684  278.3    849    837    325  490.1  793.5    396   43.5
##      1      1      1      1      1      1      1      1      1      1      1
##    960  299.4    773    472  143.3  279.4  144.3  133.1  685.2   35.2    556
##      1      1      1      1      1      1      1      1      2      1      1
##  529.2    589     69  460.1  399.3  948.2    2.1    342     83    428     90
##      1      1      1      2      1      1      1      1      1      1      1
##  155.1    433   21.2    795  949.1    668    509  416.2    221  945.6     36
##      1      1      1      1      1      1      1      1      1      1      1
##  285.3   25.6  405.1    269  926.1  674.3   18.1    130    206    667    924
##      1      1      1      1      1      1      1      1      1      1      1
##  278.4    142  416.1  547.2  538.4  141.5      4    916  531.2    204  795.1
##      1      1      1      1      1      1      1      1      1      1      1
##    8.2     34   71.1  274.2   24.4   35.3    656    2.3    272  299.1  665.2
##      1      1      1      1      1      1      1      1      1      1      1
##  927.6    350    937    675  668.6    463   25.2    344  553.1  550.5    997
##      1      1      1      1      1      1      1      1      1      1      1
##    925    670    954    976  153.6    115  676.4    607  667.7    400     28
##      1      1      1      1      1      1      1      1      1      1      1
##    273    628  980.1  531.4  795.5  868.3  407.2    609   12.1   66.4  926.4
##      1      1      1      1      1      1      1      1      1      1      1
##    501    305    952    862    389  591.2    117    8.4    511     11    560
##      1      1      1      1      1      1      1      1      1      1      1
##  276.3    228    471  667.1    422    987    961    265    222    4.3  558.3
##      1      1      1      1      1      1      1      1      1      1      1
##    447    664  589.2  101.3    262  999.3     99  793.4    119  399.2  407.1
##      1      1      1      1      1      2      1      1      1      1      1
##    537  268.5    176   10.5    505  268.4     56    594  954.3  222.1  274.1
##      1      1      1      1      1      1      1      1      1      1      1
##   19.2  531.5  162.1    276   43.2    622  153.4  925.3  287.1  490.2    514
##      1      1      1      1      1      1      1      1      1      1      1
##    217  930.2    6.2    497    312    137    638  598.5    824    975    260
##      1      1      1      1      1      1      1      1      1      1      1
##  101.2    496   49.3    337  140.2   16.4    440      3    723   59.1  683.4
##      1      1      1      1      1      1      1      1      1      1      1
##    701    324  490.5    642  282.4    869    353  295.2  957.3    807    693
##      1      1      1      1      1      1      1      1      1      1      1
##    769  287.3  598.4    163  570.2    127    799  538.3    672    405    830
##      1      1      1      1      1      1      1      1      1      1      1
##  937.5    528    216    145    481    785  678.3     46  234.2    868  945.1
##      1      1      1      1      1      1      1      1      1      1      1
##  667.5  589.3     41    453    863    489    373    106    479    897    515
##      1      1      1      1      1      1      1      1      1      1      1
##  795.4    815  406.2    759    138    194    413  149.2    143    614  320.4
##      1      1      1      1      1      1      1      1      1      1      1
##    424    294    395    596    929    207  661.1    158  399.1  144.5  278.5
##      1      1      1      1      1      1      1      1      1      1      1
##    200     22    956  268.2  857.3  670.2    323  857.2    180    955    941
##      1      1      1      1      1      1      1      1      1      1      1
##    739    8.1     48    738    557    521    534    321     94    298    340
##      1      1      1      1      1      1      1      1      1      1      1
```

```
