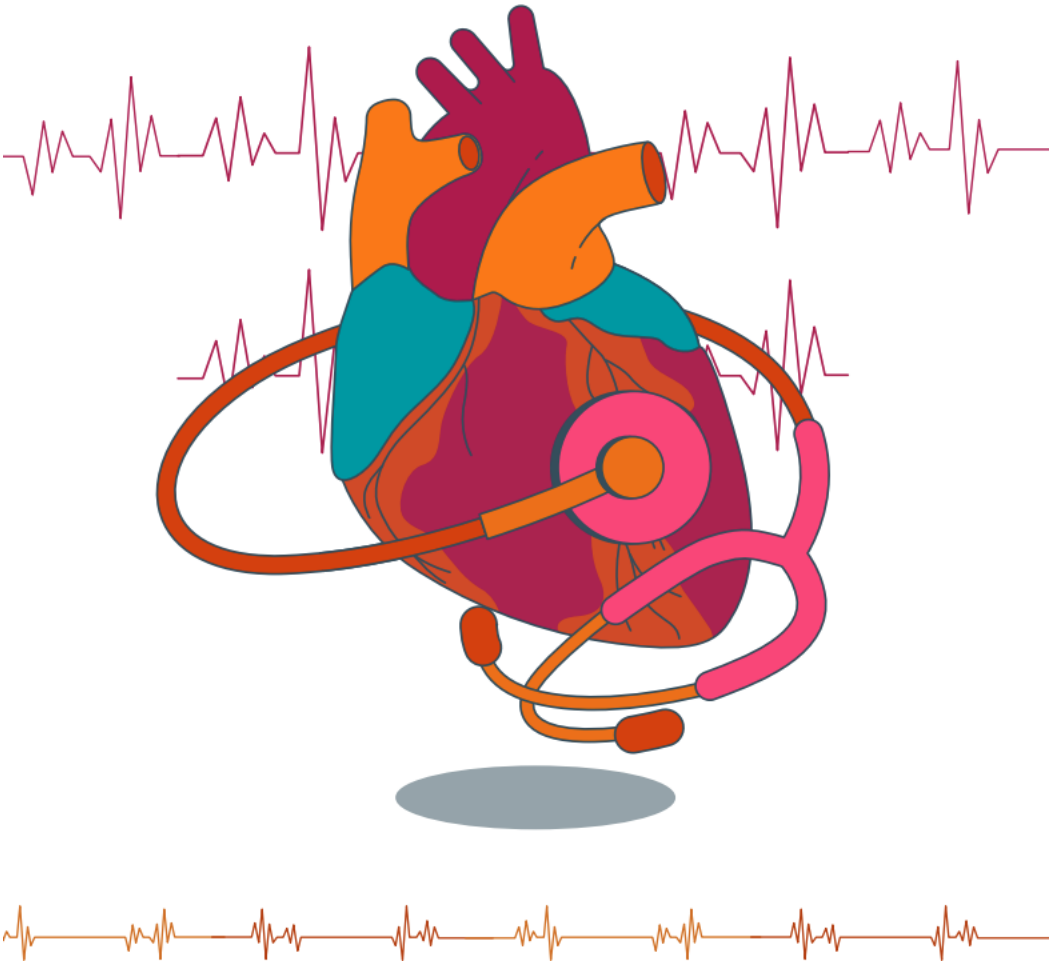
AI-DRIVEN STATISTICAL MODELING AND MACHINE LEARNING ANALYSIS OF CARDIOVASCULAR PATIENT HOSPITAL ADMISSIONS



FINAL PROJECT REPORT

SANJIR INAM SALSABIL

DEPARTMENT OF DATA SCIENCE

JANUARY 19, 2023

TABLE OF CONTENTS

LIST OF FIGURES

1. INTRODUCTION

The human body constitutes of many diverse types of cells that together create tissues and subsequently organ systems. The heart is at the focal point of the human body responsible for pumping blood throughout the body and keeping us alive. Our focus of analysis are patients admitted to the hospital having cardiovascular disease which is one of the leading causes of death globally. Below is an analysis of the top 5 leading causes of death in Canada from 2017-2021. It is observed from the highlighted portions that death related to heart related diseases occupy 2nd and 4th rank for majority of the years.

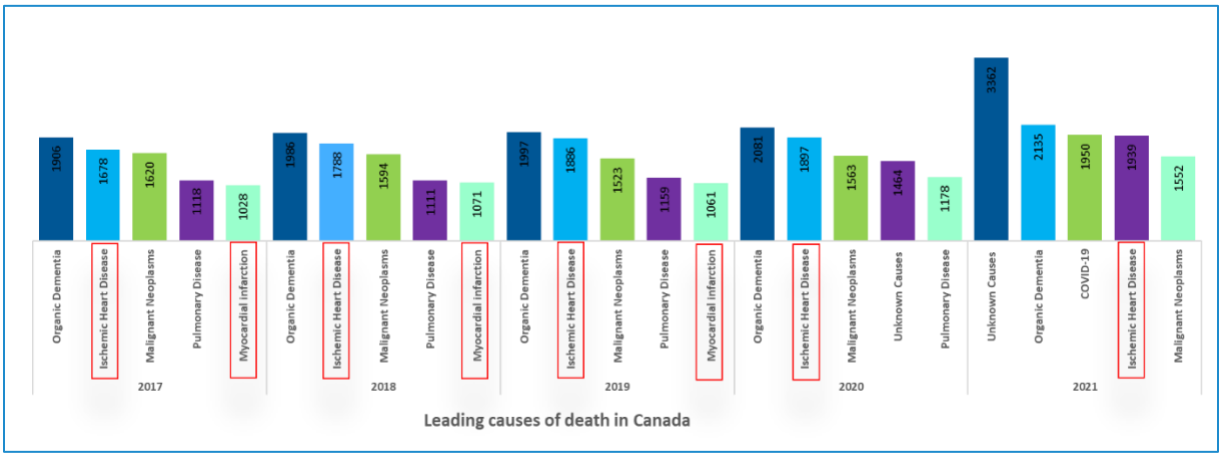


Figure 1: Bar plot showing the top 5 leading causes of death in Canada from 2017-2021

We tried to identify the following from our analysis:

- Observing patients admitted to the hospital with diverse cardiovascular diseases
- Analyzing the underlying conditions associated with the patient such as heart failure, acute kidney disease
- Analyzing the outcome of the patient for the duration of hospital admission

We had selected two datasets from Kaggle and described them in the next section.

2. DATASETS UTILIZED

We have utilized the "Hospital Admissions Data" dataset (File size: 2.6 MB, Rows: 15757 K, Columns: 56) available in the Kaggle portal (https://www.kaggle.com/datasets/ashishsahani/hospital-admissions-data/discussion/302894?resource=download&select=HDHI+Admission+data.csv)**[Ref. 1]** in CSV format.

The Kaggle portal provides this dataset free of charge and can be used for research/project purposes. Details of their conditions are available in https://doi.org/10.3390/diagnostics12020241**[Ref. 2]**. This dataset is being provided under creative commons License (Attribution-Non-Commercial-Share Alike 4.0 International (CC BY-NC-SA 4.0)) https://creativecommons.org/licenses/by-nc-sa/4.0/ [**Ref. 3]**.

This data was collected from patients admitted over a period of two years (1 April 2017 to 31 March 2019) at Hero DMC Heart Institute, Unit of Dayanand Medical College and Hospital, Ludhiana, Punjab, India.

During the study period, the cardiology unit had 14,845 admissions corresponding to 12,238 patients. 1921 patients who had multiple admissions.

DATASET	ATTRIBUTES	DESCRIPTION	VARIABLE TYPE
HDHI Admission data	Rows:15757, Columns:56 File Type: CSV	Dataset were related to patients • date of admission and discharge • demographics • type of admission and outcome • patient history such as smoking, alcohol, hypertension. • lab parameters such as hemoglobin, platelets, glucose, urea, creatinine. • Other 28 features including heart failure, acute kidney disease.	Categorical: 40 Numerical: 16

DATASET	ATTRIBUTES	DESCRIPTION
Table Headings	Rows: 57, Columns:2 File Type: CSV	This data table has the descriptive headlines for all columns for the HDHI Admission data file.

Figure 2: Overview of the datasets utilized

3. DESCRIPTION OF SOME VARIABLES

As presented in previous item, the original dataset contains several variables. We used below elimination method:

- Qualitative variables: By counting the "1" for the qualitative variables. When the count was less than 1000 then we dropped that variable as it was insignificant compared to the data size.
- Quantitative variables: Most were kept except for BNP (B-TYPE NATRIURETIC PEPTIDE) which had 10000 entries missing. We also dropped Serial No. and MRD No. as it was not relatable.

At the end the main ones used in this project (for example, indicated significant by some models) are described in **Figure 3**.

MAIN QUANTITATIVE VARIABLES	
DURATION.OF.STAY	Quantity of days the patient stayed on the hospital (Days)
AGE	Age of patient on hospital admission (Years)
GLUCOSE	Level of Glucose of the patient (mg/dL)
HB	Level of Hemoglobin (g/dL)
TLC	Total Leukocytes Count (units)
CREATININE	Level of Creatinine of the patient (mg/dL)
UREA	Level of UREA of the patient (mg/dL)
EF	Ejection Fraction Index (%)
PLATELETS	Platelets count (units)

MAIN QUALITATIVE VARIABLES	
HEART.FAILURE	Heart Failure (1,0 or Y,N)
AKI	Acute Kidney Injury (1,0 or Y,N)
OUTCOME	Hospital Outcome (EXPIRED = died, DISCHARGE by doctor approval, DAMA = Discharge without approval)
GENDER	Patient gender: Male of Female (M, F)
RAISED.CARDIAC.ENZYMES	The patient has high levels of cardiac enzymes (1,0 or Y,N)
PRIOR.CMP	The patient has Cardiomyopathy (1,0 or Y,N)
STEMI	stands for ST Elevation Myocardial Infarction(1,0 or Y,N)
DM	The patient has Diabetes Mellitus (1,0 or Y,N)
HTN	The patient has Hypertension (1,0 or Y,N)
CAD	The patient has Coronary Artery Disease (1,0 or Y,N)
CKD	The patient has Chronic Kidney Disease (1,0 or Y,N)
STABLE.ANGINA	The level of Angina is stable (1,0 or Y,N)
ALCOHOL	The patient consumes Alcohol (1,0 or Y,N)
SMOKING	The patient smokes (1,0 or Y,N)

Figure 3: Main Variables used from the Dataset

In Figure 3, the response variables that were investigated on this project were: HEART.FAILURE, AKI, OUTCOME and DURATION.OF.STAY.

4. RESEARCH QUESTIONS

1. Which independent variables are the most important to predict heart failure and acute kidney injury (AKI)?
2. Which statistical learning method best predicts heart failure and AKI in terms of misclassification rate?
3. What would be the true mean of duration of patients stay compared with the means by using different sampling methods (SRS, and stratified sampling)?
4. Can we use simple Linear Regression to predict a patient's "Duration of Stay" in admission from lab test results and gender?
5. In terms of misclassification rate which model best predicts the 3 levels of the variable OUTCOME (Expiry, Discharge, DAMA)?

5. DATA WRANGLING

Our main dataset, HDHI admission dataset had over 15757 rows and 56 columns. After initial analysis we found duplicate entries based on one of the columns (MRD No.) which was the admission no for patients. Admission numbers were unique as they vary every time a patient gets admitted to the hospital. So, we removed the duplicate entries (3513 rows). We also conducted the below cleaning steps:

1. Removal of date of admission, date of discharge and month year column to avoid time-based dependency.
2. Removal of null values and rows having "EMPTY" entries.
3. Removal of unwanted columns which we already explained in detail under Section 3 of our report.
4. Factorizing categorial variables for applying modelling techniques
5. Converting character into integers for numeric columns.

Our final dataset had 10125 rows and 29 entries after all the data cleaning.

It was mentioned in Kaggle that 1921 patients were admitted multiple times. However, there was no way to identify these patients as patient IDs (unique to each patient) were not provided in the dataset due to privacy issues. We had to assume independence among all the data entries as it was not possible to segregate among them.

6. STATISTICAL METHODS UTILIZED

For our analysis and modelling we applied several techniques.

1. Initially we started with sampling (SRS and stratified sampling) to have a generalized idea of the cleaned dataset.
2. Next, we used the 2-by-2 contingency table to identify the categorical variables that were independent among each other but were related to our response variable (Heart Failure and AKI).
3. We applied Logistic Regression, Linear Discriminant Analysis (LDA), Quadratic Discriminant Analysis (QDA), Classification tree and cross validation to identify the best model using Heart Failure and AKI as response variable.
4. We then applied linear regression and regression tree on duration of stay as response variable and one categorial variable(gender) and all quantitative variables as explanatory variables.
5. Our last analysis focused on multinomial regression, classification tree and cross validation in order to predict the Outcome variable which had 3 levels (Discharge, DAMA, Expiry).

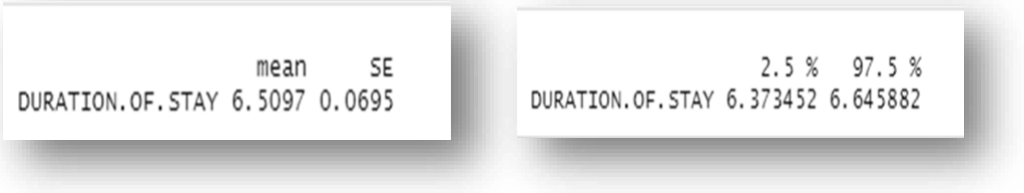
7. ANALYSIS

7.1 SAMPLING

After cleaning the dataset, the quantitative variables that remained were DURATION.OF.STAY, AGE, Hemoglobin (HB), TOTAL LEUKOCYTES COUNT (TLC), PLATELETS, UREA, CREATININE, GLUCOSE and Ejection Fraction (EF). As all these variables were either demographic or lab parameters of the patients so we focused on DURATION.OF.STAY as our response variable for sampling.

7.1.1 SIMPLE RANDOM SAMPLING (SRS)

For simple random sampling our response variable was DURATION.OF.STAY for the patients in the hospital. Our population size was 10125 and our sample size was 3000 which was almost one-third of the population size. The outcome of the SRS are shared below:



	mean	SE
DURATION.OF.STAY	6.5097	0.0695

	2.5 %	97.5 %
DURATION.OF.STAY	6.373452	6.645882

Figure 4: - Result from Simple Random Sampling (SRS)

From the above results we can conclude that on average patients stayed 6.5 days in the hospital irrespective of their underlying conditions with standard deviation of 0.0695.

7.1.2 STRATIFIED SAMPLING

For stratified sampling we also used DURATION.OF.STAY as the response variable, but we used 2 different stratums. In one case we used TYPE OF ADMISSION-EMERGENCY/OPD (emergency/outdoor patient) as the stratum and in another case, we used OUTCOME (discharge, expiry and DAMA) as the stratum. Below is an overview of the DURATION.OF.STAY with respect to these two stratums.

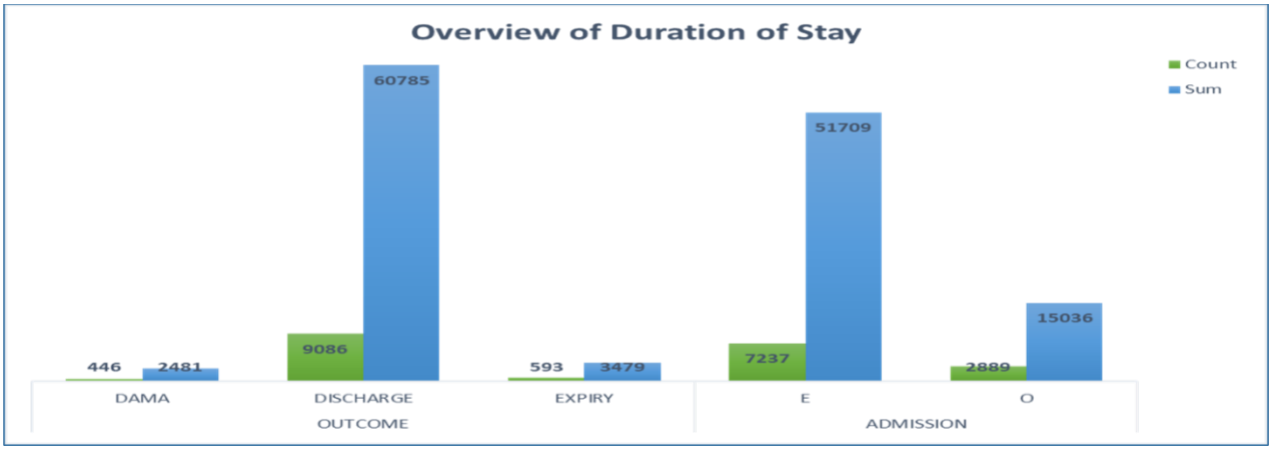
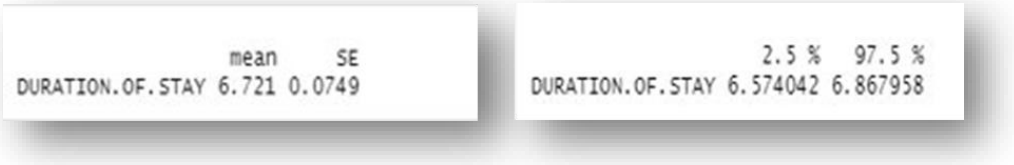


Figure 5: Bar plot showing the duration of stay with respect to the stratums

The above bar plot shows the count and sum of the DURATION.OF.STAY based on every strata of the two stratums. We observed that DURATION.OF.STAY had the highest entry both in terms of overall count and sum for DISCHARGE under OUTCOME stratum followed by EXPIRY and DAMA (discharge against medical advice). From the TYPE OF ADMISSION-EMERGENCY/OPD stratum, DURATION.OF.STAY was highest for emergency admitted patients.

First we proceed with stratified sampling using DURATION.OF.STAY as the response variable and TYPE OF ADMISSION-EMERGENCY/OPD as the stratum. We used proportional allocation to find the individual sample size of each strata with an overall sample size of 3000. We used sample size of Emergency-2144 and OPD-856 and obtained the below results.



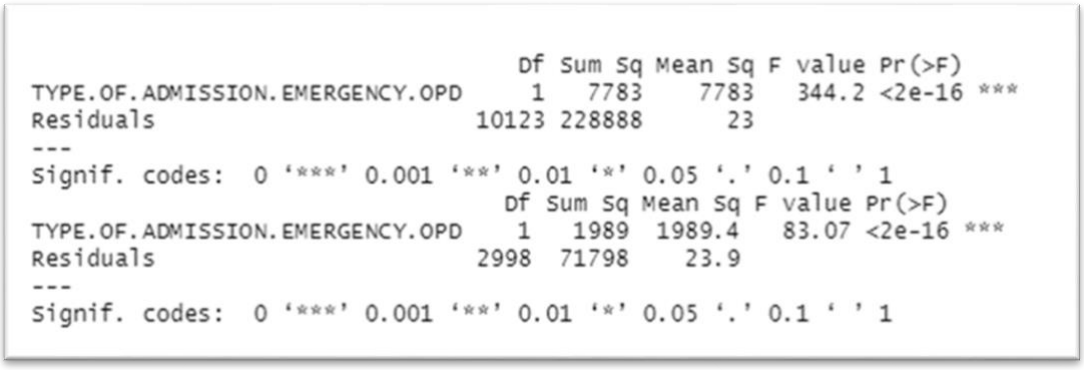
```
                    mean     SE
DURATION.OF.STAY 6.721 0.0749
```

```
                     2.5 %   97.5 %
DURATION.OF.STAY 6.574042 6.867958
```

Figure 6: Result from Stratified sampling with TYPE OF ADMISSION as stratum

From the above results we can conclude that on average patients stayed 6.7 days in the hospital irrespective of their underlying conditions with standard deviation of 0.0749.

Proceeding with the Anova table for further comparison between the stratified sampling dataset and the overall population dataset we obtained the below result.

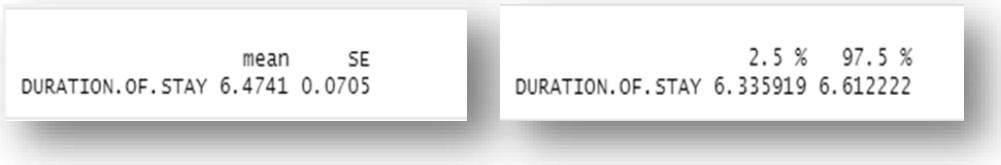


```
                                 Df Sum Sq Mean Sq F value Pr(>F)
TYPE.OF.ADMISSION.EMERGENCY.OPD     1   7783    7783   344.2 <2e-16 ***
Residuals                       10123 228888      23
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
                                Df Sum Sq Mean Sq F value Pr(>F)
TYPE.OF.ADMISSION.EMERGENCY.OPD    1   1989  1989.4   83.07 <2e-16 ***
Residuals                       2998  71798    23.9
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Figure 7: ANOVA table comparison with TYPE OF ADMISSION as stratum

From the above result we observed that p-value is significant for both the cases which means that TYPE OF ADMISSION-EMERGENCY/OPD is an important variable impacting the response variable DURATION.OF.STAY. However, the SSB is very small when compared to the SSW which means that stratified sampling is not a good fit for this model. We also calculated the estimated variance and found it as 62.46922 which is smaller than SSB but not significantly smaller.

Next we proceed with stratified sampling using DURATION.OF.STAY as the response variable and OUTCOME as the stratum. We used proportional allocation to find the individual sample size of each strata with an overall sample size of 3000. We used sample sizes of DAMA-132, DISCHARGE-2692 and EXPIRY-176 and obtained the below results.



```
                    mean     SE
DURATION.OF.STAY 6.4741 0.0705
```

```
                     2.5 %   97.5 %
DURATION.OF.STAY 6.335919 6.612222
```

Figure 8: Result from Stratified sampling with OUTCOME as stratum

From the above results we can conclude that on average patients stayed 6.47 days in the hospital irrespective of their underlying conditions with standard deviation of 0.0705.

Proceeding with the Anova table for further comparison between the stratified sampling dataset and the overall population dataset we obtained the below result.

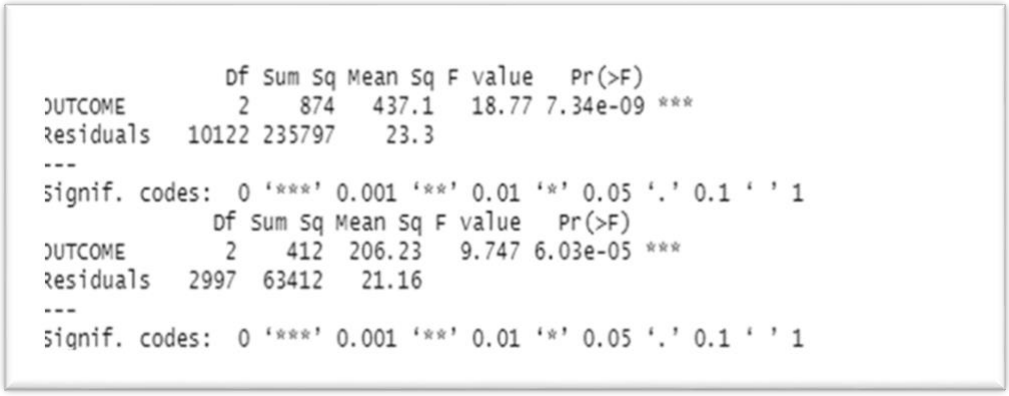


Figure 9: ANOVA table comparison with OUTCOME as stratum

From the above result we observed that p-value is significant for both the cases which means that OUTCOME is an important variable impacting the response variable DURATION.OF.STAY. However, the SSB is negligible when compared to the SSW which means that stratified sampling is not a good fit for this model. We also calculated the estimated variance and found it as 76.86822 which is smaller than SSB but not significantly smaller.

7.1.3. COMPARISON OF SAMPLING TECHNIQUES

Upon comparing the sampling techniques, we received the below outcome:

Type	Mean
Population	6.5935
SRS	**6.5097**
Stratified(Admission)	6.721
Stratified(Outcome)	6.4741

Figure 10: Comparison of the sampling techniques

So we conclude the following:

1. SRS gives better sampling results than stratified sampling based on the comparison of the means obtained from the sampling techniques and the population mean.

2. Even though the stratum brings significant p-value for both cases, but the SSB is negligible compared to the SSW which does not make stratified sampling a good fit.
3. Estimate of variance is not significantly smaller than SSB so we can say the stratified sampling does no give a stable estimate of variance.
4. Only quantitative variables could be used as the variable of interest and duration of stay providing the best option.

7.2. CONTINGENCY TABLE FOR INDEPENDENCE CHECKING

As shown in Section 3, we could reduce the number of variables by dropping some columns which have significant amount of missing data for quantitative variables (e.g., BNP), and for categorical ones without relevant values in specific class (e.g., INFECTIVE ENDOCARDITIS). However, there are still more than 10 categorical variables to be considered for modelling.

For modelling, it is important to consider only explanatory variables which are dependent on response variables, but at the same time they must be independent among them. For quantitative variables, it is common not to use anything related to time series, except if the time series techniques indicate that the parameters are stationary (e.g., by ADF test), but specific statistical learning methods are applied (e.g., ARMA, AR, MA). However, for categorical variables which do not have influence of time, it was learnt in DATA-606 that we can infer the independence using Contingency Table by doing different tests (e.g., test the ratio, test based on the differences, test the odds ratio). Based on this, we applied this to pairs (2x2 Contingency table using Test based on the differences) of our remaining qualitative variables as for example presented below when checking HEART.FAILURE with HTN.