##  136.2   584   282   791  49.1   794 670.3   665     5   780   731
##      1     1     1     1     1     1     1     1     1     1     1
##  954.2   149   848   116   990   362   121   382   513   901 937.2
##      1     1     1     1     1     1     1     1     1     1     1
##  320.3   385 152.2   730   996   669 409.6   753  35.1  14.2  31.1
##      1     1     1     1     1     1     1     1     1     1     1
##    301   928  22.2   884   696   813   335 282.2   732   977   383
##      1     1     1     1     1     1     1     1     1     1     1
##    523   948   745 958.2   239   318   558   212   296   942 682.1
##      3     1     1     1     1     1     1     1     1     1     1
##    113  23.6 667.6 939.2    60   404   637   971   380   834   633
##      1     1     1     1     1     1     1     1     1     1     1
##    880 543.2  54.5 671.6   455   429  66.1 234.4   516   197   259
##      1     1     1     1     1     1     1     1     1     1     1
##    887   288 683.3 273.1    91    70   553    23   762 287.4 718.5
##      1     1     1     1     1     1     1     1     1     1     1
##    758    15 927.2   199 409.5   309  21.1 945.5   456   271   603
##      1     1     1     1     1     1     1     1     1     1     1
##    4.4   980 932.5   307 670.1   316    81   756 888.3 533.3   818
##      1     1     1     1     1     1     1     1     1     1     1
##  598.6   933  16.3   360 153.2   533 940.2 947.1    74 285.2  13.2
##      1     1     1     1     1     1     1     1     1     1     1
##  927.7   253   409   545   7.2   246 222.2   247  62.2   625    49
##      1     1     1     1     1     1     1     1     1     1     1
##   54.3   877     7  19.3 331.2   913   778 674.2   744     1   249
##      1     1     1     1     1     1     1     1     1     1     1
##  948.3   779   430 532.1   786  16.1 153.3   186 276.2 857.1   488
##      1     1     1     1     1     1     1     1     1     1     1
##    520 285.4 888.1   857   254   538   896   467 137.2 133.6  44.1
##      1     1     1     1     1     1     1     1     1     1     1
##  162.3   411   864   531   252   448   101 925.5 268.8    67 399.4
##      1     1     1     1     1     1     1     1     1     1     1
##    905   144   393   691  25.3   541 282.5 405.3   853   706  26.2
##      1     1     1     1     1     1     1     1     1     1     1
##  544.3 598.3 137.1 929.1  24.5   968 278.2   917 150.2 925.1   704
##      1     1     1     1     1     1     1     1     1     1     1
##  409.4 278.1 405.2   402   304 549.5 795.2   593   241   236   423
##      1     1     1     1     1     1     1     1     1     1     1
##  591.1   361 406.1   164   151   102 162.2 939.1   674   772 150.1
##      1     1     1     1     1     1     1     1     1     1     1
##    717  67.2  10.3   237 454.1   654   539 140.1    82   891 980.4
##      1     1     1     1     1     1     1     1     1     1     1
##   22.4 222.4   627   947   278 926.2   886   575   124  30.3 538.1
##      1     1     1     1     1     1     1     1     1     1     1
##   11.1    77   986   727   9.2   263   686 490.3 265.1   985    76
##      1     1     1     1     1     1     1     1     1     1     1
##    185   705   988   915 397.2    80   248   711   114   700   370
##      1     1     1     1     1     1     1     1     1     1     1
##    566   442 409.2   716  61.1   227   994  66.3   452 668.3 278.8
##      1     1     1     1     1     1     1     1     1     1     1
##      8 676.1   502   931 570.1   963   612   120 952.2    31   733
##      1     1     1     1     1     1     1     1     1     1     1
##    872   9.1   766 136.1 939.3   147   788    13   708   322   284
##      1     1     1     1     1     1     1     1     1     1     1
```

```
##   13.1    349    243  409.3    7.1    141  982.1    861     29     51    485
##      1      1      1      1      1      1      1      1      1      1      1
##    707    595    724   16.2   43.1    821  274.3    426    969  718.2  278.6
##      3      1      1      1      1      1      1      1      1      1      1
##    636  544.5  675.2    655    907     19    859  945.4    734    742    763
##      1      1      1      1      1      1      1      1      1      1      1
##     39  868.1     47    946  418.2    330    503    590    921    892  536.2
##      1      1      1      1      1      1      1      1      1      1      1
##     30  506.8    333  284.1    5.2  797.2  683.2    852    474  300.2    287
##      1      1      1      1      1      1      1      1      1      1      1
##  267.3    660  558.2    737  