```
[1] "HEART.FAILURE" "HTN"
                 -           -
                 Cases People at risk         Risk
Exposed   3.771000e+03   7.217000e+03 5.225163e-01
Unexposed 1.479000e+03   2.908000e+03 5.085970e-01
Total     5.250000e+03   1.012500e+04 5.185185e-01

        Risk difference and its significance probability
(H0: The difference equals to zero)

data:  3771 1479 7217 2908
p-value = 0.2048
95 percent confidence interval:
 -0.007597001  0.035435615
sample estimates:
[1] 0.01391931
```

Figure 11: Test based on the differences for HEART.FAILURE and HTN table contingency

It is possible to verify that the test based on the risk difference indicated for the categorical variables HEART.FAILURE and HTN a p-value higher than 0.05 which we should not reject the null hypothesis and conclude that they are independent at 5% level.

The results in terms of p-value for pair of all remaining (after data cleaning) categorical variables are presented in the heatmap below. Each cell indicates the results of 2x2 Table Contingency table using Test based on the Risk Differences for the respective pair of variables.

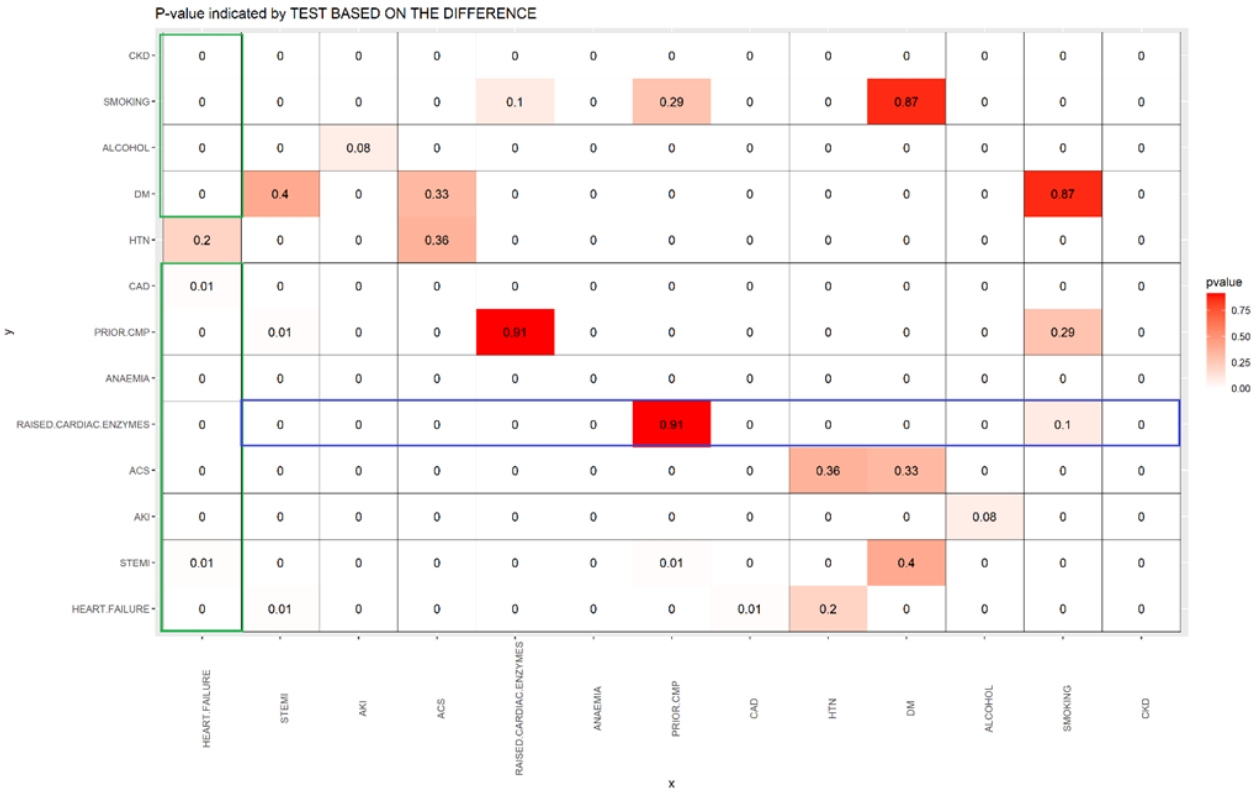


Figure 12: Heatmap with summary results (p-value) indicated test based on the difference applied 2x2 Table Contingency for pair of categorical variables

In **Figure 12**, it is possible to see that the test based on the difference only indicated HTN as independent variable of HEART.FAILURE because p-value is higher than 0.05, and consequently we cannot consider HTN as an explanatory variable of HEART.FAILURE. In addition, we can see that if we select RAISED.CARDIAC.ENZYMES as an explanatory variable of HEART.FAILURE, it is only possible to include PRIOR.CMP and SMOKING as other explanatory variables seeing that they are independent among them, but at the same time dependent of the response variable HEART.FAILURE. Based on this, we could obtain from the heatmap the possibilities of HEART.FAILURE models with at least two explanatory variables (with only one explanatory variable it is also possible for all the categorical values, except HTN) the following models:

a. HEART.FAILURE in function of RAISED.CARDIAC.ENZYMES, PRIOR.CMP and SMOKING
b. HEART.FAILURE in function of RAISED.CARDIAC.ENZYMES and PRIOR.CMP
c. HEART.FAILURE in function of RAISED.CARDIAC.ENZYMES and SMOKING
d. HEART.FAILURE in function of PRIOR.CMP and SMOKING
e. HEART.FAILURE in function of STEMI and DM
f. HEART.FAILURE in function of AKI and ALCOHOL
g. HEART.FAILURE in function of ACS and DM

For AKI as response variable, ALCOHOL is independent of this response variable and consequently must not be an explanatory variable. For the other ones, with at least two explanatory variables, the possibilities based on the heatmap are:

a. AKI in function of STEMI and DM
b. AKI in function of ACS and DM
c. AKI in function of ACS and HTN
d. AKI in function of RAISED.CARDIAC.ENZYMES, PRIOR.CMP and SMOKING
e. AKI in function of RAISED.CARDIAC.ENZYME and, PRIOR.CMP
f. AKI in function of RAISED.CARDIAC.ENZYMES and SMOKING
g. AKI in function of PRIOR.CMP and SMOKING
h. AKI in function of HEART.FAILURE and HTN

In summary, we could find by the test of independence by applying test based on the differences using 2x2 Table Contingency the dependent explanatory variables of our response variables, at the same time these explanatory variables being independent among them.

7.3. MODELLING & PREDICTION

This part shows the models that were created for different response variables, qualitative ones (HEAT.FAILURE and AKI) and quantitative (DURATION.OF.STAY). For modelling of qualitive response variables, the cleaned dataset was initially divided in 75% train part and 25% test part using stratified sampling and created the models using the statistical learning methods described below, and after were created new model with each method below using 10-fold stratified cross-validation (it is only presented the summary of this at the end).

Statistical learning methods applied for our categorical responses with only two possible responses (HEART.FAILURE and AKI): Logistic Regression, Linear Discriminant Analysis (LDA), Quadratic Discriminant Analysis (QDA) and Classification Tree. For categorical ones with three possible responses (OUTCOME), models were built with Multinomial and Classification tree.

Statistical learning methods applied for our quantitative response (DURATION.OF.STAY): Linear Regression and Regression Tree.

In terms of explanatory variables, we used initially all the quantitative variables after our data was cleaned, but for the categorical ones, we used for only the possibilities indicated by the test of independence obtained with 2x2 table contingency with test based on risk differences presented on item 7.2. It is only shown here one of the possible combinations of categorical variables presented in the previous item. For the quantitative response, we applied initially all the quantitative explanatory variables and only GENDER as qualitative. Based on the initial modelling (logistic and linear regression), some of the initial explanatory variables were removed by not being significant at 5% level, or problem of multicollinearity, or the coefficient did not show what was expected, or based on some assumption for the statistical learning method (e.g., LDA assumption of normal distribution of variables).

7.3.1 CATEGORICAL RESPONSE VARIABLE (HEART FAILURE)

For HEART.FAILURE as response variable, some preliminary analysis with LOGISTIC REGRESSION was done initially considering the different possibilities of the model with the categorical variables indicated by the independence test presented in item 7.2. It was selected not only based on the lowest

misclassification rate, but also the one with more categorical explanatory variables including GENDER and all the dependent quantitative variables.

Based on that, the analysis started considering HEART.FAILURE in function of RAISED.CARDIAC.ENZYMES, PRIOR.CMP, SMOKING, GENDER and all dependent quantitative variables initially. The logistic regression was the first statistical learning method applied seeing that the p-values of each explanatory variable initially also helped to remove not significant ones. After that, we kept the same variables for all the other analysis.

7.3.1.1 HEART FAILURE: LOGISTIC REGRESSION

The first logistic regression model considering RAISED.CARDIAC.ENZYMES, PRIOR.CMP, SMOKING, GENDER and all the quantitative dependent variables obtained based on the train part is presented below.

```
Call:
glm(formula = HEART.FAILURE ~ factor(GENDER) + AGE + GLUCOSE +
    HB + TLC + PLATELETS + UREA + CREATININE + EF + factor(RAISED.CARDIAC.ENZYMES) +
    factor(PRIOR.CMP) + factor(SMOKING), family = binomial, data = train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.6565  -0.7346  -0.4786   0.8514   2.7395

Coefficients:
                                 Estimate Std. Error z value Pr(>|z|)
(Intercept)                     0.9412779  0.3076927   3.059 0.002220 **
factor(GENDER)M                -0.3755875  0.0634067  -5.923 3.15e-09 ***
AGE                             0.0151023  0.0023756   6.357 2.05e-10 ***
GLUCOSE                         0.0012053  0.0003197   3.770 0.000163 ***
HB                             -0.0783684  0.0140360  -5.583 2.36e-08 ***
TLC                             0.0154811  0.0044585   3.472 0.000516 ***
PLATELETS                      -0.0003877  0.0002878  -1.347 0.178044
UREA                            0.0066479  0.0011014   6.036 1.58e-09 ***
CREATININE                     -0.0982801  0.0350772  -2.802 0.005081 **
EF                             -0.0554185  0.0029041 -19.083  < 2e-16 ***
factor(RAISED.CARDIAC.ENZYMES)1 0.4327435  0.0652899   6.628 3.40e-11 ***
factor(PRIOR.CMP)1              0.5820183  0.0871285   6.680 2.39e-11 ***
factor(SMOKING)1               -0.3204094  0.1421573  -2.254 0.024202 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 9107.1  on 7593  degrees of freedom
Residual deviance: 7496.5  on 7581  degrees of freedom
AIC: 7522.5

Number of Fisher Scoring iterations: 4


                                        2.5 %        97.5 %
(Intercept)                     0.3380351900  1.5443919831
factor(GENDER)M                -0.4999546664 -0.2513675853
AGE                             0.0104621944  0.0197758048
GLUCOSE                         0.0005780096  0.0018316580
HB                             -0.1058905190 -0.0508597872
TLC                             0.0069267176  0.0243862091
PLATELETS                      -0.0009542849  0.0001744202
UREA                            0.0045023370  0.0088224100
CREATININE                     -0.1679815306 -0.0302947647
EF                             -0.0611389269 -0.0497527336
factor(RAISED.CARDIAC.ENZYMES)1 0.3046014661  0.5605689273
factor(PRIOR.CMP)1              0.4115782911  0.7531665960
factor(SMOKING)1               -0.6042808020 -0.0464456706
```

Figure 13: Logistic Regression model for HEART.FAILURE considering RAISED.CARDIAC.ENZYMES, PRIOR.CMP, SMOKING, GENDER and all the quantitative dependent variables

Based on the results in **Figure 13**, it is possible to verify that PLATELETS is the only variable with p-value higher than 0.05 that it is possible to conclude that this is not significant at 5% and consequently it should be removed from the model. In addition, at 95% confidence interval, it was the only one with zero between lower and upper bounds. Another variable that it was decided to remove SMOKING seeing that the signal of the coefficient of this is negative, in other words, smoking helps to reduce heart failure, that this is not what is expected for this variable. In addition, SMOKING was the least significant variable among all the relevant variables in this figure.

After removing PLATELETS and SMOKING, the final logistic model for HEART.FAILURE is presented below.

```
Call:
glm(formula = HEART.FAILURE ~ factor(GENDER) + AGE + GLUCOSE +
    HB + TLC + CREATININE + UREA + EF + factor(RAISED.CARDIAC.ENZYMES) +
    factor(PRIOR.CMP), family = binomial, data = train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.6277  -0.7381  -0.4793   0.8544   2.6576

Coefficients:
                                Estimate Std. Error z value Pr(>|z|)
(Intercept)                    0.7838871  0.2922122   2.683 0.007305 **
factor(GENDER)M               -0.3905739  0.0625224  -6.247 4.19e-10 ***
AGE                            0.0156926  0.0023666   6.631 3.34e-11 ***
GLUCOSE                        0.0012128  0.0003194   3.797 0.000147 ***
HB                            -0.0767528  0.0139145  -5.516 3.47e-08 ***
TLC                            0.0142416  0.0043806   3.251 0.001150 **
CREATININE                    -0.0966244  0.0350299  -2.758 0.005810 **
UREA                           0.0067814  0.0010966   6.184 6.24e-10 ***
EF                            -0.0552717  0.0029030 -19.039  < 2e-16 ***
factor(RAISED.CARDIAC.ENZYMES)1  0.4306312  0.0652276   6.602 4.06e-11 ***
factor(PRIOR.CMP)1             0.5882911  0.0870157   6.761 1.37e-11 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 9107.1  on 7593  degrees of freedom
Residual deviance: 7503.6  on 7583  degrees of freedom
AIC: 7525.6

Number of Fisher Scoring iterations: 4


                                        2.5 %       97.5 %
(Intercept)                       0.2108339659  1.356496889
factor(GENDER)M                  -0.5131946993 -0.268074301
AGE                               0.0110701812  0.020348625
GLUCOSE                           0.0005859531  0.001838617
HB                               -0.1040340104 -0.049479931
TLC                               0.0058178899  0.022973938
CREATININE                       -0.1662154019 -0.028712426
UREA                              0.0046458055  0.008946933
EF                               -0.0609899743 -0.049608018
factor(RAISED.CARDIAC.ENZYMES)1   0.3026091094  0.558332218
factor(PRIOR.CMP)1                0.4180764076  0.759222029
```

Figure 14: Logistic Regression model for HEART.FAILURE prediction considering RAISED.CARDIAC.ENZYMES, PRIOR.CMP, GENDER and all the quantitative dependent variables, except PLATELETS.

For the model in **Figure 14.** all the explanatory variables used to create this are significant and without zero in 95% confidence interval of the coefficients.

It was checked if there is some problem of multicollinearity among the explanatory variables by checking Variance Inflation Factor (VIF), but it is possible to see below that only UREA and CREATININE presented moderate collinearity ($2 < VIF < 5$), but not severe ($VIF > 5$) and for this reason it was decided to keep them.

factor(GENDER)	AGE	GLUCOSE	HB
1.139570	1.057903	1.047091	1.253363
TLC	CREATININE	UREA	EF
1.048484	2.285658	2.360371	1.595851
factor(RAISED.CARDIAC.ENZYMES)	factor(PRIOR.CMP)		
1.046516	1.558209		

Figure 15: VIF values for the explanatory variable used in the final logistic regression model – Multicollinearity Check.

It was also checked if some quantitative variables were indicating the probability of heart failure in the final logistic regression model (**Figure 14**) as expected by keeping all the other explanatory variables as means (quantitative ones) and medians (qualitative ones) and separated by gender. For example, the plots presented in the next figure for UREA and EF (the most significant explanatory variables).

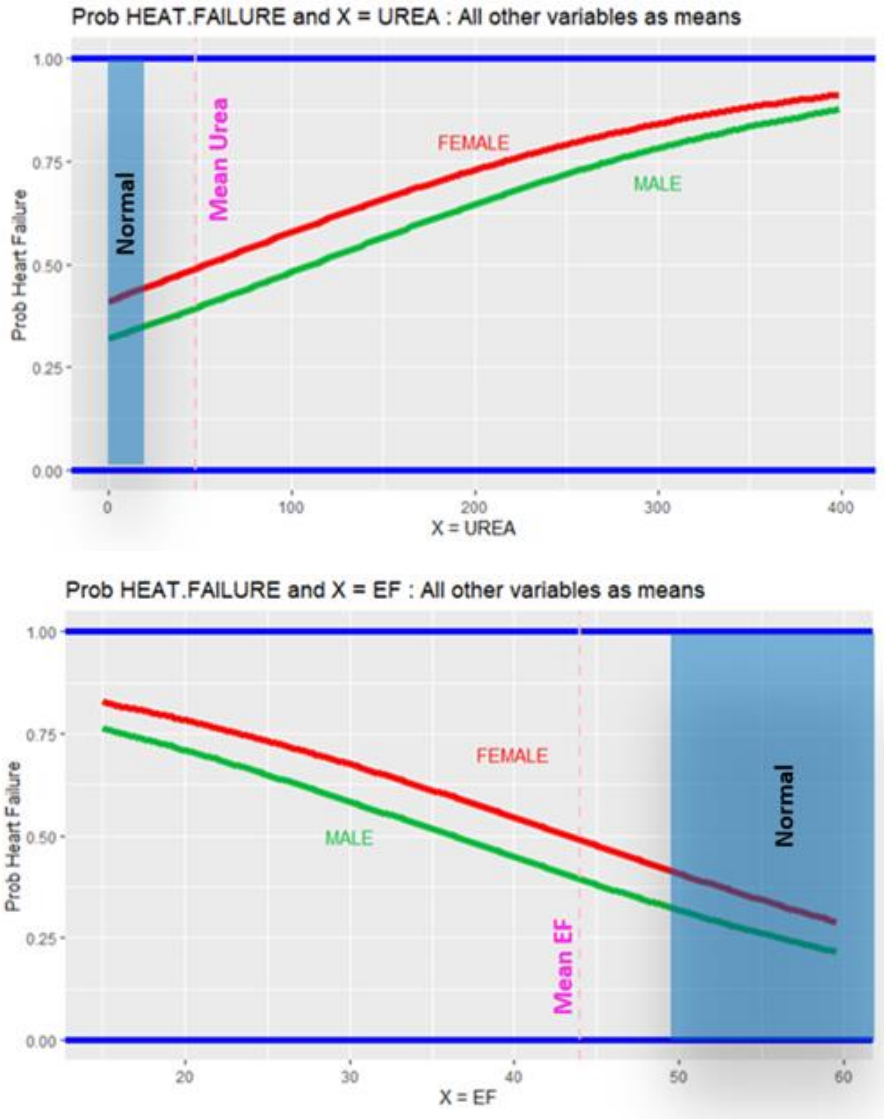


Figure 16: Probability of heart failure in function of UREA and EF indicated by the final Logistic Regression model considering all other variables with means/medians of train part

Figure 16 shows that the probability of heart failure increases with concentration of urea and decreasing with ejection fraction (EF). This is what it would be expected in terms of behavior seeing that they are far from the levels recommended by American Heart Association **[Ref.6]** for people in normal condition.

As everything indicates that the final logistic model indicates properly the behavior of probability of heart failure, it was applied to the test part to verify the prediction performance of this. The confusion matrix and the respective misclassification rate are presented below.

```
                        actual
HEART_FAILURE.predict    0    1
                    0 1636  437      [1] 0.239036
                    1  168  290
```

Figure 17: Confusion table (on the left) and misclassification rate (on the right) obtained after applying the test part in the final logistic regression model of HEART.FAILURE prediction

The confusion table indicates that the final logistic model was possible to predict correctly in the test part 290 and 1636 patients with and without heart failure, respectively. The misclassification rate for this test part was approximately 23.9%.

7.3.1.2 HEART FAILURE: LINEAR DISCRIMINANT ANALYSIS (LDA) -

For the LDA model which has the assumption of normal variance or variance-covariance matrix of each class, it is necessary to check at least if the quantitative variables used in our final logistic regression model are normally distributed. For this check, it was not only plotted histogram and Q-Q plot of each quantitative variable, but also it was done the Kolmogorov=Smirnov and Shapiro-Wilk (just 5000 points for this due to a limitation on this) tests. Based on them, like the results for GLUCOSE and UREA, it was possible to conclude that none of the quantitative variables are normally distributed.

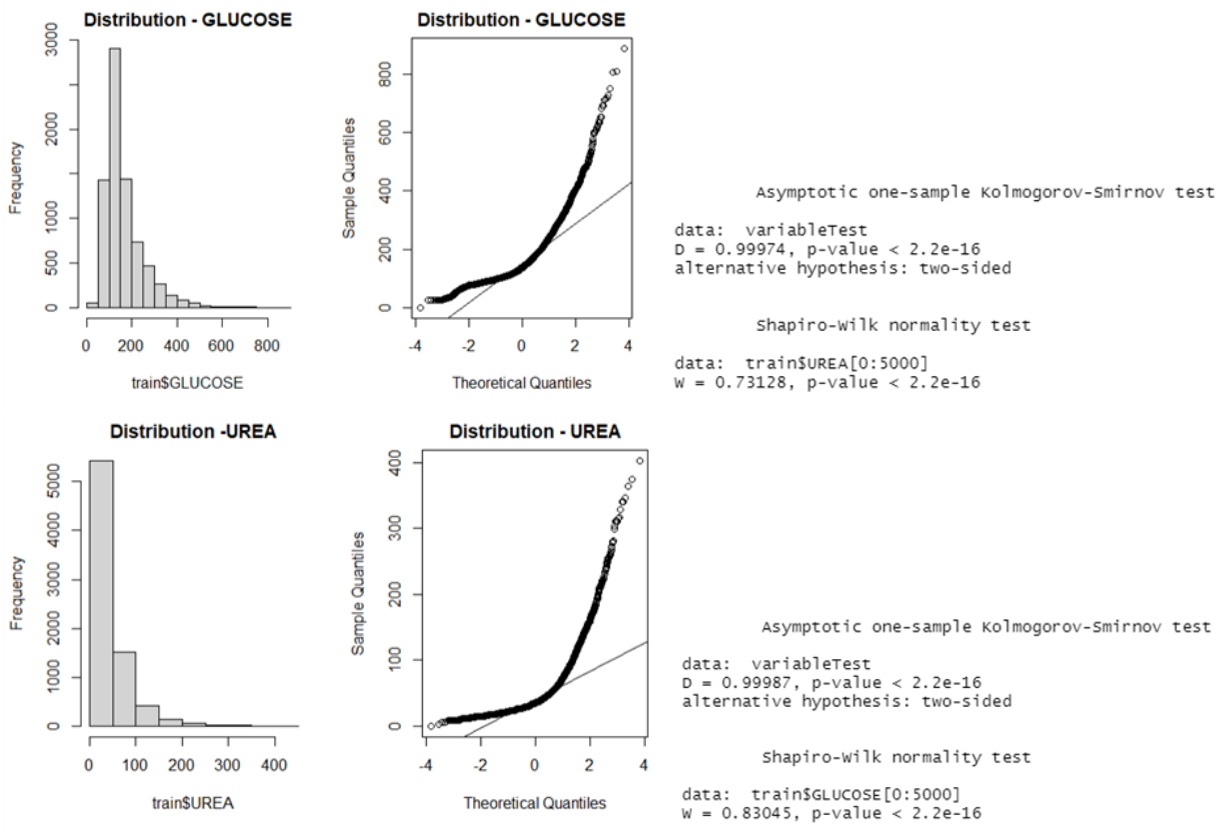


Figure 18: Normality check for the quantitative variables GLUCOSE and UREA

As none of the quantitative variables were normally distributed, it was considered on the LDA modelling only the qualitative ones used in the final logistic regression model. The LDA model generated is presented below.