275.2    258    211    198    375  675.1  300.1
##      1      1      1      1      1      1      1      1      1      1      1
##   68.1  544.1    230    547  276.1   26.1  671.2    819    583    209  401.1
##      1      1      1      1      1      1      1      1      1      1      1
##  677.2    577    403    844  414.1    678  283.2    492   10.1  927.5    993
##      1      1      1      1      1      1      1      1      1      1      1
##     10    364    280    443  947.5    565  279.3  948.1  668.5    184    300
##      1      1      1      1      1      1      1      1      1      1      1
##  141.3    587    673  795.3  544.4   22.3     65     37   17.4    995  276.5
##      1      1      1      1      1      1      1      1      1      1      1
##  868.2    751  153.1    525    803    357    890  553.3  265.2  925.2    367
##      1      1      1      1      1      1      1      1      1      1      1
##  667.4  406.3    802  409.7    543    5.1    8.3    100  417.4    331    490
##      1      1      1      1      1      1      1      1      1      1      1
##    909    245  275.1     75    873    169  668.9      2  279.2    823    215
##      1      1      1      1      1      1      1      1      1      1      1
##    644    153  709.1    109    639    118  677.1   67.3    659    820    768
##      1      1      1      1      1      1      1      1      1      1      1
##    449     84   17.3    965    798    792    943  547.3  409.1  598.2  541.1
##      1      1      1      1      1      1      1      1      1      1      1
##    981    348  144.1    290    369    202    133    192    437    811    591
##      1      1      1      1      1      1      1      1      1      1      1
##    585    484    781   68.3    231    722    418  549.1    110    843    775
##      1      1      1      1      1      1      1      1      1      3      3
##    386    179    339    311    623  506.3  268.7    904     43    927    291
##      1      1      1      1      1      1      1      1      1      1      1
##    165    621    856    338  927.3    414    908    845  868.4    911   62.7
##      1      1      1      1      1      1      1      1      1      1      1
##  677.4    468   25.4  495.4     18    345   62.1    233     55  397.4    624
##      1      1      1      1      1      1      1      1      1      1      1
##    183    936  552.1    920    381    478  234.3    407  932.4    177    690
##      1      1      1      1      1      1      1      1      1      1      1
##    605    698    187    935  495.3    973  598.1    991  402.1    782  124.2
##      1      1      1      1      1      1      1      1      1      1      1
##  954.4  932.2    881     38   18.4    619    808    462    240   23.1    661
##      1      1      1      1      1      1      1      1      1      1      1
##  144.6    8.5    470    588    140    507    107    951   36.2    320  947.3
##      1      1      1      1      1      1      1      1      1      1      1
##  506.6  144.4  234.6    178  234.5    582  541.4    657    499  949.2    934
##      1      1      1      1      1      1      1      1      1      1      1
##  940.3    368  795.6    939   18.2  671.7    555    893    128   49.2  857.4
##      1      1      1      1      1      1      1      1      1      1      1
##    461    366    728  405.5  299.3   24.8     97    213    962  543.5    749
##      1      1      1      1      1      1      1      1      1      1      1
```

```
##    736    387  939.5    879    677    944    195    108  506.2  999.1    257
##      1      1      1      1      1      1      1      1      1      2      1
##    699     73  549.3    602  270.2  718.1    279  668.1    285    302    950
##      1      1      1      1      1      1      1      1      1      1      1
##    754    829    256    679    103    486  982.2    979    888  541.3  537.2
##      1      1      1      1      1      1      1      1      1      1      1
##  954.1     26    135    729    421    308  549.2  945.9    867    652    441
##      1      1      1      1      1      1      1      1      1      1      1
##    540  461.1    747    152    922    286  397.5  547.4  495.2    600  267.2
##      1      1      1      1      1      1      1      1      1      1      1
##    150    