```
Call:
lda(HEART.FAILURE ~ factor(GENDER) + factor(RAISED.CARDIAC.ENZYMES) +
    factor(PRIOR.CMP), data = train)

Prior probabilities of groups:
        0         1
0.7127996 0.2872004

Group means:
  factor(GENDER)M factor(RAISED.CARDIAC.ENZYMES)1 factor(PRIOR.CMP)1
0       0.6545354                       0.1912064          0.0859043
1       0.5910133                       0.3131591          0.3351674

Coefficients of linear discriminants:
                                    LD1
factor(GENDER)M                -0.498434
factor(RAISED.CARDIAC.ENZYMES)1  1.012301
factor(PRIOR.CMP)1               2.653732
```

Figure 19: LDA model for HEART.FAILURE prediction considering RAISED.CARDIAC.ENZYMES, PRIOR.CMP and GENDER

For the LDA model presented in **Figure 19**, it is possible to verify that the prior probabilities in the train part used to build this were 28.7 and 71.3 % with and without heat failure observations, respectively. In

addition, it also shows that group means. For example, 33.5% of patients with cardiomyopathy (PRIOR.CMP) had heart failure in the train set, whereas only 8.6% of patients with this diagnostic did not have heart failure. As the group means presented what is expected, it was applied the test part to obtain the confusion matrix and the respective misclassification rate are presented below.

```
     0   1
0 1653 493
1  151 234
```

```
[1] 0.2544449
```

Figure 20: Confusion table (on the left) and misclassification rate (on the right) obtained after applying the test part in the LDA model of HEART.FAILURE prediction

The confusion table indicates that the LDA model with only RAISED.CARDIAC.ENZYMES, PRIOR.CMP and GENDER explanatory variables could predict correctly in the test part 234 and 1653 patients with and without heart failure, respectively. The misclassification rate for this test part with this model was approximately 25.4%.

7.3.1.3 HEART FAILURE: QUADRATIC DISCRIMINANT ANALYSIS (QDA)

For the QDA statistical learning method which is not strict to the normality of the variables, it was considered all the explanatory variables used logistic regression when QDA model was as shown in the next figure.

```
Call:
qda(HEART.FAILURE ~ factor(GENDER) + AGE + GLUCOSE + HB + TLC +
    CREATININE + UREA + EF + factor(RAISED.CARDIAC.ENZYMES) +
    factor(PRIOR.CMP), data = train)

Prior probabilities of groups:
        0         1
0.7127996 0.2872004

Group means:
  factor(GENDER)M      AGE  GLUCOSE       HB      TLC CREATININE     UREA       EF factor(RAISED.CARDIAC.ENZYMES)1 factor(PRIOR.CMP)1
0       0.6545354 60.04545 157.8815 12.57658 11.17966   1.200482 42.06529 47.29012                          0.1912064          0.0859043
1       0.5910133 63.93994 180.6666 11.79837 12.76730   1.564241 61.28317 35.85282                          0.3131591          0.3351674
```

Figure 21: QDA model for HEART.FAILURE prediction considering RAISED.CARDIAC.ENZYMES, PRIOR.CMP, GENDER and all the quantitative dependent variables, except PLATELETS

For the QDA model presented in **Figure 21**, it is possible to verify the same prior probabilities of HEART.FAILURE as LDA seeing that it was used the same train part. In addition, it also shows that group means not only for the categorical explanatory variables like LDA, but also the group means of the quantitative explanatory variables. For example, patients with heart failure in train set has higher UREA than patients without heart failure on average that is compatible with what was presented in the logistic regression.

For the QDA, it is presented the partition plot for the pairs of variables so that it is possible to verify the distribution of train units together the respective decision boundary of this model.

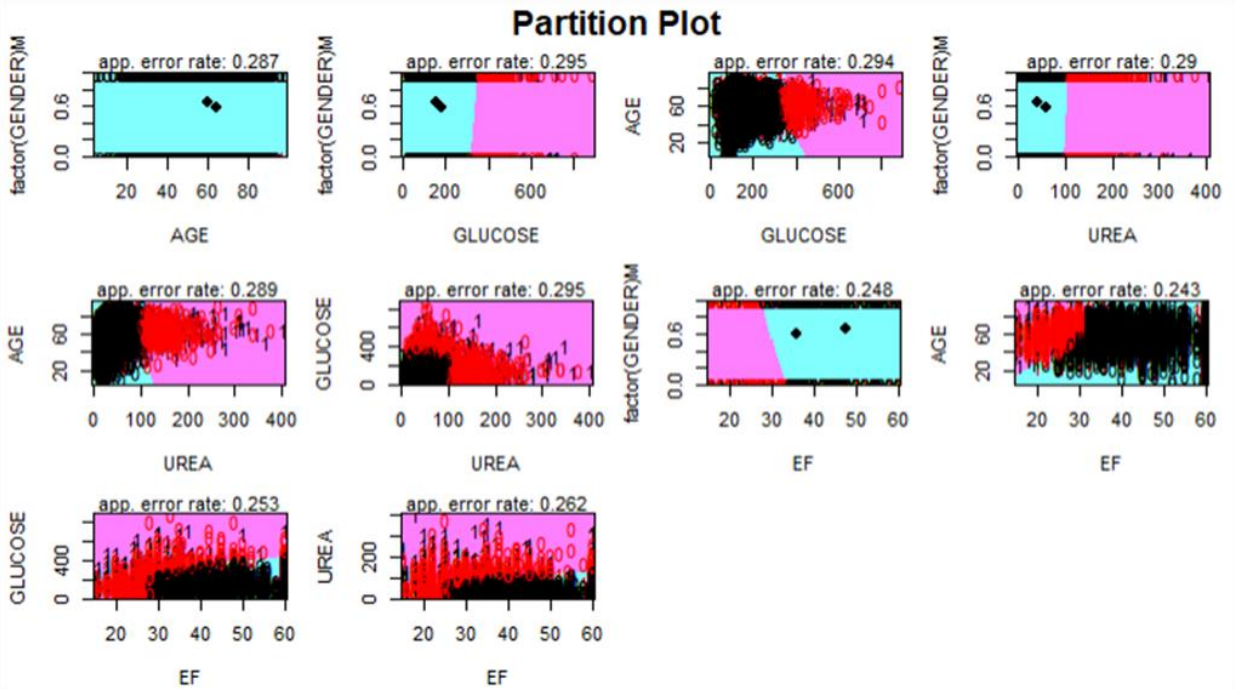


Figure 22: Partition Plot of variables used in QDA model for HEART.FAILURE prediction

In terms of performance of the QDA model after applying the test part, the confusion matrix and the respective misclassification rate are presented below.

```
HEART.pred    0    1
         0 1546  402
         1  258  325
```

```
[1] 0.2607665
```

Figure 23: Confusion table (on the left) and misclassification rate (on the right) obtained after applying the test part in the QDA model of HEART.FAILURE prediction.

The confusion table indicates that the QDA model could predict correctly in the test part 325 and 1546 patients with and without heart failure, respectively. The misclassification rate for this test part with this model was approximately 26.1%.

7.3.1.4 HEART FAILURE: CLASSIFICATION TREE

The next statistical learning method named Classification Tree was used to create a model to predict HEART.FAILURE with the same explanatory variables used in Logistic Regression and QDA. For tree creation based on train, it was used “tree” function in “tree” package as shown in next figure.

```
Classification tree:
tree(formula = HEART.FAILURE ~ factor(GENDER) + AGE + GLUCOSE +
    HB + TLC + CREATININE + UREA + EF + factor(RAISED.CARDIAC.ENZYMES) +
    factor(PRIOR.CMP), data = train)
Variables actually used in tree construction:
[1] "EF"   "UREA"
Number of terminal nodes:  4
Residual mean deviance:  1.019 = 7735 / 7590
Misclassification error rate: 0.2447 = 1858 / 7594
```

Figure 24: Classification Tree model for HEART.FAILURE prediction considering RAISED.CARDIAC.ENZYMES, PRIOR.CMP, GENDER and all the quantitative dependent variables, except PLATELETS.

Figure 24 shows that only two explanatory variables (UREA and EF) were used to build the three with the total of four terminal nodes and the misclassification rate based on train part of approximately 24.5%. The schematic of this tree with the respective rules in the split nodes is presented in the next figure.

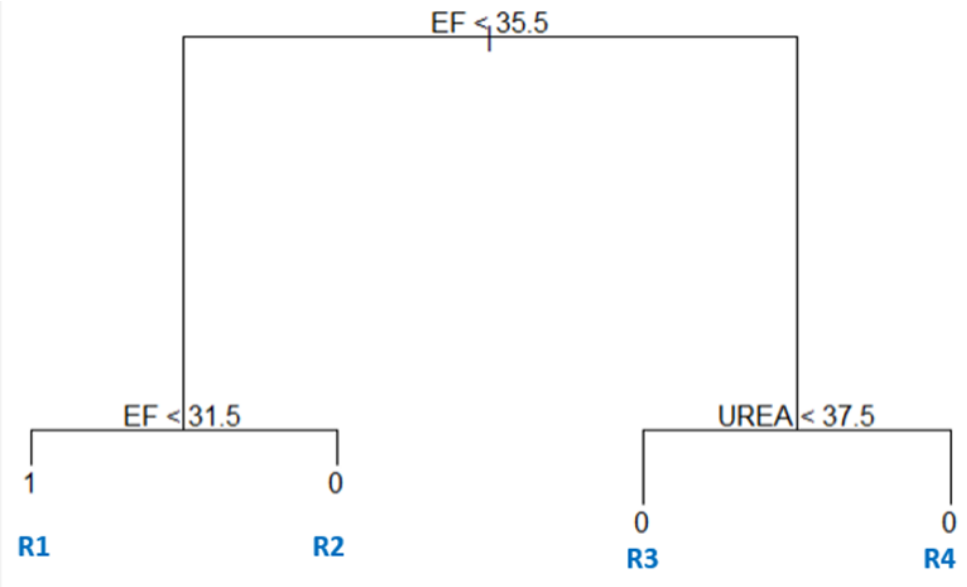


Figure 25: Tree for HEART.FAILURE prediction

The tree presented in **Figure 25** presented at the top the first rule related to EF. If this value is lower than 35.5, there are two possible predictions: the predicts heart failure (region R1) if EF < 31.5, else predicts no heart failure (region R2). On the right side of the top of the tree (EF >= 35.5), it only predicts no heart failure. However, UREA lower than 37.5 goes to region R3, whereas other value of UREA goes to region R4. The tree is divided in this way because for Classification Tree is not only important to predict, but also to know the probabilities of each class in the nodes. For this reason, it is shown in the next figure the probabilities of the constructed tree.

```
node), split, n, deviance, yval, (yprob)
      * denotes terminal node

1) root 7594 9107 0 ( 0.7128 0.2872 )
  2) EF < 35.5 2605 3604 1 ( 0.4737 0.5263 )
    4) EF < 31.5 1593 2142 1 ( 0.3986 0.6014 ) *R1
    5) EF > 31.5 1012 1369 0 ( 0.5919 0.4081 ) *R2
  3) EF > 35.5 4989 4426 0 ( 0.8376 0.1624 )
    6) UREA < 37.5 3225 2185 0 ( 0.8936 0.1064 ) *R3
    7) UREA > 37.5 1764 2039 0 ( 0.7353 0.2647 ) *R4
```

Figure 26: Probabilities in the nodes for the constructed tree for HEART.FAILURE prediction

Figure 26 shows that the probabilities of correctly predicting no heart failure in regions R3 (mainly) and R4 is higher than 70% based on train set, but the other regions the differences of class probabilities are not too large, so it is more difficult to predict correctly. It is possible to verify this in the next plot.

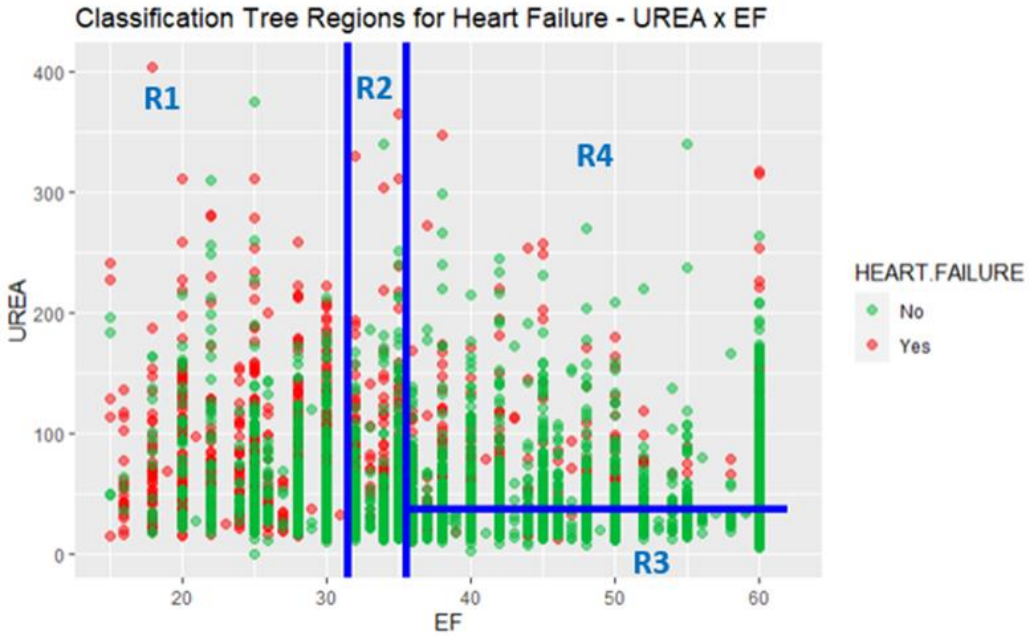


Figure 27: UREA versus EF in the train together the tree regions

Regarding the performance of the classification tree model with four terminal nodes after applying the test part, the confusion matrix and the respective misclassification rate are presented below.

HEART_tree.pred	0	1
0	1605	421
1	199	306

```
[1] 0.2449625
```

Figure 28: Confusion table (on the left) and misclassification rate (on the right) obtained after applying the test part in the classification tree model of HEART.FAILURE prediction

The confusion table indicates that the classification tree model could predict correctly in the test part 306 and 1605 patients with and without heart failure, respectively. The misclassification rate for this test part with this model was approximately 26.1%.

In terms of pruning the tree, it was decided not to do this seeing that the obtained tree was not complex.

7.3.1.5 HEART FAILURE: SUMMARY

This item shows the summary results of the models created to predict HEART.FAILURE not only using stratified sampling (75% train part, 25% test part), but also with 10-fold Stratified Cross-validation, which the results in term of misclassification rate are presented in **Figure 29** considering the following relevant explanatory variables:

- Quantitative: AGE, GLUCOSE, HB, TLC, CREATININE, UREA and EF
- Qualitative: GENDER, RAISED.CARDIAC.ENZYMES and PRIOR.CMP

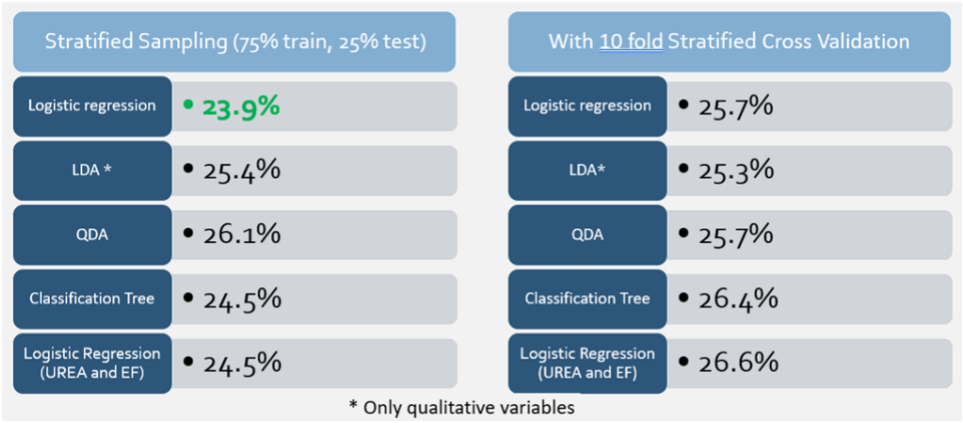


Figure 29: HEART. FAILURE prediction models Summary: Misclassification Rate comparison

Based on the results above, the Logistic Regression model with stratified sampling indicated the lowest misclassification rate (23.9%) among all the generated models to predict HEART.FAILURE. However, LDA (with only qualitative variables due to no normality of distribution of any quantitative variable required to this statistical learning method) that indicated the best performance (misclassification rate of 25.3%) among all the methods considering in the generation 10-fold stratified cross-validation.

It is also important to mention that QDA models that predicted in test/validation part more correctly the heart failure among all the models, whereas LDA which one that predicted more correctly no heart failure.

7.3.2 CATEGORICAL RESPONSE VARIABLE ACUTE KIDNEY INJURY (AKI)

For Acute Kidney Injury (AKI) as a response variable, some preliminary initial analysis was performed with LOGISTIC REGRESSION considering all the dependent quantitative variables and with selected STEMI and DM categorical variables indicated by the independence test presented in item 7.2. This selection of categorical explanatory variables was considered based on the lowest misclassification rate.

During this preliminary analysis, we applied first the statistical learning method to identify the significant ones by observing the p-values of each variable. This analysis helped us to remove the not significant ones. After that, we kept the same variables for all the other model analyses.

7.3.2.1 AKI: LOGISTIC REGRESSION

We created a logistic regression model with and without 10 k-fold cross-validations (CV) to compare the results with each other.

Initially, we performed sampling in the cleaned dataset considering 75% train and 25% test set. Then Logistic regression model was created with a 75% train set considering quantitative and qualitative variables.

```
Call:
glm(formula = AKI ~ factor(GENDER) + AGE + GLUCOSE + HB + TLC +
    PLATELETS + UREA + CREATININE + EF + factor(STEMI) + factor(DM),
    family = binomial, data = train_AKI)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-1.91423  -0.00001   0.00000   0.00000   2.75900

Coefficients:
                  Estimate Std. Error z value Pr(>|z|)
(Intercept)     -7.163e+01  6.876e+00 -10.418  < 2e-16 ***
factor(GENDER)M  2.010e-01  3.574e-01   0.562  0.57381
AGE             -1.178e-02  1.381e-02  -0.853  0.39360
GLUCOSE         -3.506e-03  1.587e-03  -2.210  0.02714 *
HB               4.864e-02  7.683e-02   0.633  0.52668
TLC              1.104e-02  1.380e-02   0.800  0.42345
PLATELETS       -4.668e-04  1.624e-03  -0.288  0.77372
UREA            -1.840e-03  6.666e-03  -0.276  0.78254
CREATININE       4.861e+01  4.480e+00  10.849  < 2e-16 ***
EF              -7.922e-03  1.127e-02  -0.703  0.48209
factor(STEMI)1   2.676e-01  4.060e-01   0.659  0.50979
factor(DM)1      9.513e-01  3.399e-01   2.798  0.00514 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 7840.97  on 7593  degrees of freedom
Residual deviance:  299.23  on 7582  degrees of freedom
AIC: 323.23

Number of Fisher Scoring iterations: 13
```

Figure 30: Initial logistic model for AKI

From the initial model output, we see GLUCOSE, CREATININE, and DM variables are significant only. Next, we created the final fitted model by removing all non-significant variables and keeping only these three.

```
Call:
glm(formula = AKI ~ CREATININE + GLUCOSE + factor(DM), family = binomial,
    data = train_AKI)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-1.80962  -0.00001   0.00000   0.00000   2.79040

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -71.102440   6.553882 -10.849  < 2e-16 ***
CREATININE   48.002438   4.397677  10.915  < 2e-16 ***
GLUCOSE      -0.003436   0.001510  -2.275  0.02290 *
factor(DM)1   0.856600   0.317553   2.698  0.00699 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 7840.97  on 7593  degrees of freedom
Residual deviance:  304.16  on 7590  degrees of freedom
AIC: 312.16

Number of Fisher Scoring iterations: 13
```

Figure 31: Re-fitted model with significant variables only for AKI

We also checked multicollinearity on those variables and found all VIF values are less than 2. Thus, we can say there is no multicollinearity exists between these variables.

```
CREATININE     GLUCOSE factor(DM)
  1.026479    1.119551    1.133526
```

Figure 32: Multicollinearity checking for AKI

Next, we applied the test part to the fitted logistic regression model to predict the “1” and “0” which indicate patients have kidney injury and don't have acute kidney injury (AKI) respectively.

Prediction based on Test part (25%)

```
            actual_AKI
AKI.predict    0    1
          0 1973    4
          1   22  532
```

Misclassification rate

```
[1] 0.01027262
```

Figure 33: Confusion table and misclassification rate without CV for logistic regression of AKI

The confusion table indicates 1973 patients do not have AKI (true negative) whereas 4 false positives and 532 patients have AKI (true positive) whereas 22 false negatives.

After calculation, we found the misclassification ratio is 1.03%, in other words, 98.97% were predicted correctly in the test part.

To compare, we also built a model with 10 k-fold CV and analysis the outcome where we found the misclassification rate is 1.11% which is similar than the result we obtained without CV.

We have created a plot of "Prob AKI vs CREATININE and taking All other variables as means" to see the trends of probability of AKI.

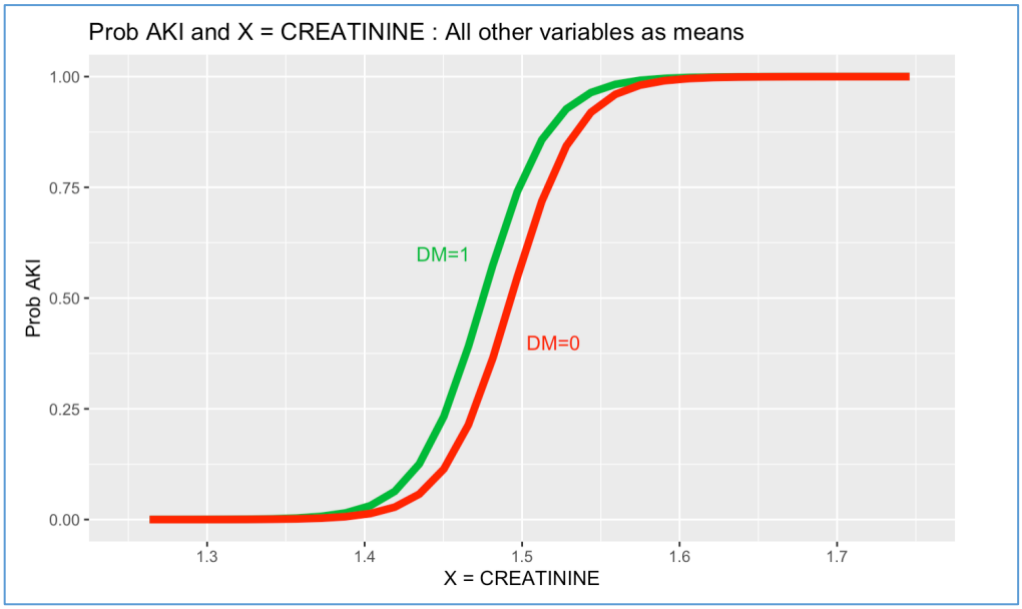


Figure 34: Plot for Prob AKI vs CREATININE and taking All other variables as means

Here we see that in terms of CREATININE, the probability of having AKI is slightly different between having or no DM. “0” probability is expected for values of CREATININE lower than approximately 1.35 whereas 100% of probability of AKI for values approximately higher than 1.6. Between these values, the probability follows this curve. The value obtained here assumed the mean of CREATININE in the train part.

7.3.2.2 AKI: LINEAR DISCRIMINANT ANALYSIS (LDA)

We tried to build an LDA model with and without 10 k-fold cross validation (CV) to see the results.

To build the LDA model, we need to fulfil the assumption that the variables need to be normally distributed. Therefore, we checked the normality of the quantitative variables and found all are not normally distributed as the p-value is less than 0.05, which rejects the null hypothesis. So, we exclude all quantitative variables. The below figure shows the normality test for one of the quantitative variables’ creatinine.

```
        One-sample Kolmogorov-Smirnov test

data:  train_AKI$CREATININE
D = 0.67724, p-value < 2.2e-16
alternative hypothesis: two-sided


        Shapiro-Wilk normality test

data:  train_AKI$CREATININE[0:5000]
W = 0.97627, p-value < 2.2e-16
```

Figure 35: Normality test for Creatinine

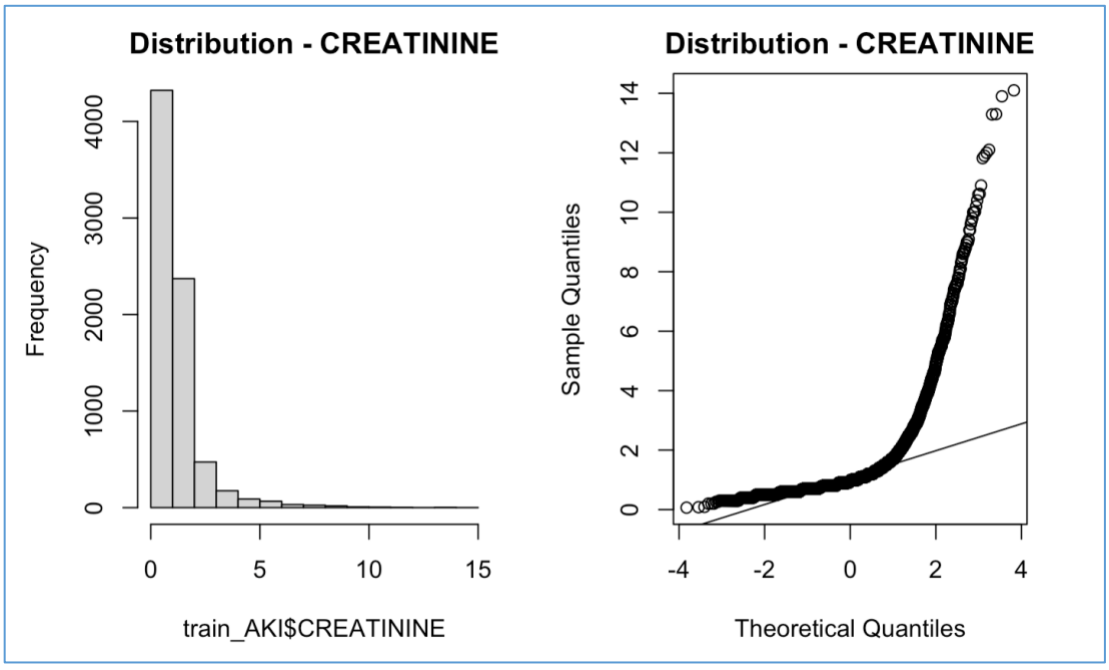


Figure 36: Normality Plot for Creatinine

As among our significant variables, both creatinine and glucose quantitative variables are not normally distributed, we cannot make the LDA model with only one categorical variable DM. Therefore, the LDA model is not valid for this case.

However, we tried to build an LDA model with suggested models with the categorical variables indicated by the independence test presented in item 7.2. So, we considered one of the suggested models with categorical variables STEMI and DM to build the LDA model and do further analysis to see the result of the misclassification rate. We tried another suggested model with variables RAISED.CARDIAC.ENZYMES and PRIOR.CMP but found misclassification rate is higher than this.

```
Call:
lda(AKI ~ factor(STEMI) + factor(DM), data = train_AKI)

Prior probabilities of groups:
        0         1
0.7882539 0.2117461

Group means:
  factor(STEMI)1 factor(DM)1
0      0.1772469   0.2943535
1      0.1293532   0.4465174

Coefficients of linear discriminants:
                       LD1
factor(STEMI)1 -0.983984
factor(DM)1     2.002377
```

Figure 37: LDA model with categorical variables for AKI

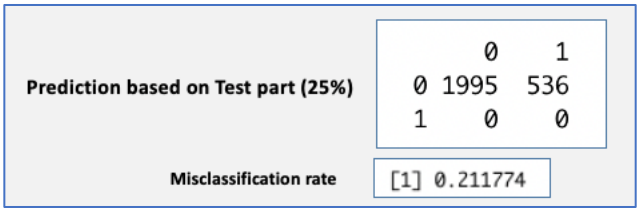


Figure 38: Confusion table and misclassification rate without CV for LDA of AKI

The confusion table indicates, in terms of "0" or true negative, patients do not have AKI (true negative) whereas 536 false positives. In terms of " 1", none predicted true positive and false negative, and the misclassification rate is 21.2%.

To compare, we also built a model with a 10 k-fold CV and analyzed the outcome where we found the misclassification rate is the same.

7.3.2.3 AKI: QUADRATIC DISCRIMINANT ANALYSIS (QDA)

We build the QDA model with and without 10 k-fold cross-validation (CV) to compare the results. QDA analysis was performed considering significant variables only that is Creatinine, Glucose, and DM. For the model built without CV, we found QDA model output indicates that 78.8% of the training observations are patients who are not having ACUTE KIDNEY INJURY and 21.2 % represent those that are having ACUTE KIDNEY INJURY. It also provided the group means of each variable.

```
Call:
qda(AKI ~ CREATININE + GLUCOSE + factor(DM), data = train_AKI)

Prior probabilities of groups:
        0         1
0.7882539 0.2117461

Group means:
  CREATININE  GLUCOSE factor(DM)1
0   0.893588 159.5743   0.2943535
1   2.827289 180.4093   0.4465174
```

Figure 39: AKI QDA model

The confusion table indicates 1987 patients do not have AKI (true negative) whereas 39 false positives and 497 patients have AKI (true positive) whereas 8 false negatives.

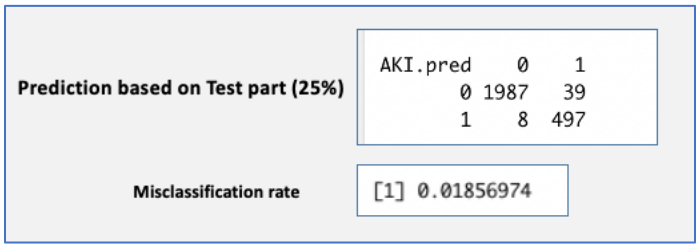


Figure 40: Confusion table and misclassification rate without CV for QDA of AKI

Lastly, we calculate the misclassification rate which is 0.01856974 (1.86%), in other words, 98.14% were predicted correctly.

We have drawn the partition plot to identify the lowest error rate for the associated variables. Here we included the STEMI category variable with other significant ones due to the LDA model built considering STEMI and to observe its error rate with others.

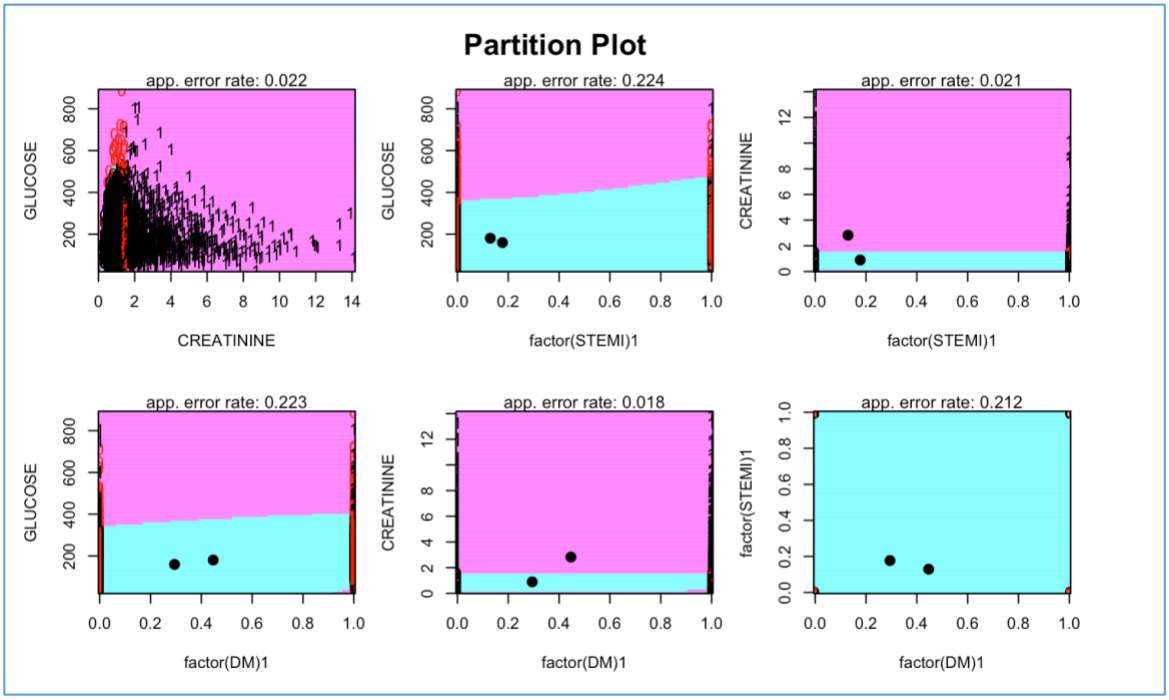


Figure 41: QDA Partition Plot for AKI

From the partition plot, we see that Creatinine and DM are showing the lowest error rate which is 0.018.

We tried redoing the QDA model with only these two variables to see the result and we found the misclassification rate is 0.01738443 (1.72%) which is slightly improved from the actual QDA model rate of 1.86%.

To compare, we also built a model with a 10 k-fold CV and analyzed the outcome where we found the misclassification rate is 2.34% which is higher than the result we obtained without a CV.

7.3.2.4 AKI: CLASSIFICATION TREE

We have built a classification tree model with and without 10 k-fold cross-validation (CV) to see the results.

For the model built without CV, we found the classification model only selected creatinine variable to construct the tree and the total number of terminal nodes selected by the model is 3.

```
Classification tree:
tree(formula = AKI ~ CREATININE + GLUCOSE + factor(DM), data = train_AKI)
Variables actually used in tree construction:
[1] "CREATININE"
Number of terminal nodes:  3
Residual mean deviance:  0.03923 = 297.8 / 7591
Misclassification error rate: 0.0129 = 98 / 7594
```

Figure 42: AKI classification tree model

Further, we plotted the tree for better visualization and due to only 3 terminal nodes, we did not prune the tree as this shows the optimal view/result for the prediction.

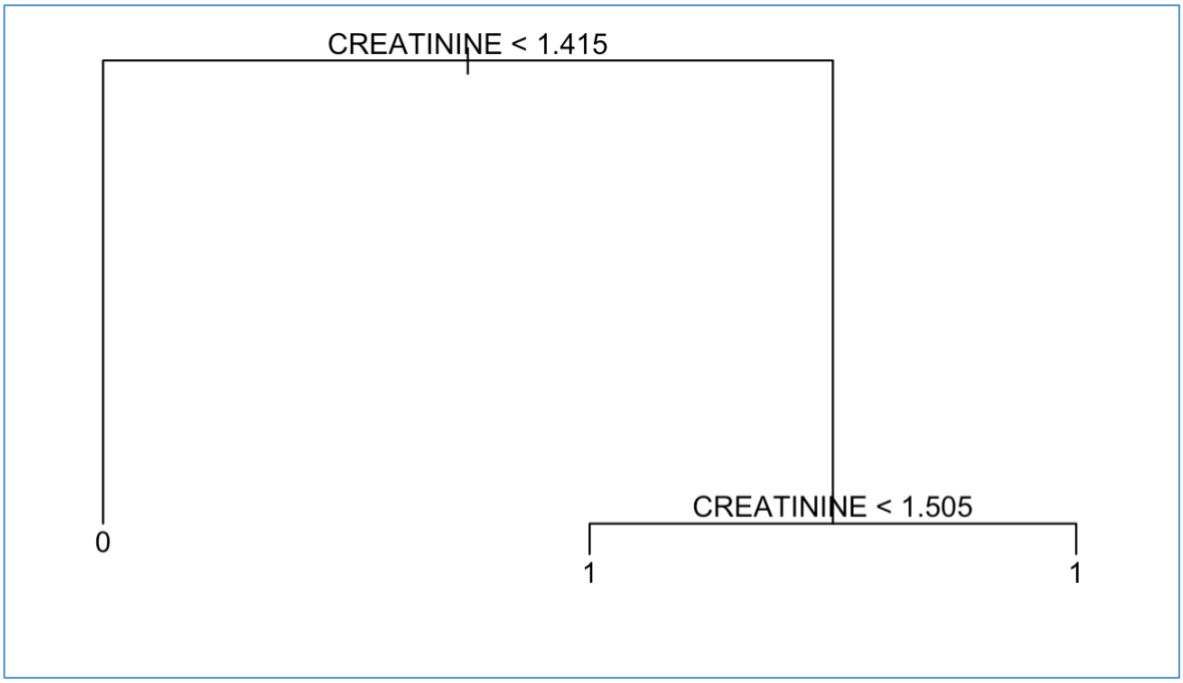


Figure 43: AKI classification Tree plot

We have identified the probabilities in each terminal node as follows:

```
node), split, n, deviance, yval, (yprob)
      * denotes terminal node

1) root 7594 7841.00 0 ( 0.7882539 0.2117461 )
  2) CREATININE < 1.415 5890   19.36 0 ( 0.9998302 0.0001698 ) *
  3) CREATININE > 1.415 1704  744.40 1 ( 0.0569249 0.9430751 )
    6) CREATININE < 1.505 201  278.40 1 ( 0.4825871 0.5174129 ) *
    7) CREATININE > 1.505 1503    0.00 1 ( 0.0000000 1.0000000 ) *
```

Figure 44: AKI probability chart

From the above probability of tree classification, we can identify the probability of each terminal node.

Here we see that, in the case of CREATININE > 1.505 where purity has reached 100% of ”1" as predicted whereas CREATININE < 1.505 shows almost similar probability of 48% and 51% for both ”0” and ”1. Consequently, the majority of wrong predictions may happen there. For the left side of the tree where creatinine <1.4, the difference in probability is also significant, the “0” probability is showing 99.98% so the probability of predicting ”0" is very high.

The confusion table indicates 1962 patients do not have AKI (true negative) whereas 0 false positives and 536 patients have AKI (true positive) whereas 33 false negatives.

Prediction based on Test part (25%)

```
AKI_tree.pred    0    1
            0 1962    0
            1   33  536
```

Misclassification rate

```
[1] 0.01303832
```

Figure 45: Confusion table and misclassification rate without CV for classification tree of AKI

Lastly, we calculated the misclassification rate and found 1.3% in other words, 98.7% were predicted correctly.

Also, we have drawn the classification tree regions for Creatinine vs Glucose along with AKI for better visualization.

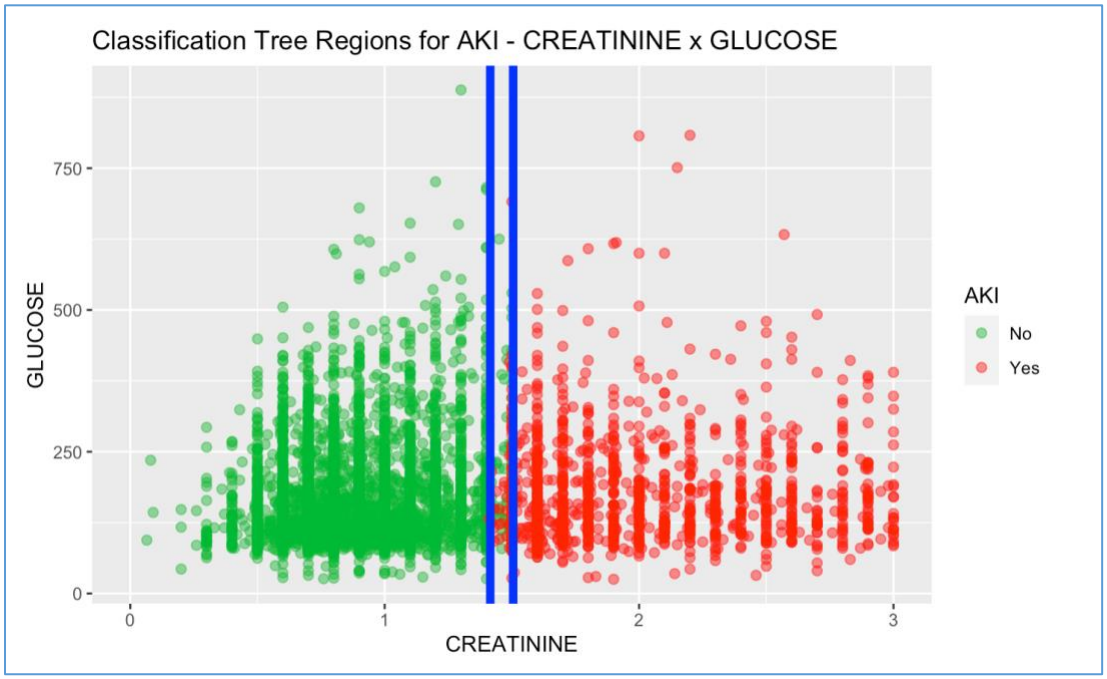


Figure 46: Scattered plot for classification tree regions of AKI

From this plot, we see that patients who have AKI that is “1” and who do not have AKI that is “0”, are well divided into two regions. Both have and do not have AKI data merged within the blue line where the creatinine value is around 1.5 and most of the wrong predictions happened here.

To compare, we also built a model with a 10 k-fold CV and analyzed the outcome where we found the misclassification rate is 1.33% which is almost similar to the result we obtained without CV.

7.3.2.5 AKI: SUMMARY

This item shows the summary results of the different models created to predict AKI not only using stratified sampling (75% train part, 25% test part) but also with a 10 k-fold Stratified Cross-validation, which the results in terms of misclassification rate are presented below in **Figure 47** considering the following relevant explanatory variables:

- Quantitative: GLUCOSE, and CREATININE
- Qualitative: STEMI and DM

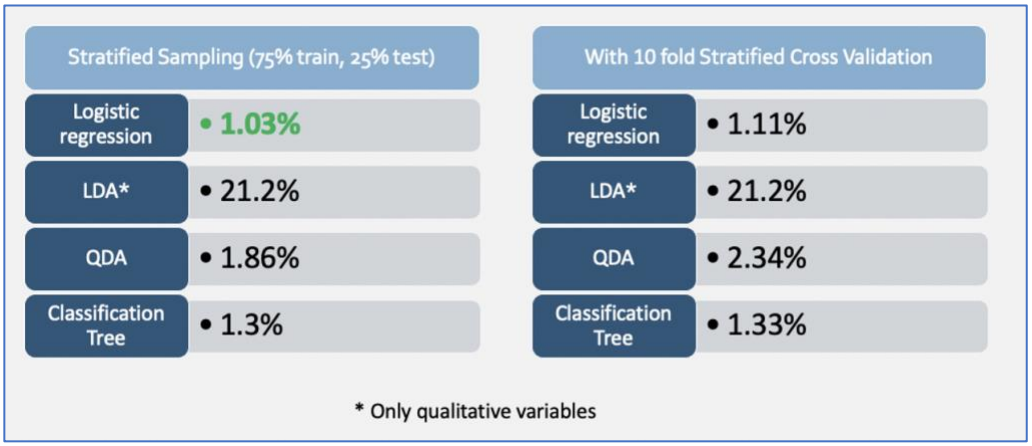


Figure 47: AKI prediction models Summary: Misclassification Rate comparison

Based on the results above, the Logistic Regression model without stratified sampling indicated the lowest or best misclassification rate of 1.03% among all the generated models to predict AKI. However, we did not observe significant differences with logistic regression with a 10 k-fold stratified cross-validation and classification tree with and without cross-validation.

7.3.3 QUANTITAIVE RESPONSE VARIABLE (DURATION OF STAY)

Another question that we were trying to answer with our analysis was, is it possible to predict the "DURATION.OF.STAY" of patients at the hospital based on their medical laboratory results and gender. The laboratory results were mainly quantitative variables, while the gender was the only qualitative variable considered in the analysis. The figure below shows the first pass using these independent variables for analysis.

```
linearModel1 <- lm(DURATION.OF.STAY~factor(GENDER)+AGE+HB+TLC+PLATELETS+GLUCOSE+UREA+CREATININE, data = dataClean)

summary(linearModel1)
```

```
Call:
lm(formula = DURATION.OF.STAY ~ factor(GENDER) + AGE + HB + TLC +
    PLATELETS + GLUCOSE + UREA + CREATININE, data = dataClean)

Residuals:
    Min      1Q  Median      3Q     Max
-15.280  -2.801  -0.844   1.529  89.521

Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)    6.1511961  0.4271455  14.401  < 2e-16 ***
factor(GENDER)M 0.2257423  0.1026968   2.198    0.028 *
AGE            0.0164249  0.0036104   4.549 5.44e-06 ***
HB            -0.2430356  0.0228688 -10.627  < 2e-16 ***
TLC            0.0714295  0.0067349  10.606  < 2e-16 ***
PLATELETS      0.0001566  0.0004690   0.334    0.738
GLUCOSE        0.0037358  0.0005517   6.772 1.34e-11 ***
UREA           0.0156683  0.0018748   8.357  < 2e-16 ***
CREATININE     0.0510935  0.0604433   0.845    0.398
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 4.657 on 10116 degrees of freedom
Multiple R-squared:  0.07289,   Adjusted R-squared:  0.07216
F-statistic: 99.41 on 8 and 10116 DF,  p-value: < 2.2e-16
```

Figure 48: Linear Regression using all quantitative laboratory measurements and “GENDER” to model “DURATION.OF.STAY”

From Figure 1 above, the high P-value for “GENDER”, “PLATELETS and “CREATININE” means that we fail to reject the null hypothesis that states that their coefficients are not significant since these P-values are greater than p-value = 0.05.

We also looked at the relationships between “DURATION.OF.STAY” and other independent variables. Atypical example is shown in Figure 2 below:

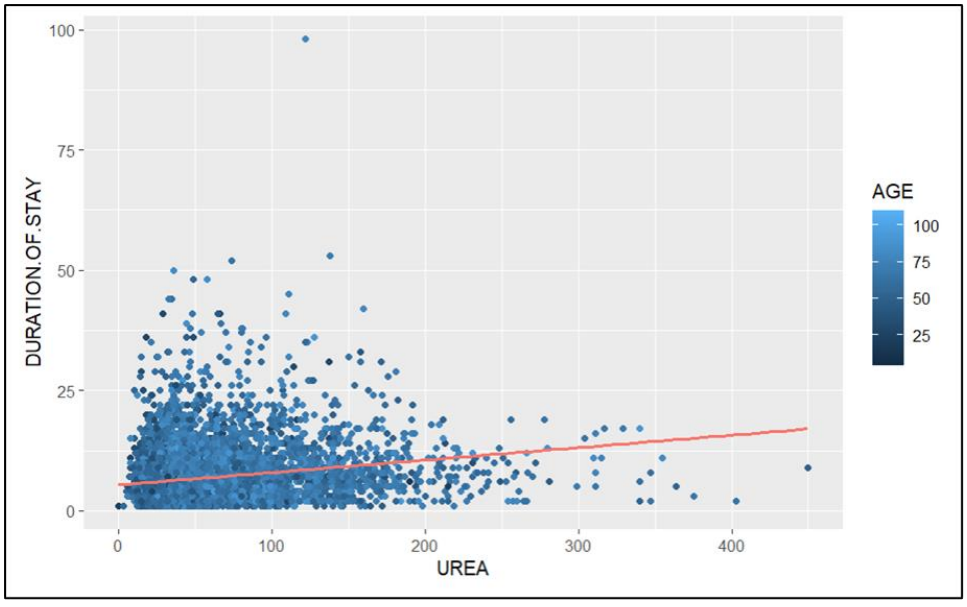


Figure 49: Plot of the dependent variable “DURATION.OF.STAY” vs UREA

As can be observed, a strong linear relationship between the dependent variable and the independent variable was not evident in this case. This is also similar to the relationship with other independent variables.

This is one of the reasons why the adjusted r-squared is low. At 0.072, it implies that our linear model can only account for seven percent (7%) of the variation of the response variable "DURATION.OF.STAY". New linear models were tested after removing the three insignificant variables. The new models did not show any significant improvement in the adjusted r-squared. The figure below highlights this fact.