466   62.6    268    420    586    783    842    397    203    315
##      1      1      1      1      1      1      1      1      1      1      1
##  857.5    2.2    450    653   30.2    989    703    836    512    784    235
##      1      1      1      1      1      1      1      1      1      1      1
##    974  957.1    363    242    694    865  946.3    581    438    688    883
##      1      1      1      1      1      1      1      1      1      1      1
##  532.4    817  137.3  141.7   64.1    721  676.3    498    439     85  946.2
##      1      1      1      1      1      1      1      1      1      1      1
##  532.2   62.5    475   26.5     40    406  932.1    255  529.3    828  540.1
##      1      1      1      1      1      1      1      1      1      1      1
##    267    229    651    898    251    919  718.3  320.1    726   66.2    401
##      1      1      1      1      1      1      1      1      1      1      1
##    190    193    926    159    343    718  414.3    568  285.5    156  459.2
##      1      1      1      1      1      1      1      1      1      1      1
##    858   20.1    809  550.1    155    208    608    480  284.3    415    201
##      1      1      1      1      1      1      1      1      1      1      1
##  550.4   14.3  461.2    506    378    451    648    398    572     57   11.3
##      1      1      1      1      1      1      1      1      1      1      1
##    189    771  665.1    232  277.1    459  794.1    663    432    580    129
##      1      1      1      1      1      1      1      1      1      1      1
##   11.2  685.3     14    313    161    329    685    899    796    640    139
##      1      2      1      1      1      1      2      1      1      1      1
##    310    839   26.4    293  927.1  331.1    774    303    7.3    681    902
##      1      1      1      1      1      1      1      1      1      1      1
##   21.3    111    676    957    876  282.1    885  532.3  143.1    748    454
##      1      1      1      1      1      1      1      1      1      1      1
##    958    563    760    797    702  541.2    918  287.2     42    132  793.6
##      1      1      1      1      1      1      1      1      1      1      1
##  286.2   68.2    641  283.1    105    822    162    561    573  506.4    377
##      1      1      1      1      1      1      1      1      1      1      1
##    680  661.2  549.4    549  667.3    755    894  406.4    576   21.5  668.8
##      1      1      1      1      1      1      1      1      1      1      1
##    616  277.2   22.5  533.2    846    618    777    536  299.2    550    336
##      1      1      1      1      1      1      1      1      1      1      1
##    446    889    131   54.4  133.5    166    173    940    998    417  397.3
##      1      1      1      1      1      1      1      1      1      1      1
##  718.4  124.1  490.4  279.1    275    495    359    425    650  677.3  937.1
##      1      1      1      1      1      1      1      1      1      1      1
##    964  144.2     88  667.2  538.2   11.5  285.1   26.3  149.1    570    464
##      1      1      1      1      1      1      1      1      1      1      1
##    500    776  533.4    544  793.1    841    740    838    620    646  552.2
##      1      1      1      1      1      1      1      1      1      1      1
##    851  947.6    725  589.1    518  879.1    529  544.2    6.3   23.4   30.4
##      1      1      1      1      1      1      1      1      1      1      1
```

```
##    181  143.4     63  945.2    277    332  670.5    146    714    354  671.3
##      1      1      1      1      1      1      1      1      1      1      1
##    445  506.5   87.1  417.1   17.1  543.6    787    606    804  133.2  459.1
##      1      1      1      1      1      1      1      1      1      1      1
##  397.1  495.1    626  538.6  454.3   15.2  402.3    847    535  793.3   11.4
##      1      1      1      1      1      1      1      1      1      1      1
##  533.1    912    710    741  506.1  409.8  558.1    196   67.1  536.1  937.3
##      1      1      1      1      1      1      1      1      1      1      1
##  945.3   24.1  932.3  957.2   14.1    274  529.1  300.3    855  407.5     53
##      1      1      1      1      1      1      1      1      