```
Call:
lm(formula = DURATION.OF.STAY ~ AGE + HB + TLC + GLUCOSE + UREA,
    data = dataClean)

Residuals:
    Min      1Q  Median      3Q     Max
-15.205  -2.793  -0.846   1.525  89.332

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  6.1874430  0.3926364  15.759  < 2e-16 ***
AGE          0.0163724  0.0036028   4.544 5.58e-06 ***
HB          -0.2304180  0.0214050 -10.765  < 2e-16 ***
TLC          0.0716079  0.0066435  10.779  < 2e-16 ***
GLUCOSE      0.0036914  0.0005513   6.696 2.25e-11 ***
UREA         0.0170265  0.0013236  12.863  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4.658 on 10119 degrees of freedom
Multiple R-squared:  0.07232,   Adjusted R-squared:  0.07186
F-statistic: 157.8 on 5 and 10119 DF,  p-value: < 2.2e-16
```

Figure 50: Linear regression with DURATION OF STAY as response variable

From above, we then tested for the linear regression assumptions to see if our variables meet these criteria. The first was to correlate between the independent variables to see the kind of relationships to consider.

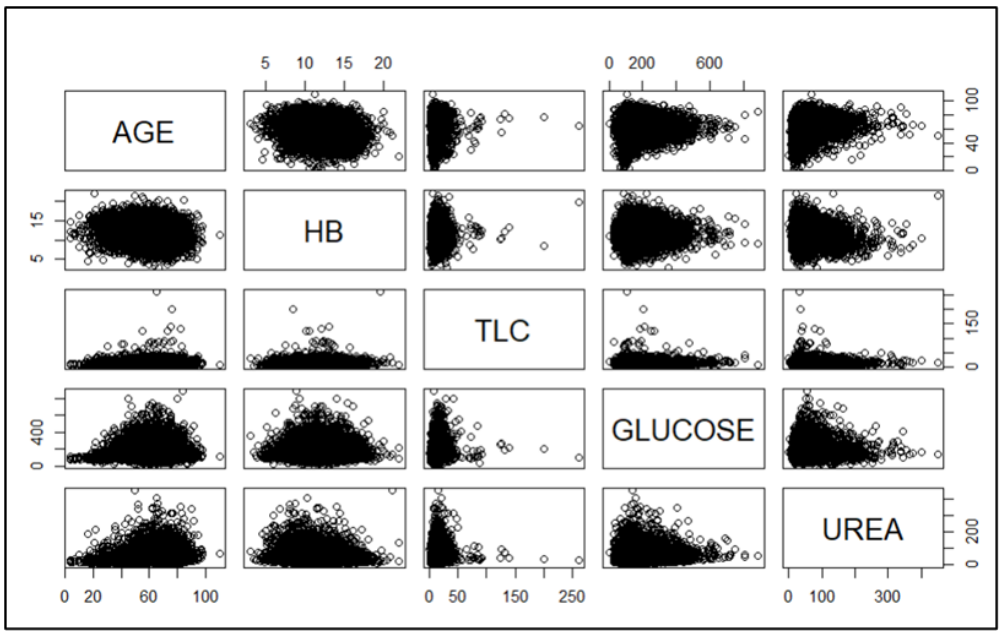


Figure 51: Correlation between independent variables for quantitative variables

As can be seen above, there are no strong linear relationships between the individual independent variables.

We also tested for multi-collinearity using the VIF method. This test did not show any form of multicollinearity as shown below.

```
Call:
imcdiag(mod = linearModel3, method = "VIF")


 VIF Multicollinearity Diagnostics

           VIF detection
AGE     1.0690         0
HB      1.1435         0
TLC     1.0400         0
GLUCOSE 1.0412         0
UREA    1.1829         0

NOTE:  VIF Method Failed to detect multicollinearity


0 --> COLLINEARITY is not detected by the test

===================================
```

Figure 52: VIF Test for multicollinearity for quantitative variables

Following the above check, we now tested for homostedasticity (or heterostedasticity is not present). To check this, we first plotted the residuals against fitted values as shown below.

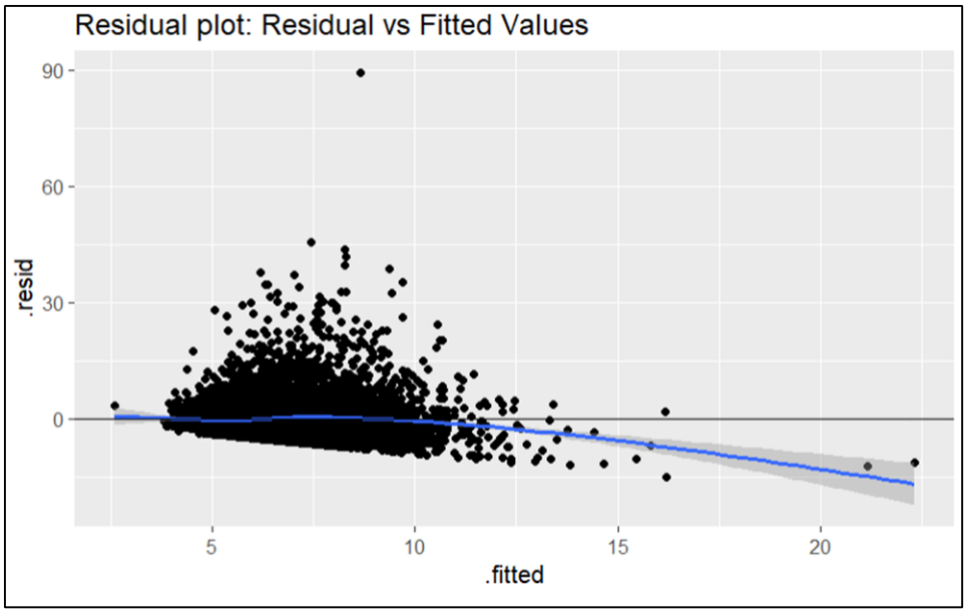


Figure 53: Linearmodel3 Residuals Vs Fitted Values plot

The above plot shows some spread in the residuals. Even the Scale-Location plot below showed some scatter in the residuals, which implies that there is heteroscedasticity present in the data.

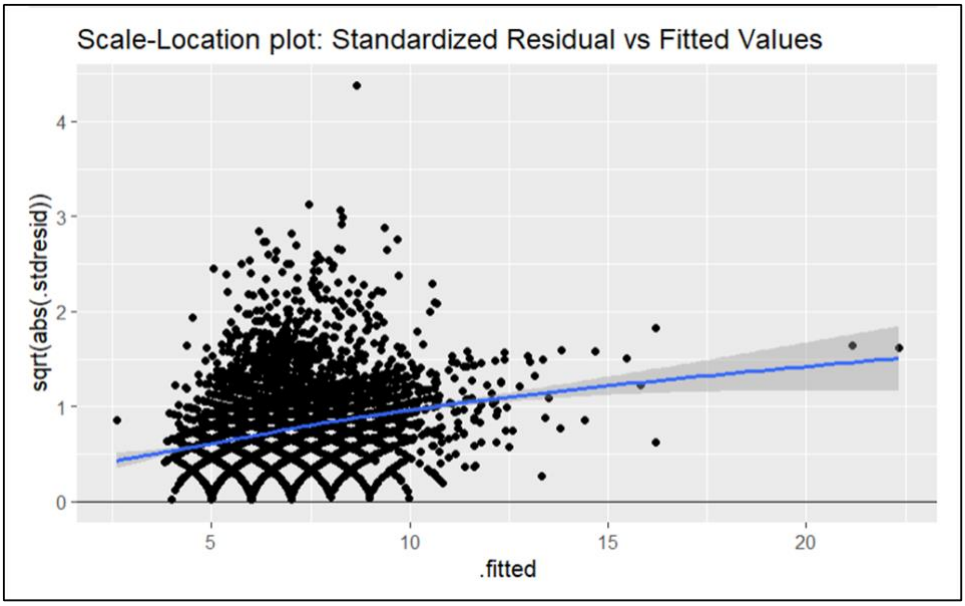


Figure 54: Linearmodel3 Scale-Location plot of Residuals Vs Fitted Values

Our postulation was further confirmed with the Breusch-Pagan test that computed a P-value of (p-value < 2.2e-16) which clearly indicates that we should reject the null hypothesis that heteroscedasticity does not exist.

The next test we carried out was the check for normality test. The first thing we did was to check the the Q-Q plot of the residuals as shown below.

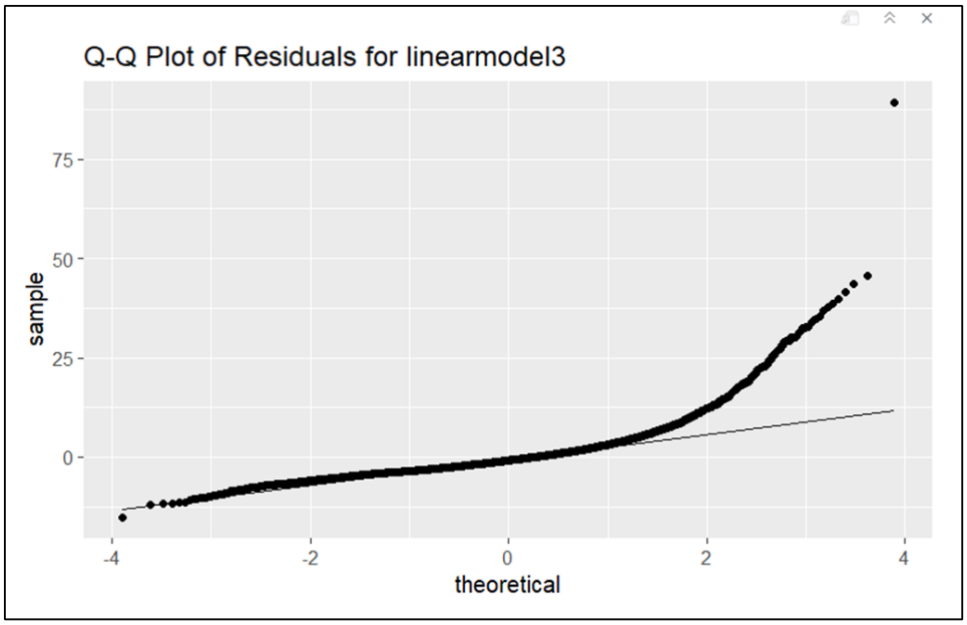


Figure 55: Linearmodel3 Q-Q plot of Residuals Vs theoretical

From the above plot, it is clear that the residuals do not follow normal distribution based on the departure of the right tail end of the plot from the normal-normal line. To confirm this postulation, we computed the Shapiro-Wilk test value which is shown below.

```
        Shapiro-Wilk normality test

data:  residuals(linearModel3)[10:5000]
W = 0.7804, p-value < 2.2e-16
```

Figure 56: Linearmodel3 Shapiro-Wilk test result

From the result above, since the P-value (2.2e-16 < P-value < 0.05) is less than 0.05, we will reject the null hypothesis that states that the residuals follow normal distribution.

We also checked to see if any outlier within the data itself is weighing heavily on this unexpected outcome of the linear model. So, we computed the Cook's distance and found out that it was less than 0.5 in value for the entire data interval. Kindly see the figure below.

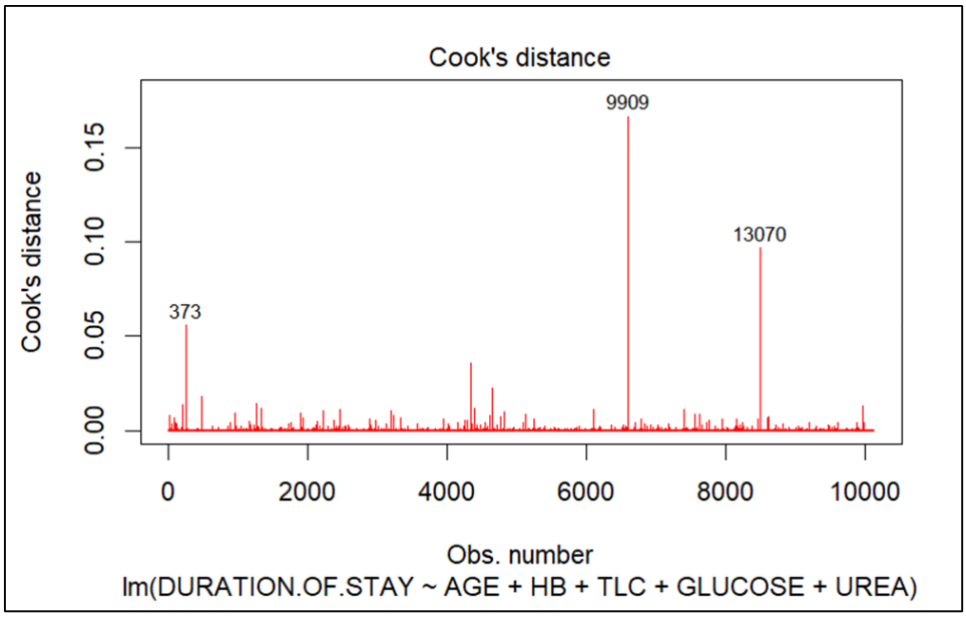


Figure 57: Linearmodel3 Cook's Distance result

We also computed the stepwise backward and forward models using all the independent variables as in LinearModel1 and ended up with the same result as in the manual selection. We therefore concluded that the linear model may not be the best option to predict the "DURATION.OF.STAY" in the hospital.

The next step was to run a Regression Tree to check if we would get a better Residual Standard Error and comparing it with the current LinearModel3 by splitting into 75% for training set and 25% for test set and thereafter also do 10 folds split for training and testing. The results are shown below.

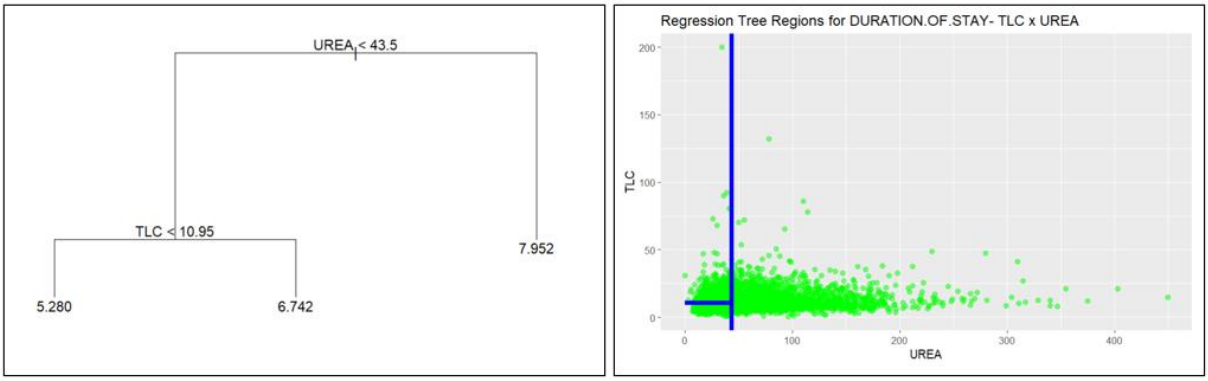


Figure 58: Regression Tree results. Only UREA and TLC were used in predicting DURATION.OF.STAY

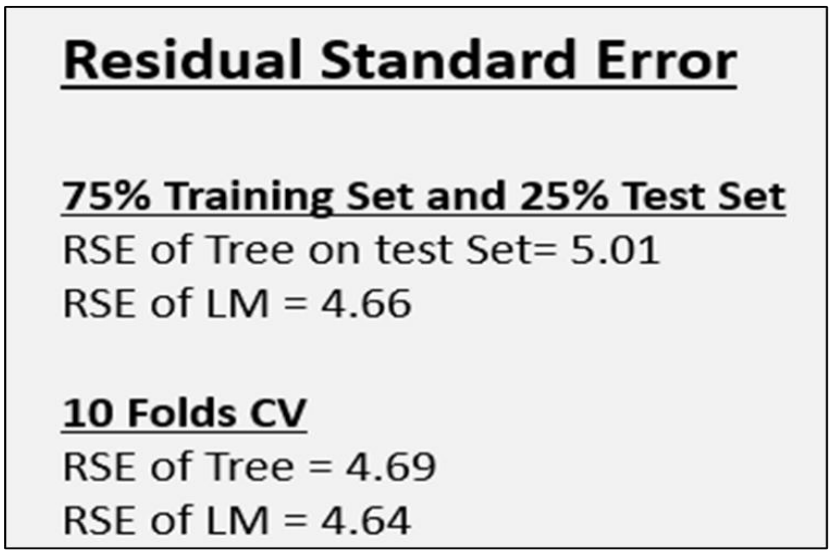


Figure 59: Comparison of Residual Standard Error of Linear Regression and Regression Tree

As can be seen from above, the linear model produces a better Residual Standard Error compared to the Regression Tree. Knowing that the linear model can only explain 7% of the "DURATION.OF.STAY" response function, we can conclude that the Tree may be less accurate than the Linear Regression. The low R-squared value of the Linear Regression made it difficult to pursue interaction terms or even higher order relationship. Therefore, predicting the "DURATION.OF.STAY" at the hospital may best be served by other machine learning algorithms that are best suited to the dataset.

7.3.4 CATEGORICAL RESPONSE VARIABLE (OUTCOME)

7.3.4.1 OUTCOME: MULTINOMIAL REGRESSION

Building the model

For multinomial Logistic regression, I divided the dataset into 75% training and 25% test parts. In the model, the response/dependent variable is the OUTCOME variable which has 3 levels **EXPIRY**, **DISCHARGE** and **DAMA**. We took the EXPIRY level as our reference level for our multinomial logistic regression model. We have tried our model with all the independent variables since multinomial logistic regression does not need any normality test. The model runs **60 iterations** to reduce the error.

```
# weights:  54 (34 variable)
initial  value 8287.931106
iter  10 value 5466.778223
iter  20 value 4332.846418
iter  30 value 2937.212536
iter  40 value 2620.909149
iter  50 value 2350.746483
iter  60 value 2343.184090
iter  60 value 2343.184080
iter  60 value 2343.184080
final  value 2343.184080
converged
```

Figure 60: Number of Iterations to minimize the error for Multinomial Regression

For getting the variables that are statistically significant i.e., p-value<0.05 and providing us with a 95% confidence interval we have used two-tailed z-test. We have kept the variables that are at least showing one p-value lower than statistically significant value 0.05 for either of the levels DISCHARGE or DAMA.

```
          (Intercept)         AGE EF          TLC           HB DURATION.OF.STAY     ALCOHOL1   SMOKING1
DAMA      5.232993e-06 0.095542611  0 6.068049e-05 2.638401e-02     5.585761e-01 1.386207e-05 0.01042824
DISCHARGE 1.056725e-03 0.001723679  0 0.000000e+00 7.867416e-09     1.324052e-12 2.401272e-04 0.01969746
                 DM1         CAD1         CKD1    PLATELETS     GLUCOSE       UREA STABLE.ANGINA1
DAMA      2.569705e-04 4.959428e-07 2.534108e-04 5.837651e-03 0.249856151 0.00873129              0
DISCHARGE 1.322067e-06 0.000000e+00 2.702286e-06 5.481682e-11 0.001422216 0.00000000              0
                 ACS1     STEMI1
DAMA      4.726124e-01 0.02575397
DISCHARGE 5.589304e-07 0.10009029
```

Figure 61: Variables that are statistically significant I.e., p-value < 0.05 for Multinomial Regression

The significant variables are:

- Age
- Ejection Fraction (EF)
- Total Leukocytes Count (TLC)
- Hemoglobin (HB)
- Duration of Stay

- Alcohol
- Smoking
- Diabetes Mellitus (DM).
- Coronary Artery Disease (CAD)
- PLATELETS
- GLUCOSE
- UREA
- STABLE.ANGINA
- Acute Coronary Syndrome
- STEMI
- Chronic Kidney Disease

Confusion Matrix and Misclassification error rate of training and test sets

For Training Set

	EXPIRY	DAMA	DISCHARGE
EXPIRY	127	18	48
DAMA	0	1	1
DISCHARGE	332	303	6714

Figure 62: Confusion Matrix of the training set for OUTCOME variable

For the training set, the model shows 90.69% accuracy, and the Misclassification error rate is **9.30%.**

For Expiry

The True positive for Expiry is 127. The value of False Negative is 66. The value of False Positive is 332. The value of True Negative is 7019.

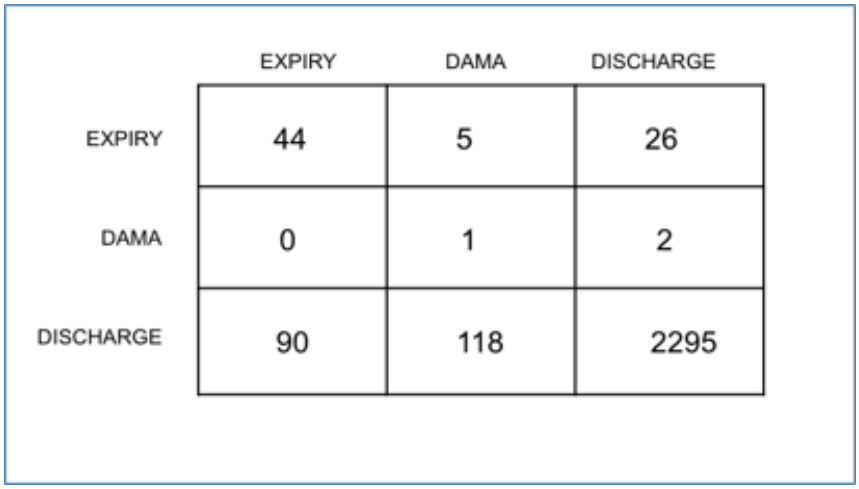
For DAMA

The True positive for DAMA is 1. The value of False Negative is 1. The value of False Positive is 321. The value of True Negative is 7221.

For DISCHARGE

The True positive for **DISCHARGE** is 6714. The value of False Negative is 635. The value of False Positive is 49. The value of True Negative is 146.

For Test Set



	EXPIRY	DAMA	DISCHARGE
EXPIRY	44	5	26
DAMA	0	1	2
DISCHARGE	90	118	2295

Figure 63: Confusion Matrix of the test set for OUTCOME variable

For the test set, the model shows **90.66%** accuracy, and the Misclassification error rate is **9.33%.**

For Expiry

The True positive for Expiry is 44. The value of False Negative is 31. The value of False Positive is 90. The value of True Negative is 2416.

For DAMA

The True positive for DAMA is 1. The value of False Negative is 2. The value of False Positive is 123. The value of True Negative is 2455.

For DISCHARGE

The True positive for **DISCHARGE** is 2295. The value of False Negative is 208. The value of False Positive is 28. The value of True Negative is 50.

MODEL ASSESMENT	
Training Set	**Test Set**
For Discharge the model shows accuracy of around 99.27 %, for Expiry and DAMA it shows 27.66% and 0.31% respectively.	For Discharge the model shows the accuracy of around 98.79%, for Expiry and DAMA it shows 32.83% and 0.80% respectively.

Figure 64: Model Assessment for multinomial regression of OUTCOME variable

7.3.4.2 OUTCOME: CLASSIFICATION TREE

For further analysis we have tried Classification Tree on the OUTCOME variable. First, we have created the classification tree.

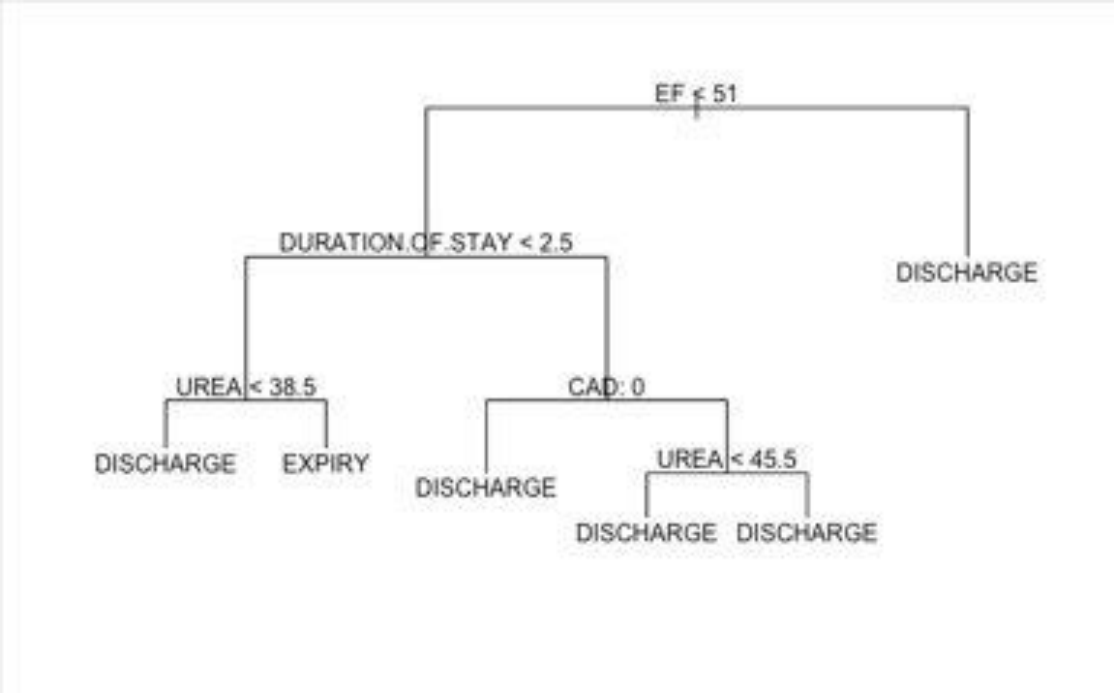


Figure 65: Classification tree for OUTCOME variable

Let ‘s check the probability in each terminal node

```
node), split, n, deviance, yval, (yprob)
      * denotes terminal node

 1) root 7544 6079.0 DISCHARGE ( 0.0608431 0.0426829 0.8964740 )
   2) EF < 51 4872 4874.0 DISCHARGE ( 0.0938013 0.0484401 0.8577586 )
     4) DURATION.OF.STAY < 2.5 520 1025.0 DISCHARGE ( 0.3307692 0.1461538 0.5230769 )
       8) UREA < 38.5 272  407.2 DISCHARGE ( 0.1250000 0.1323529 0.7426471 ) *
       9) UREA > 38.5 248  484.8 EXPIRY ( 0.5564516 0.1612903 0.2822581 ) *
     5) DURATION.OF.STAY > 2.5 4352 3454.0 DISCHARGE ( 0.0654871 0.0367647 0.8977482 )
      10) CAD: 0 1055 1386.0 DISCHARGE ( 0.1611374 0.0597156 0.7791469 ) *
      11) CAD: 1 3297 1866.0 DISCHARGE ( 0.0348802 0.0294207 0.9356991 )
        22) UREA < 45.5 2119  696.9 DISCHARGE ( 0.0089665 0.0240680 0.9669655 ) *
        23) UREA > 45.5 1178 1046.0 DISCHARGE ( 0.0814941 0.0390492 0.8794567 ) *
   3) EF > 51 2672  792.9 DISCHARGE ( 0.0007485 0.0321856 0.9670659 ) *
```

Figure 66: Node probabilities (without pruning) for OUTCOME

The majority of terminal nodes with more than 70% for the class indicated by the tree. In addition, the tree model will never predict DAMMA.

Here, we can see that for almost all the terminal nodes the predictions are toward Discharge level and the prediction values are significant. Except for UREA > 38.5 where it shows a probability of 55.64% for Expiry but this prediction can be wrong because the probability percentage is low.

Applying the unpruned tree to Test set

	EXPIRY	DAMA	DISCHARGE
EXPIRY	19	14	48
DAMA	0	0	0
DISCHARGE	89	110	2304

Figure 67: Confusion Matrix of the test set (unpruned tree) for OUTCOME variable

For test set (unpruned tree)

The overall **Misclassification Rate** for OUTCOME with Classification Tree is 0.08988764 (8.99%). However, the model will never predict DAMMA.

For Expiry

The True positive for Expiry is 19. The value of False Negative is 62. The value of False Positive is 89. The value of True Negative is 2414.

For DAMA

The True positive for DAMA is 0. The value of False Negative is 0. The value of False Positive is 124. The value of True Negative is 2460.

For DISCHARGE

The True positive for **DISCHARGE** is 2304. The value of False Negative is 199. The value of False Positive is 48. The value of True Negative is 33.

Pruning the tree

As the tree has several terminal nodes lets prune the tree. Checking the cross-validation error versus the number of Terminal nodes to Prune the tree using FUN

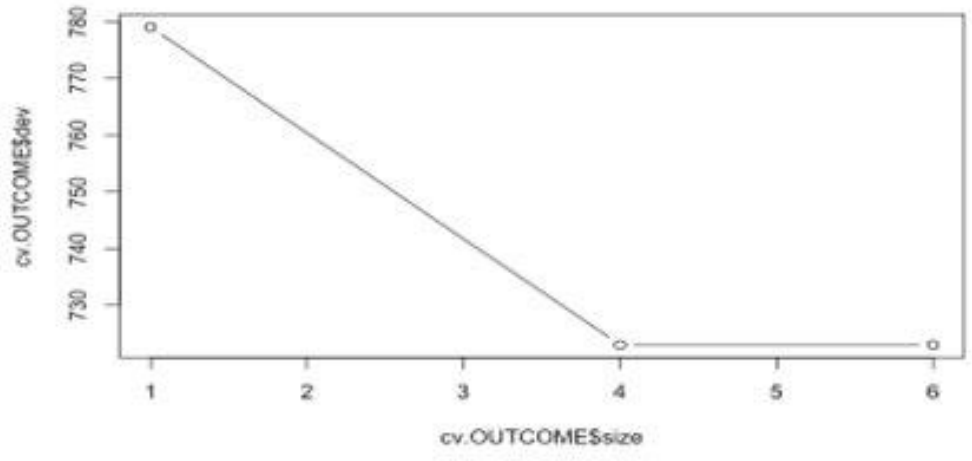


Figure 68: Cross-validation error before pruning the tree of OUTCOME variable

Based on the cross-validation error, it looks like the cross-validation error with 4 terminal nodes is like 6. Based on this, let's prune the tree to 4 terminal nodes.

Pruning the tree with 4 terminal nodes

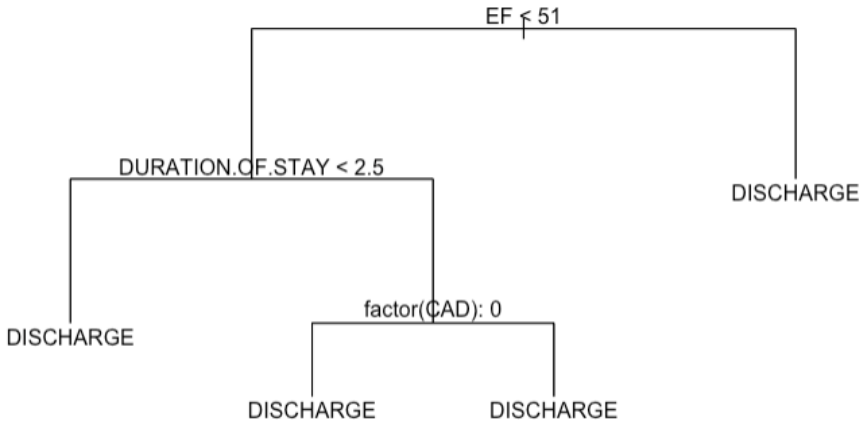


Figure 69: Pruned tree for OUTCOME variable

```
1) root 7544 6079.0 DISCHARGE ( 0.0608431 0.0426829 0.8964740 )
  2) EF < 51 4872 4874.0 DISCHARGE ( 0.0938013 0.0484401 0.8577586 )
    4) DURATION.OF.STAY < 2.5 520 1025.0 DISCHARGE ( 0.3307692 0.1461538 0.5230769 ) *
    5) DURATION.OF.STAY > 2.5 4352 3454.0 DISCHARGE ( 0.0654871 0.0367647 0.8977482 )
     10) factor(CAD): 0 1055 1386.0 DISCHARGE ( 0.1611374 0.0597156 0.7791469 ) *
     11) factor(CAD): 1 3297 1866.0 DISCHARGE ( 0.0348802 0.0294207 0.9356991 ) *
  3) EF > 51 2672  792.9 DISCHARGE ( 0.0007485 0.0321856 0.9670659 ) *
```

Figure 70: Node probabilities of pruned tree for OUTCOME variable

The pruned tree just indicates DISCHARGE for OUTCOME. Based on the probabilities in the terminal node, most wrong predictions should happen at the terminal node of: EF < 51 and DURATION.OF. STAY < 2.5.

Applying the pruned tree to Test set

	EXPIRY	DAMA	DISCHARGE
EXPIRY	0	0	0
DAMA	0	0	0
DISCHARGE	134	124	2323

Figure 71: Confusion Matrix of the test set (pruned tree) for OUTCOME variable

For test set (pruned tree)

The overall misclassification rate for OUTCOME with the Pruned Classification Tree is 0.09996126 (10%). However, the model will only predict DISCHARGE.

For Expiry

The True positive for Expiry is 0. The value of False Negative is 0. The value of False Positive is 134. The value of True Negative is 2447.

For DAMA

The True positive for DAMA is 0. The value of False Negative is 0. The value of False Positive is 124. The value of True Negative is 2457.

For DISCHARGE

The True positive for **DISCHARGE** is 2323. The value of False Negative is 258. The value of False Positive is 0. The value of True Negative is 0.

8. CONCLUSION AND RECOMMENDATIONS

Cluster sampling was not a candidate for our dataset based on the columns available in the cleaned dataset. SRS and stratified sampling were utilized, and SRS provided better accuracy when comparing the sampling means with the population mean. Moreover, SSB was very small compared to SSW which adds another point against stratified sampling being a good candidate for this dataset.

Analysis of independence using Table Contingency helped to select categorical exploratory variables that would be independent among them, but at the same time dependent on the response variable.

Based on the analysis done for HEART.FAILURE, the relevant and independent explanatory variables which influence on this indicate by our analysis were:

- Quantitative: AGE, GLUCOSE, HB, TLC, CREATININE, UREA and EF
- Qualitative: GENDER, RAISED.CARDIAC.ENZYMES and PRIOR.CMP

but the statistical learning models (except LDA) are possible to predict with almost the same accuracy using only EF and UREA.

Regarding the best statistical model to predict HEART FAILURE, the LOGISTIC REGRESSION without cross validation, in other words, using stratified sampling 75% of the total population as train set, indicated the best performance in prediction the training part with misclassification rate of 23.9%, following by Classification Tree without cross validation as well (misclassification of 24.5%). However, QDA model was the one that predicted more correctly the patients with heart failure, whereas LDA (even with only relevant and independent categorical explanatory variables due to no normal distribution of any quantitative variable) which predicted better the patients with no heart failure.

In terms of building the model with 10-fold stratified cross-validation, LDA (only the qualitative variables) had the best performance among all the statistical learning methods with the misclassification rate of 25.3%.

It is important to mention that the normality tests (both Shapiro-Will and Kolmogorov-Smirnov) indicated that none of the quantitative variables has normal distribution, and consequently they were not used in LDA modelling seeing that this statistical learning method requires normal distribution of explanatory variables.

Comparing the HEART.FAILURE model and prediction on this project with the one provided by the professor [**Ref.5**], the final cleaned datasets are totally different between these two projects, for example on this project will have more than 10000 units, whereas the previous work approximately 300 units. In addition, the majority of the explanatory variables used in the previous project were totally different from what was used on our project, for example in the previous project just one categorical variable was available to be used as explanatory variable.

While analysis of different model of acute kidney injury (AKI) and comparing model results, the Logistic Regression model without stratified sampling indicated the lowest or best misclassification rate of **1.03%** among all the generated models to predict AKI. However, we did not observe significant differences with logistic regression with 10 k-fold stratified cross validation and classification tree with and without cross validation.

Knowing that the linear model can only explain 7% of the "DURATION.OF.STAY" response function and that the Regression Tree had similar Residual Standard Error as the Linear Regression, we can conclude that the Regression Tree and Linear Regression are not able to predict the "DURATION.OF.STAY" at the hospital properly. Also, the low R-squared value of the Linear Regression made it difficult to pursue interaction terms or even higher order relationship. Therefore, predicting the "DURATION.OF.STAY" at the hospital may best be served by other machine learning algorithms that are best suited to the dataset.

For multinomial logistic regression we have examined 3 levels (EXPIRY, DAMA, DISCHARGE) the model provides 90.69% accuracy and the Misclassification error rate of 9.30% for the training set. Similarly, the model shows 90.66% accuracy for the test set 90.66% accuracy, and the Misclassification error rate is 9.33%. For both training and test sets, DISCHARGE offers the highest accuracy percentage of 99.27% and 98.79% respectively.

For the classification tree with the OUTCOME variable applying unpruned tree to the test set, we get the Misclassification error rate of 0.08988764 (8.99%). However, the model will never predict DAMMA. Again, applying the pruned tree to the test set with 4 terminal nodes the Misclassification error rate is 0.09996126 (10%). This time also the model will only predict DISCHARGE.

So, from both multinomial logistic regression and classification taking OUTCOME as the response variable we can see that the misclassification rates are significantly low to be precise not more than 10%.

9. DIVISION OF WORK

We think everything collaborated equally in our team, for example at the beginning by doing some initial Data Analysis in EXCEL to define the possibilities with the dataset in terms of applying what you learned not only in DATA-606, but also in DATA-603.

We worked with multinomial logistic regression and classification tree models on the variable OUTCOME which has 3 levels EXPIRY, DISCHARGE and DAMA. Divided the entire dataset into 75% training set and 25% test set. For multinomial logistic regression, all the independent variables for p-values are tested that are statistically significant. Created confusion matrices to analyze misclassification rates and model accuracies. For classification tree found misclassification rates for both pruned and unpruned trees. Created classification tree, checked cross-validation error for pruning the tree.

10. REFRENCES

1.Hospital Admissions Data, Kaggle. Available at https://www.kaggle.com/datasets/ashishsahani/hospital-admissions-data/discussion/302894?resource=download&select=HDHI+Admission+data.csv (accessed on Jan. 25,2023).

2.Bollepalli, S.C.; Sahani, A.K.; Aslam, N.; Mohan, B.; Kulkarni, K.; Goyal, A.; Singh, B.; Singh, G.; Mittal, A.; Tandon, R.; Chhabra, S.T.; Wander, G.S.; Armoundas, A.A. An Optimized Machine Learning Model Accurately Predicts In-Hospital Outcomes at Admission to a Cardiac Unit. December 12,2022, Kaggle. Available at https://doi.org/10.3390/diagnostics12020241 (accessed on Jan. 30,2023).

3.Attribution-NonCommercial-ShareAlike 4.0 International (CC BY-NC-SA 4.0),Creative Commons, Available at https://creativecommons.org/licenses/by-nc-sa/4.0/ (accessed on Jan. 30,2023).

4.Leading causes of death, February 22,2022, Open Canada-Government of Alberta. Available at https://open.canada.ca/data/en/dataset/03339dc5-fb51-4552-97c7-853688fc428d (accessed on Jan. 28,2023).

5. Kramar, S.; Yang, Y; Sultan, M.. Heart Disease Prediction Models. Final Project Report of DATA 606, 2021.

6. American Heart Association: Ejection Fraction website: https://www.heart.org/en/health-topics/heart-failure/diagnosing-heart-failure/ejection-fraction-heart-failure-measurement (accessed on February 15th, 2023).

11. APPENDIX - R CODE

Project Report - Exploring Cardiovascular Disease and its underlying factors using Statistical Methods

DATA 606

February 08, 2023

Introduction

The human body constitutes of many diverse types of cells that together create tissues and subsequently organ systems. The heart is at the focal point of the human body responsible for pumping blood throughout the body and keeping us alive. Our focus of analysis is cardiovascular disease which is one of the leading causes of death globally. We will try to identify the following from our analysis: • Observing patients admitted to the hospital with diverse cardiovascular diseases. • Analyzing the underlying conditions associated with the patient. • Analyzing the outcome of the patient for the duration of hospital admission. We are still trying to assess how much scope we can handle within the time available for this project. We have identified some datasets from various sources and described them in the next section.

Dataset(s)

We have utilized the "Hospital Admissions Data" dataset (File size: 2.6 MB, Rows: 15757 K, Columns: 56) available in the Kaggle portal (https://www.kaggle.com/datasets/ashishsahani/hospital-admissions-data/discussion/302894?resource=download&select=HDHI+Admission+data.csv)[Ref. 1] in CSV format. The Kaggle portal provides this dataset free of charge and can be used for research/project purposes. Details of their conditions are available in https://doi.org/10.3390/diagnostics12020241[Ref. 2]. This dataset is being provided under creative commons License (Attribution-Non-Commercial-Share Alike 4.0 International (CC BY-NC-SA 4.0)) https://creativecommons.org/licenses/by-nc-sa/4.0/ [Ref. 3]. This data was collected from patients admitted over a period of two years (1 April 2017 to 31 March 2019) at Hero DMC Heart Institute, Unit of Dayanand Medical College and Hospital, Ludhiana, Punjab, India. During the study period, the cardiology unit had 15,757 admissions corresponding to 12,238 patients. 1921 patients who had multiple admissions. Specifically, data were related to patients ; date of admission; date of discharge; demographics, such as age, sex, locality (rural or urban); type of admission (emergency or outpatient); patient history, including smoking, alcohol, diabetes mellitus (DM), hypertension (HTN), prior coronary artery disease (CAD), prior cardiomyopathy (CMP), and chronic kidney disease (CKD); and lab parameters corresponding to

hemoglobin (HB), total lymphocyte count (TLC), platelets, glucose, urea, creatinine, brain natriuretic peptide (BNP), raised cardiac enzymes (RCE) and ejection fraction (EF). Other comorbidities and features (28 features), including heart failure, STEMI, and pulmonary embolism, were recorded, and analyzed. The outcomes indicating whether the patient was discharged or expired in the hospital were also recorded. table_headings.csv - This data table has the descriptive headlines for all columns for the HDHI Admission data file.

Research Questions

• Which independent variables have the best correlation to predicting the likelihood of the main hospital admission causes, e.g., heart failure, acute coronary syndrome (ACS) and acute kidney injury (AKI)? • Can we predict the possibility of heart failure, ACS and AKI looking at common habits of individuals? What kind of misclassification rate will we get from the study comparing different methods of predicting heart failures? • Is it possible to predict some relation between different causes of admissions with the main ones (heart failure, ACS, and AKI)? • What would be the true mean and standard deviation of duration of patients for the three different outcomes (EXPIRY, DISCHARGE and DAMA, in other words, passed away, discharge with medical approval and against medical approval, respectively) compared with the means and standard deviations by using different sampling methods (SRS, stratified and Cluster)? • If there is some auxiliary variable which we could use to increase the accuracy of sampling outcomes?

Data Wrangling & Analysis Procedure

For data cleaning/wrangling and preliminary exploration/analysis related to individual parts of the project we will be utilizing R. We will perform if needed, formatting, sorting, and cleaning of all data as explained below: • Data type conversions • Deleting date columns to remove time-based dependency. • Sorting all the datasets for systematic analysis. • Deleting non-required columns as there are many categorical variables. • Removing null/missing values. • Format checking • Cross-check for duplicates. • Perform analysis on our research questions and add necessary visualizations.

Technique used

Our main analysis for this project will be conducted using R. We are planning to implement the following techniques: • Sampling techniques such as SRS, Stratified random sampling. • Multinomial logistic regression • Linear discriminant analysis • Quadratic discriminant analysis • Resampling cross validation • Tree-based classification • Contingency table

Further techniques will be included/removed in the final analysis and an accompanying report will be provided at the end of the project work.

```r
#Library inclusion
library(olsrr)
library(ggplot2)
library(GGally)
library(lmtest)
library(mctest)
library(Ecdat)
library(MASS)
library(caret)
library(leaps)
require(car)
require(zoo)
library("dplyr")
library(sampling)

library(fmsb)
library(reshape)
library(caret)
library(MASS)
library(reshape2)
library(mctest)


set.seed(10)
```

PART I - READING DATA, UNDERSTAND DATASET AND EXPLORATION DATA ANALYSIS (EDA)

READING DATASET

```r
# Read the dataset and replacing blank values with NA
dfH <- read.csv("HDHI Admission data.csv", header =  TRUE,na.strings=c
("", "NA"))

#Checking names in the dataset
names(dfH)
```

```
##  [1] "SNO"                                "MRD.No."
##  [3] "D.O.A"                              "D.O.D"
##  [5] "AGE"                                "GENDER"
##  [7] "RURAL"                              "TYPE.OF.ADMISSION.EMERGENCY
.OPD"
##  [9] "month.year"                         "DURATION.OF.STAY"
## [11] "duration.of.intensive.unit.stay" "OUTCOME"
## [13] "SMOKING"                           "ALCOHOL"
## [15] "DM"                                "HTN"
```

```
## [17] "CAD"                                  "PRIOR.CMP"
## [19] "CKD"                                  "HB"
## [21] "TLC"                                  "PLATELETS"
## [23] "GLUCOSE"                              "UREA"
## [25] "CREATININE"                           "BNP"
## [27] "RAISED.CARDIAC.ENZYMES"               "EF"
## [29] "SEVERE.ANAEMIA"                       "ANAEMIA"
## [31] "STABLE.ANGINA"                        "ACS"
## [33] "STEMI"                                "ATYPICAL.CHEST.PAIN"
## [35] "HEART.FAILURE"                        "HFREF"
## [37] "HFNEF"                                "VALVULAR"
## [39] "CHB"                                  "SSS"
## [41] "AKI"                                  "CVA.INFRACT"
## [43] "CVA.BLEED"                            "AF"
## [45] "VT"                                   "PSVT"
## [47] "CONGENITAL"                           "UTI"
## [49] "NEURO.CARDIOGENIC.SYNCOPE"            "ORTHOSTATIC"
## [51] "INFECTIVE.ENDOCARDITIS"               "DVT"
## [53] "CARDIOGENIC.SHOCK"                    "SHOCK"
## [55] "PULMONARY.EMBOLISM"                   "CHEST.INFECTION"
```

```
#Checking just to have some idea about the variables in the dataset
head(dfH)
```

```
##   SNO MRD.No.    D.O.A     D.O.D AGE GENDER RURAL
## 1   1  234735 4/1/2017  4/3/2017  81      M     R
## 2   2  234696 4/1/2017  4/5/2017  65      M     R
## 3   3  234882 4/1/2017  4/3/2017  53      M     U
## 4   4  234635 4/1/2017  4/8/2017  67      F     U
## 5   5  234486 4/1/2017 4/23/2017  60      F     U
## 6   6  234675 4/1/2017 4/10/2017  44      M     U
##   TYPE.OF.ADMISSION.EMERGENCY.OPD month.year DURATION.OF.STAY
## 1                               E     Apr-17                3
## 2                               E     Apr-17                5
## 3                               E     Apr-17                3
## 4                               E     Apr-17                8
## 5                               E     Apr-17               23
## 6                               E     Apr-17               10
##   duration.of.intensive.unit.stay   OUTCOME SMOKING ALCOHOL DM HTN
CAD
## 1                               2 DISCHARGE       0       0  1   0
0
## 2                               2 DISCHARGE       0       1  0   1
1
## 3                               3 DISCHARGE       0       0  1   0
1
```

```
## 4                              6 DISCHARGE       0       0  0   1
1
## 5                              9 DISCHARGE       0       0  0   1
0
## 6                              8 DISCHARGE       0       0  1   1
1
##   PRIOR.CMP CKD   HB  TLC PLATELETS GLUCOSE UREA CREATININE  BNP
## 1         0   0  9.5 16.1       337      80   34        0.9 1880
## 2         0   0 13.7    9       149     112   18        0.9 <NA>
## 3         0   0 10.6 14.7       329     187   93        2.3  210
## 4         0   0 12.8  9.9       286     130   27        0.6 <NA>
## 5         1   0 13.6  9.1        26     144   55       1.25 1840
## 6         1   0 13.5 22.3       322     217   51        0.9 1720
##   RAISED.CARDIAC.ENZYMES   EF SEVERE.ANAEMIA ANAEMIA STABLE.ANGINA
ACS STEMI
## 1                      1   35              0       1             0
1     0
## 2                      0   42              0       0             0
0     0
## 3                      0 <NA>              0       0             0
0     0
## 4                      0   42              0       0             0
0     0
## 5                      0   16              0       0             0
0     0
## 6                      0   25              0       0             0
1     0
##   ATYPICAL.CHEST.PAIN HEART.FAILURE HFREF HFNEF VALVULAR CHB SSS AK
I
## 1                   0             1     1     0        0   0   0
0
## 2                   0             0     0     0        0   0   0
0
## 3                   0             1     1     0        0   0   0
1
## 4                   0             0     0     0        0   0   0
0
## 5                   0             0     0     0        0   0   0
0
## 6                   0             1     1     0        0   0   0
0
##   CVA.INFRACT CVA.BLEED AF VT PSVT CONGENITAL UTI NEURO.CARDIOGENIC
.SYNCOPE
## 1           0         0  0  0    0          0   0
0
## 2           0         0  0  1    0          0   0
```

```
0
## 3           0         0  0  0    0          0   0
0
## 4           0         0  0  0    0          0   0
0
## 5           0         0  0  0    0          0   0
0
## 6           0         0  0  1    0          0   0
0
##   ORTHOSTATIC INFECTIVE.ENDOCARDITIS DVT CARDIOGENIC.SHOCK SHOCK
## 1           0                       0   0                 0     0
## 2           0                       0   0                 0     0
## 3           0                       0   0                 0     0
## 4           0                       0   0                 0     0
## 5           0                       0   0                 0     0
## 6           0                       0   0                 0     0
##   PULMONARY.EMBOLISM CHEST.INFECTION
## 1                  0               0
## 2                  0               0
## 3                  0               0
## 4                  0               0
## 5                  0               0
## 6                  0               0
```

```
#Checking dimension
dim(dfH) # we found 15757 rows and 56 columns
```

```
## [1] 15757    56
```

We see our dataset has total 15,757 rows and 56 columns.

After analysing the dataset, we found duplicate admission entries that is MRD No. in our dataset with most of them amost identical information. Therefore, we have removed those duplicates as follows.