1      1      1
##    388  155.2    827    645     33   44.2    371   24.6    346    764   30.1
##      1      1      1      1      1      1      1      1      1      1      1
##    493     89    526    112    365  797.1  418.1    469    761    510  454.2
##      1      1      1      1      1      1      1      1      1      1      1
##    174     92    866   36.1    598  668.7    770    427  285.6    295    643
##      1      1      1      1      1      1      1      3      1      1      1
##  299.5    157  275.3   31.2   62.4    967  925.6  547.1    410    319    548
##      1      1      1      1      1      1      1      1      1      1      1
##    104    854    317  153.5    167  282.3    168  937.4  140.3    800    504
##      1      1      1      1      1      1      1      1      1      1      1
##    379 945.10    358    906   24.2    220  531.1  417.3    9.3  946.1     17
##      1      1      1      1      1      1      1      1      1      1      1
##    136      6     44    666    571    435    689    125    713    416    458
##      1      1      1      1      1      1      1      1      1      1      1
##    579     62    476  278.7  927.4  668.2  268.6  945.8  662.1    983  407.4
##      1      1      1      1      1      1      1      1      1      1      1
##    182  141.4  268.1    431    2.4   87.3    483    281  295.1    519    419
##      1      1      1      1      1      1      1      1      1      1      1
##    632    374   54.2  570.4   54.1    460    611    790   20.2    562     27
##      1      1      1      1      1      2      1      1      1      1      1
##  929.2     72  278.9  234.7  407.6  531.3    945  143.2    223  947.2    154
##      1      1      1      1      1      1      1      1      1      1      1
##    546    932    629    283  222.3  926.6    347     86    376  268.3    234
##      1      1      1      1      1      1      1      1      1      1      1
##  543.1  999.2  141.2  926.3    978    923    522    683    712    895    392
##      1      2      1      1      1      1      1      1      1      1      1
##    270  152.1     64  888.2  550.3    172    175    597    826    715    719
##      1      1      1      1      1      1      1      1      1      1      1
##    494    970    832    7.4    191   17.2   23.5   24.7    658    434    966
##      1      1      1      1      1      1      1      1      1      1      1
##  671.5     58  416.3    126  685.1  133.3    334    569    473    524     61
##      1      1      1      1      2      1      1      1      1      1      1
##    219    903  673.1    408    394   15.1    599    850  273.2    697   10.4
##      1      1      1      1      1      1      1      1      1      1      1
##    551    615   18.3     95  137.4    326  538.5  939.4   25.5    264  947.7
##      1      1      1      1      1      1      1      1      1      1      1
##     71  276.4     21   25.1     50    613    578    171    938  234.1    592
##      1      1      1      1      1      1      1      1      1      1      1
##    384    900    530    810    870    789  454.4    306    341    487    261
##      1      1      1      1      1      1      1      1      1      1      1
##    835  947.4  980.3    170   24.3     59    351    982  678.2     98  674.1
##      1      1      1      1      1      1      1      1      1      1      1
##   87.2     12   49.4    391    250    482    833  461.3  683.1     25    972
##      1      1      1      1      1      1      1      1      1      1      1
```

```
##    356      9  415.1  101.4    746    564    631    465
##      1      1      1      1      1      1      1      1
```

We applied hierarchical clustering to the dataset, which is an unsupervised machine learning technique used to group similar data points based on their distance or dissimilarity. We performed hierarchical clustering with different linkage methods: complete linkage, average linkage, and single linkage. Each method differs in how it calculates the distance between clusters as they are merged. We then visualized the clustering results by plotting dendrograms which show how observations are merged at different distance levels. Finally, we assigned cluster labels to the observations, cutting the dendrogram at k=3 to create three clusters.