```
# Remove duplicates based on MRD No which is the admission number
dfH1=dfH[!duplicated(dfH$MRD.No.), ]
dim(dfH1)
```

```
## [1] 12244    56
```

```
#summary(dfH1)
```

After removing duplicate entries, we see our dataset now has 12,224 rows and 56 columns.

Visualizaiton of the dataset

Creating a special dataset for plots

```
data1xplots <- dfH1
```

Working with the original data before dropping columns for the analysis of categorical variables that may impact interpretation.

```
# plotData <- data.frame(data1xplots)

head (data1xplots)
```

```
##   SNO MRD.No.    D.O.A     D.O.D AGE GENDER RURAL
## 1   1  234735 4/1/2017  4/3/2017  81      M     R
## 2   2  234696 4/1/2017  4/5/2017  65      M     R
## 3   3  234882 4/1/2017  4/3/2017  53      M     U
## 4   4  234635 4/1/2017  4/8/2017  67      F     U
## 5   5  234486 4/1/2017 4/23/2017  60      F     U
## 6   6  234675 4/1/2017 4/10/2017  44      M     U
##   TYPE.OF.ADMISSION.EMERGENCY.OPD month.year DURATION.OF.STAY
## 1                               E     Apr-17                3
## 2                               E     Apr-17                5
## 3                               E     Apr-17                3
## 4                               E     Apr-17                8
## 5                               E     Apr-17               23
## 6                               E     Apr-17               10
##   duration.of.intensive.unit.stay   OUTCOME SMOKING ALCOHOL DM HTN
CAD
## 1                               2 DISCHARGE       0       0  1   0
0
## 2                               2 DISCHARGE       0       1  0   1
1
## 3                               3 DISCHARGE       0       0  1   0
1
## 4                               6 DISCHARGE       0       0  0   1
1
## 5                               9 DISCHARGE       0       0  0   1
0
## 6                               8 DISCHARGE       0       0  1   1
1
##   PRIOR.CMP CKD   HB  TLC PLATELETS GLUCOSE UREA CREATININE  BNP
## 1         0   0  9.5 16.1       337      80   34        0.9 1880
## 2         0   0 13.7    9       149     112   18        0.9 <NA>
## 3         0   0 10.6 14.7       329     187   93        2.3  210
## 4         0   0 12.8  9.9       286     130   27        0.6 <NA>
## 5         1   0 13.6  9.1        26     144   55       1.25 1840
```

```
## 6         1   0 13.5 22.3       322     217   51        0.9 1720
##   RAISED.CARDIAC.ENZYMES   EF SEVERE.ANAEMIA ANAEMIA STABLE.ANGINA
ACS STEMI
## 1                      1   35              0       1             0
1     0
## 2                      0   42              0       0             0
0     0
## 3                      0 <NA>              0       0             0
0     0
## 4                      0   42              0       0             0
0     0
## 5                      0   16              0       0             0
0     0
## 6                      0   25              0       0             0
1     0
##   ATYPICAL.CHEST.PAIN HEART.FAILURE HFREF HFNEF VALVULAR CHB SSS AK
I
## 1                   0             1     1     0        0   0   0
0
## 2                   0             0     0     0        0   0   0
0
## 3                   0             1     1     0        0   0   0
1
## 4                   0             0     0     0        0   0   0
0
## 5                   0             0     0     0        0   0   0
0
## 6                   0             1     1     0        0   0   0
0
##   CVA.INFRACT CVA.BLEED AF VT PSVT CONGENITAL UTI NEURO.CARDIOGENIC
.SYNCOPE
## 1           0         0  0  0    0          0   0
0
## 2           0         0  0  1    0          0   0
0
## 3           0         0  0  0    0          0   0
0
## 4           0         0  0  0    0          0   0
0
## 5           0         0  0  0    0          0   0
0
## 6           0         0  0  1    0          0   0
0
##   ORTHOSTATIC INFECTIVE.ENDOCARDITIS DVT CARDIOGENIC.SHOCK SHOCK
## 1           0                      0   0                 0     0
## 2           0                      0   0                 0     0
```

```
## 3           0                       0   0                   0     0
## 4           0                       0   0                   0     0
## 5           0                       0   0                   0     0
## 6           0                       0   0                   0     0
##   PULMONARY.EMBOLISM CHEST.INFECTION
## 1                  0               0
## 2                  0               0
## 3                  0               0
## 4                  0               0
## 5                  0               0
## 6                  0               0
```

```
library(reshape2)

unique.dim <- sort(unique(data1xplots$SMOKING))
count.smoking <- table(data1xplots$SMOKING)
count.alcohol <- table(data1xplots$ALCOHOL)
count.DM <- table(data1xplots$DM)
count.HTN <- table(data1xplots$HTN)
count.CAD <- table(data1xplots$CAD)
count.priorCMP <- table(data1xplots$PRIOR.CMP)
count.CKD <- table(data1xplots$CKD)



dfcomb <- data.frame(unique.dim, count.smoking, count.alcohol, count.D
M, count.HTN, count.CAD, count.priorCMP, count.CKD)

head(dfcomb)
```

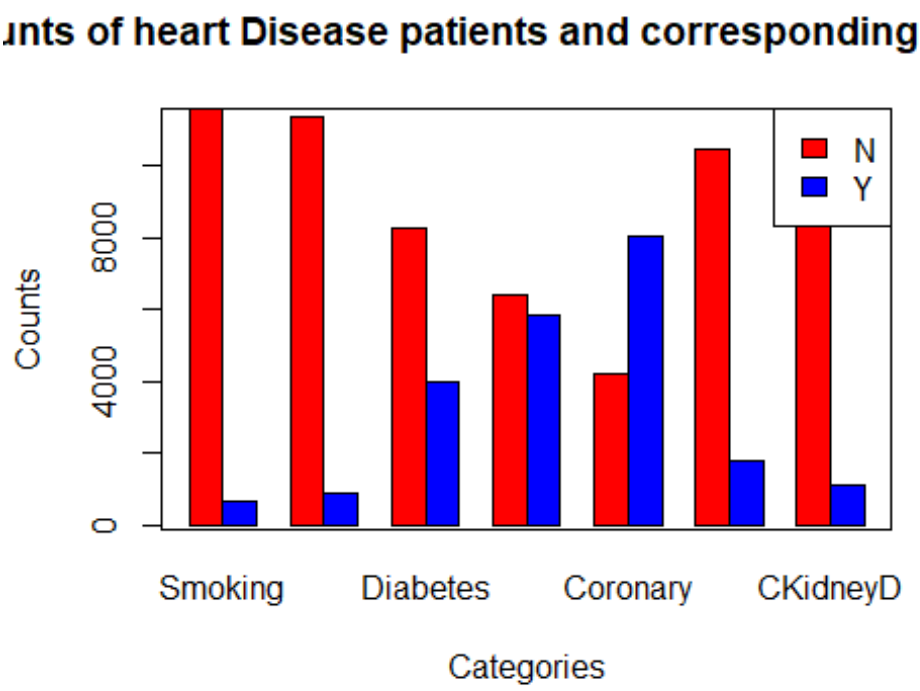
```
##   unique.dim Var1  Freq Var1.1 Freq.1 Var1.2 Freq.2 Var1.3 Freq.3 V
ar1.4 Freq.4
## 1          0    0 11578      0  11367      0   8266      0   6415
0   4230
## 2          1    1   666      1    877      1   3978      1   5829
1   8014
##   Var1.5 Freq.5 Var1.6 Freq.6
## 1      0  10472      0  11171
## 2      1   1772      1   1073
```

```
dfcomb2 = dfcomb[-2]
dfcomb3 = dfcomb2[-3]
dfcomb4 = dfcomb3[-4]
dfcomb5 = dfcomb4[-5]
dfcomb6 = dfcomb5[-6]
dfcomb7 = dfcomb6[-7]
dfcomb8 = dfcomb7[-8]
```

```
# head(dfcomb8)
names(dfcomb8) <- c("unique.dim", "Smoking", "Alcohol", "Diabetes", "H
yper", "Coronary", "CardioM", "CKidneyD")
head(dfcomb8)
```

```
##   unique.dim Smoking Alcohol Diabetes Hyper Coronary CardioM CKidne
yD
## 1          0   11578   11367     8266  6415     4230   10472    111
71
## 2          1     666     877     3978  5829     8014    1772     10
73
```

```
colours = c("red", "blue")
barplot(as.matrix(dfcomb8[-1]), main = "Counts of heart Disease patien
ts and corresponding diagnosis", ylab="Counts", xlab = "Categories", b
eside = TRUE, col = colours)

box()

legend ('topright', fill = colours, legend=c('N', 'Y'))
```



Facing set2

```
count.anaemia <- table(data1xplots$ANAEMIA)
count.sangina <- table(data1xplots$STABLE.ANGINA)
count.acs <- table(data1xplots$ACS)
```

```r
count.stemi <- table(data1xplots$STEMI)
count.achestpain <- table(data1xplots$ATYPICAL.CHEST.PAIN)
count.hfref <- table(data1xplots$HFREF)
count.hfnef <- table(data1xplots$HFNEF)

dfcombx <- data.frame(unique.dim, count.anaemia, count.sangina, count.
acs, count.stemi, count.achestpain, count.hfref, count.hfnef)

head(dfcombx)
```

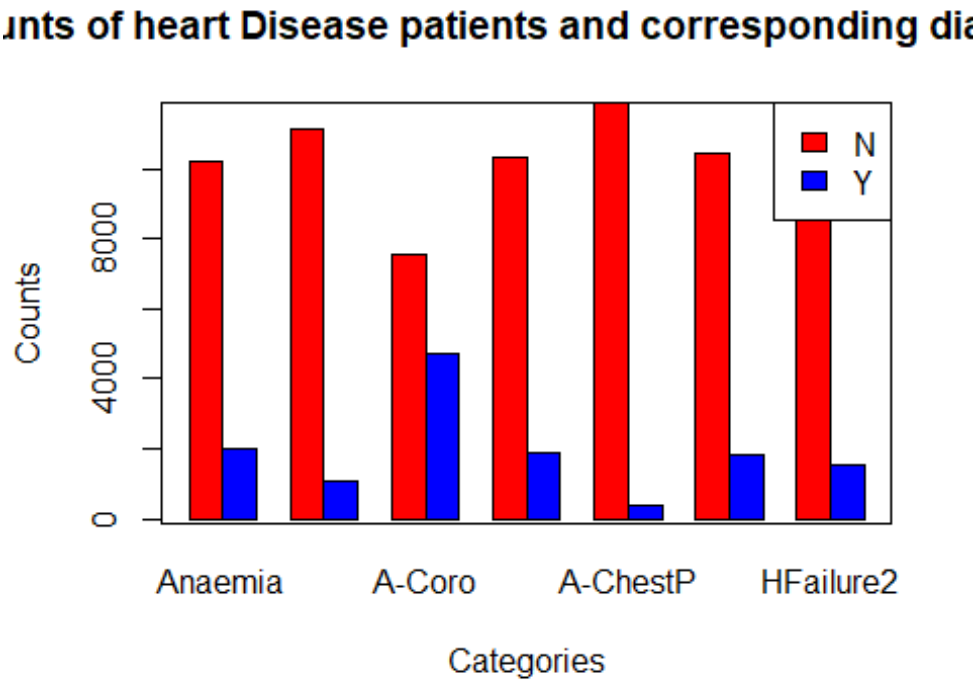
```
##   unique.dim Var1  Freq Var1.1 Freq.1 Var1.2 Freq.2 Var1.3 Freq.3 V
ar1.4 Freq.4
## 1          0    0 10224      0  11161      0   7535      0  10361
0  11890
## 2          1    1  2020      1   1083      1   4709      1   1883
1    354
##   Var1.5 Freq.5 Var1.6 Freq.6
## 1      0  10434      0  10713
## 2      1   1810      1   1531
```

```r
dfcombx2 = dfcombx[-2]
dfcombx3 = dfcombx2[-3]
dfcombx4 = dfcombx3[-4]
dfcombx5 = dfcombx4[-5]
dfcombx6 = dfcombx5[-6]
dfcombx7 = dfcombx6[-7]
dfcombx8 = dfcombx7[-8]

# head(dfcomb8)
names(dfcombx8) <- c("unique.dim", "Anaemia", "SAngina", "A-Coro", "My
ocard", "A-ChestP", "HFailure1", "HFailure2")
head(dfcombx8)
```

```
##   unique.dim Anaemia SAngina A-Coro Myocard A-ChestP HFailure1 HFai
lure2
## 1          0   10224   11161   7535   10361    11890     10434
10713
## 2          1    2020    1083   4709    1883      354      1810
1531
```

```r
colours = c("red", "blue")
barplot(as.matrix(dfcombx8[-1]), main = "Counts of heart Disease patie
nts and corresponding diagnosis", ylab="Counts", xlab = "Categories",
beside = TRUE, col = colours)

box()
```

```
legend ('topright', fill = colours, legend=c('N', 'Y'))
```



Set 3

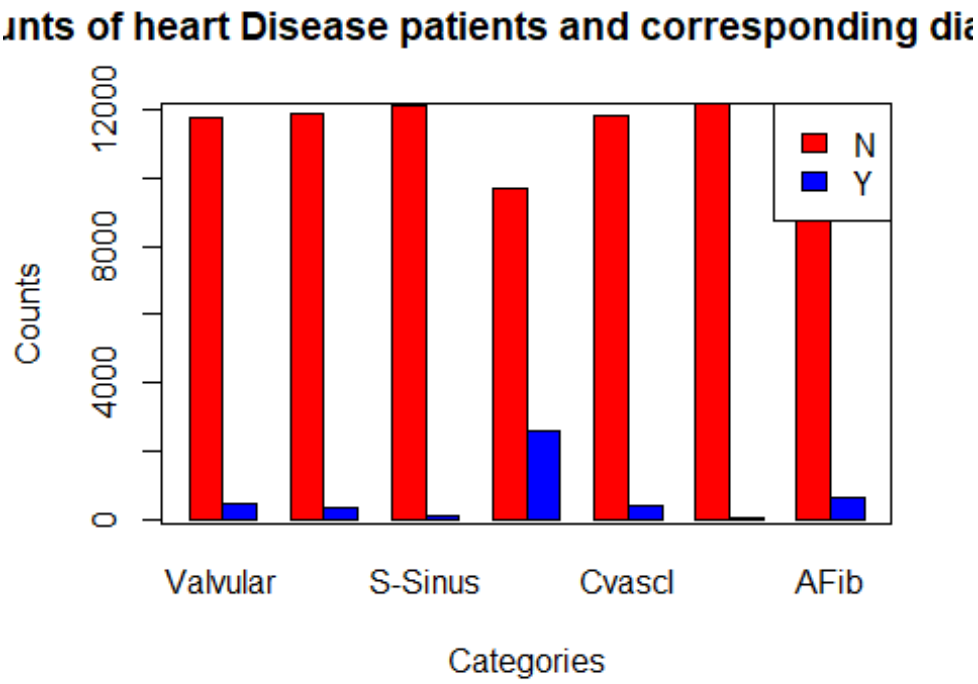
```
count.valvular <- table(data1xplots$VALVULAR)
count.chb <- table(data1xplots$CHB)
count.sss <- table(data1xplots$SSS)
count.aki <- table(data1xplots$AKI)
count.cvainfract <- table(data1xplots$CVA.INFRACT)
count.cvableed <- table(data1xplots$CVA.BLEED)
count.af <- table(data1xplots$AF)

dfcomby <- data.frame(unique.dim, count.valvular, count.chb, count.sss
, count.aki, count.cvainfract, count.cvableed, count.af)

head(dfcomby)

##   unique.dim Var1  Freq Var1.1 Freq.1 Var1.2 Freq.2 Var1.3 Freq.3 V
ar1.4 Freq.4
## 1          0    0 11816      0  11911      0  12155      0   9686
0  11872
## 2          1    1   428      1    333      1     89      1   2558
1    372
##   Var1.5 Freq.5 Var1.6 Freq.6
```

```
## 1      0  12191      0  11633
## 2      1     53      1    611
```

```
dfcomby2 = dfcomby[-2]
dfcomby3 = dfcomby2[-3]
dfcomby4 = dfcomby3[-4]
dfcomby5 = dfcomby4[-5]
dfcomby6 = dfcomby5[-6]
dfcomby7 = dfcomby6[-7]
dfcomby8 = dfcomby7[-8]

# head(dfcomb8)
names(dfcomby8) <- c("unique.dim", "Valvular", "CHblock", "S-Sinus", "
AKidney-I", "CvascI", "CvascB", "AFib")
head(dfcomby8)
```

```
##   unique.dim Valvular CHblock S-Sinus AKidney-I CvascI CvascB  AFib
## 1          0    11816   11911   12155      9686  11872  12191 11633
## 2          1      428     333      89      2558    372     53   611
```

```
colours = c("red", "blue")
barplot(as.matrix(dfcomby8[-1]), main = "Counts of heart Disease patie
nts and corresponding diagnosis", ylab="Counts", xlab = "Categories",
beside = TRUE, col = colours)

box()

legend ('topright', fill = colours, legend=c('N', 'Y'))
```



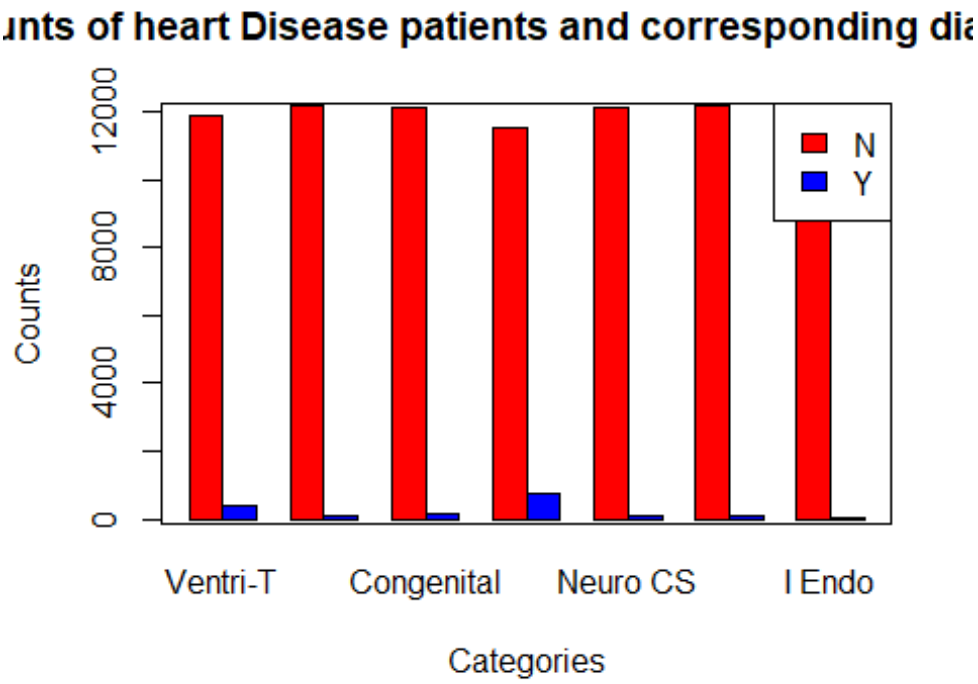
Set 4

```
count.vt <- table(data1xplots$VT)
count.psvt <- table(data1xplots$PSVT)
count.congenital <- table(data1xplots$CONGENITAL)
count.uti <- table(data1xplots$UTI)
count.ncs <- table(data1xplots$NEURO.CARDIOGENIC.SYNCOPE)
count.orthostatic <- table(data1xplots$ORTHOSTATIC)
count.iendocarditis <- table(data1xplots$INFECTIVE.ENDOCARDITIS)

dfcombz <- data.frame(unique.dim, count.vt, count.psvt, count.congenit
al, count.uti, count.ncs, count.orthostatic, count.iendocarditis)

head(dfcombz)

##   unique.dim Var1  Freq Var1.1 Freq.1 Var1.2 Freq.2 Var1.3 Freq.3 V
ar1.4 Freq.4
## 1          0    0 11866      0  12159      0  12097      0  11514
0  12133
## 2          1    1   378      1     85      1    147      1    730
1    111
##   Var1.5 Freq.5 Var1.6 Freq.6
## 1      0  12144      0  12219
## 2      1    100      1     25
```

```r
dfcombz2 = dfcombz[-2]
dfcombz3 = dfcombz2[-3]
dfcombz4 = dfcombz3[-4]
dfcombz5 = dfcombz4[-5]
dfcombz6 = dfcombz5[-6]
dfcombz7 = dfcombz6[-7]
dfcombz8 = dfcombz7[-8]

# head(dfcomb8)
names(dfcombz8) <- c("unique.dim", "Ventri-T", "PSVentri", "Congenital
", "Uri-TI", "Neuro CS", "Ortho", "I Endo")
head(dfcombz8)
```

```
##   unique.dim Ventri-T PSVentri Congenital Uri-TI Neuro CS Ortho I E
ndo
## 1          0    11866    12159      12097  11514    12133 12144  12
219
## 2          1      378       85        147    730      111   100
25
```

```r
colours = c("red", "blue")
barplot(as.matrix(dfcombz8[-1]), main = "Counts of heart Disease patie
nts and corresponding diagnosis", ylab="Counts", xlab = "Categories",
beside = TRUE, col = colours)

box()

legend ('topright', fill = colours, legend=c('N', 'Y'))
```



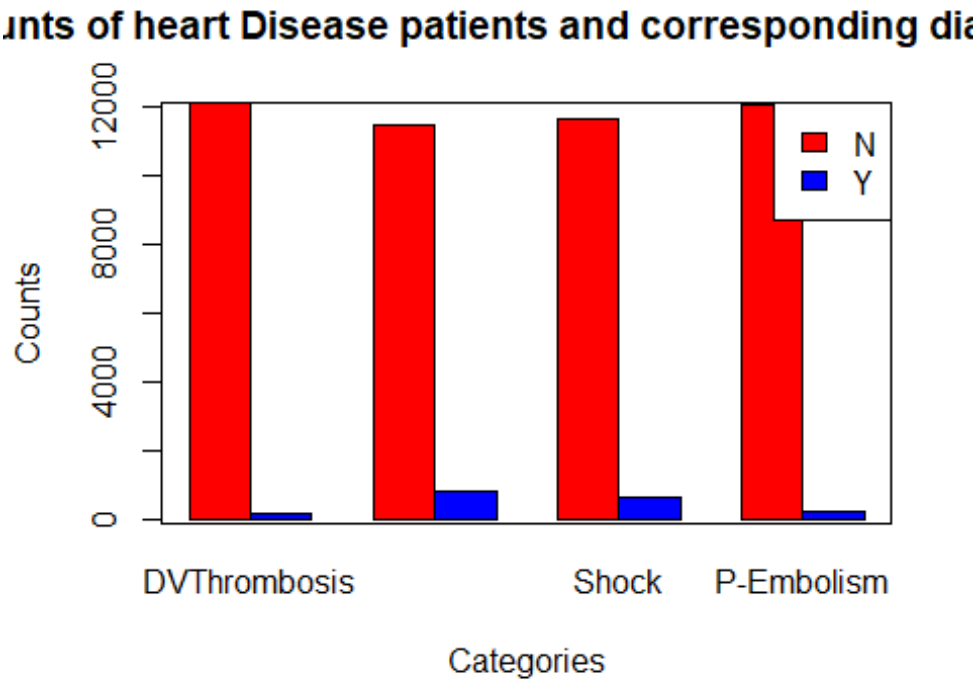
Set 5

```
count.dvt <- table(data1xplots$DVT)
count.cshock <- table(data1xplots$CARDIOGENIC.SHOCK)
count.shock <- table(data1xplots$SHOCK)
count.pulmonaryE <- table(data1xplots$PULMONARY.EMBOLISM)


dfcombt <- data.frame(unique.dim,count.dvt, count.cshock, count.shock,
count.pulmonaryE)

head(dfcombt)
```

```
##   unique.dim Var1  Freq Var1.1 Freq.1 Var1.2 Freq.2 Var1.3 Freq.3
## 1          0    0 12082      0  11459      0  11633      0  12052
## 2          1    1   162      1    785      1    611      1    192
```

```
dfcombt2 = dfcombt[-2]
dfcombt3 = dfcombt2[-3]
dfcombt4 = dfcombt3[-4]
dfcombt5 = dfcombt4[-5]

# head(dfcomb8)
names(dfcombt5) <- c("unique.dim", "DVThrombosis", "Cardio-Shock", "Sh
```

```
ock", "P-Embolism")
head(dfcombt5)

##   unique.dim DVThrombosis Cardio-Shock Shock P-Embolism
## 1          0        12082        11459 11633      12052
## 2          1          162          785   611        192

colours = c("red", "blue")
barplot(as.matrix(dfcombt5[-1]), main = "Counts of heart Disease patie
nts and corresponding diagnosis", ylab="Counts", xlab = "Categories",
beside = TRUE, col = colours)

box()

legend ('topright', fill = colours, legend=c('N', 'Y'))
```



#Removing unwanted columns

```
#Removing unwanted columns

# "SNO","MRD.No.","D.O.A","D.O.D",
### "AGE","GENDER","RURAL",>>>>>>>>>>>>>>>>>>>>>>>>>>>kept
# "TYPE.OF.ADMISSION.EMERGENCY.OPD","month.year", "DURATION.OF.STAY",
# "duration.of.intensive.unit.stay", "OUTCOME",
## "SMOKING","ALCOHOL","DM","HTN","CAD","PRIOR.CMP","CKD",>>>>>>>>>>>>
>>>>>>kept
```

```
## "HB","TLC","PLATELETS","GLUCOSE","UREA","CREATININE",>>>>>>>>>>>>>>
>>>>>>kept
# "BNP",
## "RAISED.CARDIAC.ENZYMES",>>>>>>>>>>>>>>>>>>>>kept
# "EF",>>>>>>>>>>>>>>>>>>>>>>>>>>kept
## "SEVERE.ANAEMIA",
## "ANAEMIA",>>>>>>>>>>>>>>>>>>>>>>>>kept
# "STABLE.ANGINA",
## "ACS",>>>>>>>>>>>>>>>>>>>>>>>>kept
##  "STEMI", >>>>>>>>>>>>>>>>>>>>>>>>kept
# "ATYPICAL.CHEST.PAIN",
## "HEART.FAILURE",>>>>>>>>>>>>>>>>>>>>>>>>kept
# "HFREF","HFNEF","VALVULAR","CHB","SSS",
## "AKI",>>>>>>>>>>>>>>>>>>>>>>>>kept
# "CVA.INFRACT","CVA.BLEED","AF","VT","PSVT","CONGENITAL","UTI",
# "NEURO.CARDIOGENIC.SYNCOPE","ORTHOSTATIC","INFECTIVE.ENDOCARDITIS",
# "DVT","CARDIOGENIC.SHOCK","SHOCK","PULMONARY.EMBOLISM","CHEST.INFECT
ION"

dfH1$SNO<-NULL
dfH1$MRD.No<-NULL
dfH1$D.O.A <-NULL
dfH1$D.O.D  <-NULL

#dfH1$TYPE.OF.ADMISSION.EMERGENCY.OPD <-NULL<<<<<sampling
dfH1$month.year  <-NULL
#dfH1$DURATION.OF.STAY  <-NULL <<<<<<< linear regression/sampling
dfH1$duration.of.intensive.unit.stay  <-NULL
#dfH1$OUTCOME  <-NULL <<<<<<<<< mulitnomial

dfH1$BNP <-NULL
#dfH1$RAISED.CARDIAC.ENZYMES <-NULL

# dfH1$SEVERE.ANAEMIA  <-NULL

# dfH1$STABLE.ANGINA <-NULL

# dfH1$STEMI <-NULL
dfH1$ATYPICAL.CHEST.PAIN  <-NULL

dfH1$HFREF  <-NULL
dfH1$HFNEF  <-NULL
dfH1$VALVULAR <-NULL
dfH1$CHB  <-NULL
dfH1$SSS  <-NULL
```

```
dfH1$CVA.INFRACT  <-NULL
dfH1$CVA.BLEED  <-NULL
dfH1$AF<-NULL
dfH1$VT <-NULL
dfH1$PSVT<- NULL
dfH1$CONGENITAL  <-NULL
dfH1$UTI  <-NULL
dfH1$NEURO.CARDIOGENIC.SYNCOPE  <-NULL
dfH1$ORTHOSTATIC  <-NULL
dfH1$INFECTIVE.ENDOCARDITIS  <-NULL
dfH1$DVT  <-NULL
dfH1$CARDIOGENIC.SHOCK <-NULL
dfH1$SHOCK  <-NULL
dfH1$PULMONARY.EMBOLISM  <-NULL
dfH1$CHEST.INFECTION  <-NULL


head(dfH1)

##   MRD.No. AGE GENDER RURAL TYPE.OF.ADMISSION.EMERGENCY.OPD DURATION
.OF.STAY
## 1  234735  81      M     R                                E
3
## 2  234696  65      M     R                                E
5
## 3  234882  53      M     U                                E
3
## 4  234635  67      F     U                                E
8
## 5  234486  60      F     U                                E
23
## 6  234675  44      M     U                                E
10
##     OUTCOME SMOKING ALCOHOL DM HTN CAD PRIOR.CMP CKD   HB  TLC PLAT
ELETS
## 1 DISCHARGE       0       0  1   0   0         0   0  9.5 16.1
337
## 2 DISCHARGE       0       1  0   1   1         0   0 13.7    9
149
## 3 DISCHARGE       0       0  1   0   1         0   0 10.6 14.7
329
## 4 DISCHARGE       0       0  0   1   1         0   0 12.8  9.9
286
## 5 DISCHARGE       0       0  0   1   0         1   0 13.6  9.1
26
## 6 DISCHARGE       0       0  1   1   1         1   0 13.5 22.3
```

```
322
##   GLUCOSE UREA CREATININE RAISED.CARDIAC.ENZYMES   EF SEVERE.ANAEMI
A ANAEMIA
## 1      80   34        0.9                      1   35
0       1
## 2     112   18        0.9                      0   42
0       0
## 3     187   93        2.3                      0 <NA>
0       0
## 4     130   27        0.6                      0   42
0       0
## 5     144   55       1.25                      0   16
0       0
## 6     217   51        0.9                      0   25
0       0
##   STABLE.ANGINA ACS STEMI HEART.FAILURE AKI
## 1             0   1     0             1   0
## 2             0   0     0             0   0
## 3             0   0     0             1   1
## 4             0   0     0             0   0
## 5             0   0     0             0   0
## 6             0   1     0             1   0
```

```
names(dfH1)
```

```
##  [1] "MRD.No."                          "AGE"
##  [3] "GENDER"                           "RURAL"
##  [5] "TYPE.OF.ADMISSION.EMERGENCY.OPD" "DURATION.OF.STAY"
##  [7] "OUTCOME"                          "SMOKING"
##  [9] "ALCOHOL"                          "DM"
## [11] "HTN"                              "CAD"
## [13] "PRIOR.CMP"                        "CKD"
## [15] "HB"                               "TLC"
## [17] "PLATELETS"                        "GLUCOSE"
## [19] "UREA"                             "CREATININE"
## [21] "RAISED.CARDIAC.ENZYMES"           "EF"
## [23] "SEVERE.ANAEMIA"                   "ANAEMIA"
## [25] "STABLE.ANGINA"                    "ACS"
## [27] "STEMI"                            "HEART.FAILURE"
## [29] "AKI"
```

```
dim(dfH1)
```

```
## [1] 12244    29
```

```
str(dfH1)
```

```
## 'data.frame':    12244 obs. of  29 variables:
##  $ MRD.No.                          : chr  "234735" "234696" "234882"
"234635" ...
##  $ AGE                              : int  81 65 53 67 60 44 56 47 65
59 ...
##  $ GENDER                           : chr  "M" "M" "M" "F" ...
##  $ RURAL                            : chr  "R" "R" "U" "U" ...
##  $ TYPE.OF.ADMISSION.EMERGENCY.OPD: chr  "E" "E" "E" "E" ...
##  $ DURATION.OF.STAY                 : int  3 5 3 8 23 10 6 13 3 3 ...
##  $ OUTCOME                          : chr  "DISCHARGE" "DISCHARGE" "D
ISCHARGE" "DISCHARGE" ...
##  $ SMOKING                          : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ ALCOHOL                          : int  0 1 0 0 0 0 0 1 0 0 ...
##  $ DM                               : int  1 0 1 0 0 1 1 1 0 1 ...
##  $ HTN                              : int  0 1 0 1 1 1 1 1 1 1 ...
##  $ CAD                              : int  0 1 1 1 0 1 1 0 0 1 ...
##  $ PRIOR.CMP                        : int  0 0 0 0 1 1 1 0 0 0 ...
##  $ CKD                              : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ HB                               : chr  "9.5" "13.7" "10.6" "12.8"
...
##  $ TLC                              : chr  "16.1" "9" "14.7" "9.9" ..
.
##  $ PLATELETS                        : chr  "337" "149" "329" "286" ..
.
##  $ GLUCOSE                          : chr  "80" "112" "187" "130" ...
##  $ UREA                             : chr  "34" "18" "93" "27" ...
##  $ CREATININE                       : chr  "0.9" "0.9" "2.3" "0.6" ..
.
##  $ RAISED.CARDIAC.ENZYMES           : int  1 0 0 0 0 0 0 0 0 0 ...
##  $ EF                               : chr  "35" "42" NA "42" ...
##  $ SEVERE.ANAEMIA                   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ ANAEMIA                          : int  1 0 0 0 0 0 0 0 0 0 ...
##  $ STABLE.ANGINA                    : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ ACS                              : int  1 0 0 0 0 1 1 0 1 0 ...
##  $ STEMI                            : int  0 0 0 0 0 0 1 0 0 0 ...
##  $ HEART.FAILURE                    : int  1 0 1 0 0 1 1 0 1 0 ...
##  $ AKI                              : int  0 0 1 0 0 0 0 0 0 0 ...

#Removing unwanted columns
# dfH2 <- dfH1[ -c(1:4,8:12,26,29,31,34,36:40,42:56) ] # removing colu
mns by index
#
# head(dfH2)
# names(dfH2)
# dim(dfH2)
# str(dfH2)
```

#removing all the rows with missing or NA values as we have found lots of missing values exists in our dataset.

```
#removing all the rows with missing or NA values.
#data1=na.omit(dfH2)
data1=na.omit(dfH1)
summary(data1)
```

```
##    MRD.No.                AGE             GENDER              RURAL
##  Length:10243       Min.   :  4.00   Length:10243       Length:1024
3
##  Class :character   1st Qu.: 53.00   Class :character   Class :char
acter
##  Mode  :character   Median : 62.00   Mode  :character   Mode  :char
acter
##                     Mean   : 61.07
##                     3rd Qu.: 70.00
##                     Max.   :110.00
##  TYPE.OF.ADMISSION.EMERGENCY.OPD DURATION.OF.STAY   OUTCOME
##  Length:10243                    Min.   : 1.000   Length:10243
##  Class :character                1st Qu.: 3.000   Class :character
##  Mode  :character                Median : 5.000   Mode  :character
##                                  Mean   : 6.572
##                                  3rd Qu.: 8.000
##                                  Max.   :98.000
##     SMOKING            ALCOHOL              DM                HTN
##  Min.   :0.00000   Min.   :0.00000   Min.   :0.0000   Min.   :0.000
0
##  1st Qu.:0.00000   1st Qu.:0.00000   1st Qu.:0.0000   1st Qu.:0.000
0
##  Median :0.00000   Median :0.00000   Median :0.0000   Median :0.000
0
##  Mean   :0.05467   Mean   :0.07254   Mean   :0.3284   Mean   :0.481
5
##  3rd Qu.:0.00000   3rd Qu.:0.00000   3rd Qu.:1.0000   3rd Qu.:1.000
0
##  Max.   :1.00000   Max.   :1.00000   Max.   :1.0000   Max.   :1.000
0
##       CAD            PRIOR.CMP            CKD                HB
##  Min.   :0.0000   Min.   :0.0000   Min.   :0.00000   Length:10243
##  1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.00000   Class :charact
er
##  Median :1.0000   Median :0.0000   Median :0.00000   Mode  :charact
er
##  Mean   :0.6729   Mean   :0.1551   Mean   :0.08484
##  3rd Qu.:1.0000   3rd Qu.:0.0000   3rd Qu.:0.00000
```

```
##  Max.   :1.0000   Max.   :1.0000   Max.   :1.00000
##      TLC              PLATELETS           GLUCOSE              UREA
##  Length:10243       Length:10243       Length:10243       Length:10
243
##  Class :character   Class :character   Class :character   Class :ch
aracter
##  Mode  :character   Mode  :character   Mode  :character   Mode  :ch
aracter
##
##
##
##   CREATININE        RAISED.CARDIAC.ENZYMES      EF            SEVER
E.ANAEMIA
##  Length:10243       Min.   :0.0000         Length:10243       Min.
:0.00000
##  Class :character   1st Qu.:0.0000         Class :character   1st Q
u.:0.00000
##  Mode  :character   Median :0.0000         Mode  :character   Media
n :0.00000
##                     Mean   :0.2265                            Mean
:0.01728
##                     3rd Qu.:0.0000                            3rd Q
u.:0.00000
##                     Max.   :1.0000                            Max.
:1.00000
##     ANAEMIA       STABLE.ANGINA         ACS              STEMI
##  Min.   :0.0000   Min.   :0.0000   Min.   :0.0000   Min.   :0.000
##  1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.000
##  Median :0.0000   Median :0.0000   Median :0.0000   Median :0.000
##  Mean   :0.1667   Mean   :0.0825   Mean   :0.4048   Mean   :0.165
##  3rd Qu.:0.0000   3rd Qu.:0.0000   3rd Qu.:1.0000   3rd Qu.:0.000
##  Max.   :1.0000   Max.   :1.0000   Max.   :1.0000   Max.   :1.000
##  HEART.FAILURE         AKI
##  Min.   :0.0000   Min.   :0.0000
##  1st Qu.:0.0000   1st Qu.:0.0000
##  Median :0.0000   Median :0.0000
##  Mean   :0.2858   Mean   :0.2123
##  3rd Qu.:1.0000   3rd Qu.:0.0000
##  Max.   :1.0000   Max.   :1.0000
```

```
dim(data1)
```

```
## [1] 10243    29
```

```
#Converting Character to numeric

data1$GLUCOSE <- as.numeric(data1$GLUCOSE)
```

```r
data1$HB <- as.numeric(data1$HB)
data1$EF <- as.numeric(data1$EF)
data1$TLC <- as.numeric(data1$TLC)
data1$PLATELETS <- as.numeric(data1$PLATELETS)
data1$UREA <- as.numeric(data1$UREA)
data1$CREATININE <- as.numeric(data1$CREATININE)
data1$EF <- as.numeric(data1$EF)

#Converting int to factor

data1$HEART.FAILURE <- as.factor(data1$HEART.FAILURE)
data1$AKI <- as.factor(data1$AKI)
data1$ACS <- as.factor(data1$ACS)
data1$CKD <- as.factor(data1$CKD)
data1$SMOKING <- as.factor(data1$SMOKING)
data1$ALCOHOL <- as.factor(data1$ALCOHOL)
data1$DM <- as.factor(data1$DM)
data1$PRIOR.CMP <- as.factor(data1$PRIOR.CMP)
data1$HTN <- as.factor(data1$HTN)
data1$CAD <- as.factor(data1$CAD)
data1$ANAEMIA <- as.factor(data1$ANAEMIA)
data1$RAISED.CARDIAC.ENZYMES <- as.factor(data1$RAISED.CARDIAC.ENZYMES
)
data1$STABLE.ANGINA <- as.factor(data1$STABLE.ANGINA)
data1$STEMI <- as.factor(data1$STEMI)
data1$TYPE.OF.ADMISSION.EMERGENCY.OPD<-as.factor(data1$TYPE.OF.ADMISSI
ON.EMERGENCY.OPD)

# data1$BNP <- as.numeric(data1$BNP)
data1$OUTCOME <- as.factor(data1$OUTCOME)
# data1$SEVERE.ANAEMIA <- as.factor(data1$SEVERE.ANAEMIA)
# data1$ATYPICAL.CHEST.PAIN <- as.factor(data1$ATYPICAL.CHEST.PAIN)
# data1$HFREF <- as.factor(data1$HFREF)
# data1$HFNEF <- as.factor(data1$HFNEF)
# data1$VALVULAR <- as.factor(data1$VALVULAR)
# data1$CHB <- as.factor(data1$CHB)
# data1$SSS <- as.factor(data1$SSS)
# data1$CVA.INFRACT <- as.factor(data1$CVA.INFRACT)
# data1$CVA.BLEED <- as.factor(data1$CVA.BLEED)
# data1$AF <- as.factor(data1$AF)
# data1$VT <- as.factor(data1$VT)
# data1$PSVT <- as.factor(data1$PSVT)
# data1$CONGENITAL <- as.factor(data1$CONGENITAL)
# data1$UTI <- as.factor(data1$UTI)
# data1$NEURO.CARDIOGENIC.SYNCOPE <- as.factor(data1$NEURO.CARDIOGENIC
.SYNCOPE)
```

```r
# data1$ORTHOSTATIC <- as.factor(data1$ORTHOSTATIC)
# data1$INFECTIVE.ENDOCARDITIS <- as.factor(data1$INFECTIVE.ENDOCARDIT
IS)
# data1$DVT <- as.factor(data1$DVT)
# data1$CARDIOGENIC.SHOCK <- as.factor(data1$CARDIOGENIC.SHOCK)
# data1$SHOCK <- as.factor(data1$SHOCK)
# data1$PULMONARY.EMBOLISM <- as.factor(data1$PULMONARY.EMBOLISM)
# data1$CHEST.INFECTION <- as.factor(data1$CHEST.INFECTION)


summary(data1)
```

```
##    MRD.No.               AGE            GENDER             RURAL
##  Length:10243       Min.   :  4.00   Length:10243       Length:1024
3
##  Class :character   1st Qu.: 53.00   Class :character   Class :char
acter
##  Mode  :character   Median : 62.00   Mode  :character   Mode  :char
acter
##                     Mean   : 61.07
##                     3rd Qu.: 70.00
##                     Max.   :110.00
##
##  TYPE.OF.ADMISSION.EMERGENCY.OPD DURATION.OF.STAY      OUTCOME
SMOKING
##  E:7290                          Min.   : 1.000   DAMA     : 456
0:9683
##  O:2953                          1st Qu.: 3.000   DISCHARGE:9177
1: 560
##                                  Median : 5.000   EXPIRY   : 610
##                                  Mean   : 6.572
##                                  3rd Qu.: 8.000
##                                  Max.   :98.000
##
##  ALCOHOL   DM       HTN      CAD      PRIOR.CMP CKD            HB
##  0:9500   0:6879   0:5311   0:3350   0:8654    0:9374   Min.   : 3.
00
##  1: 743   1:3364   1:4932   1:6893   1:1589    1: 869   1st Qu.:10.
80
##                                                       Median :12.
50
##                                                       Mean   :12.
33
##                                                       3rd Qu.:13.
90
##                                                       Max.   :26.
```

```
50
##                                                          NA's   :2
##       TLC              PLATELETS          GLUCOSE           UREA
##  Min.   :  0.3   Min.   :   0.58   Min.   :  1.2   Min.   :  0.10
##  1st Qu.:  8.0   1st Qu.: 172.00   1st Qu.:106.0   1st Qu.: 25.00
##  Median : 10.2   Median : 226.00   Median :136.5   Median : 35.00
##  Mean   : 11.6   Mean   : 238.27   Mean   :164.3   Mean   : 47.77
##  3rd Qu.: 13.6   3rd Qu.: 288.00   3rd Qu.:196.0   3rd Qu.: 55.00
##  Max.   :261.0   Max.   :1111.00   Max.   :888.0   Max.   :495.00
##  NA's   :2       NA's   :5         NA's   :49      NA's   :1
##    CREATININE     RAISED.CARDIAC.ENZYMES       EF         SEVERE.ANA
EMIA
##  Min.   : 0.065   0:7923                Min.   :14.00   Min.   :0.
00000
##  1st Qu.: 0.760   1:2320                1st Qu.:32.00   1st Qu.:0.
00000
##  Median : 0.970                         Median :44.00   Median :0.
00000
##  Mean   : 1.307                         Mean   :44.05   Mean   :0.
01728
##  3rd Qu.: 1.390                         3rd Qu.:60.00   3rd Qu.:0.
00000
##  Max.   :15.630                         Max.   :60.00   Max.   :1.
00000
##  NA's   :2                              NA's   :71
##  ANAEMIA  STABLE.ANGINA ACS      STEMI    HEART.FAILURE AKI
##  0:8535   0:9398        0:6097   0:8553   0:7316        0:8068
##  1:1708   1: 845        1:4146   1:1690   1:2927        1:2175
##
##
##
##
##
```

Changing ACS, AKI, Heart failure values from (0 , 1) to (N,Y)

```
#Make variables into Factors
# unique(data1$HEART.FAILURE)
# unique(data1$ACS)
# unique(data1$AKI)
#
# # Changing the factor and checking the order
# data1$HEART.FAILURE<-as.factor(data1$HEART.FAILURE)
# data1$HEART.FAILURE<-factor(data1$HEART.FAILURE, levels=c("0","1"),
#                       labels=c("N", "Y"))
#
```

```
#
# data1$ACS<-as.factor(data1$ACS)
# data1$ACS<-factor(data1$ACS, levels=c("0","1"),
#                       labels=c("N", "Y"))
#
#
# data1$AKI<-as.factor(data1$AKI)
# data1$AKI<-factor(data1$AKI, levels=c("0","1"),
#                       labels=c("N", "Y"))
#
#
# table(data1$ACS)
# table(data1$AKI)
# table(data1$HEART.FAILURE)

# table(data1$GENDER)
# unique(data1$GENDER)
#
# table(data1$RURAL)
# unique(data1$RURAL)
#
# table(data1$HEART.FAILURE)
# unique(data1$HEART.FAILURE)
#
# table(data1$AKI)
# unique(data1$AKI)
#
# table(data1$ACS)
# unique(data1$ACS)

# we found some empty string exists in few columns
# So replacing empty string with NA
data1[data1=="EMPTY"]<-NA

# and finally removing remaining NA values and named dataset as dataCl
ean
dataClean1=na.omit(data1)
summary(dataClean1)

##    MRD.No.               AGE           GENDER             RURAL
##  Length:10125       Min.   :  4.0   Length:10125       Length:10125
##  Class :character   1st Qu.: 53.0   Class :character   Class :chara
cter
##  Mode  :character   Median : 62.0   Mode  :character   Mode  :chara
cter
##                     Mean   : 61.1
##                     3rd Qu.: 70.0
```

```
##                       Max.   :110.0
##  TYPE.OF.ADMISSION.EMERGENCY.OPD DURATION.OF.STAY      OUTCOME
SMOKING
##  E:7236                          Min.   : 1.000   DAMA     : 446
0:9571
##  O:2889                          1st Qu.: 3.000   DISCHARGE:9086
1: 554
##                                  Median : 6.000   EXPIRY   : 593
##                                  Mean   : 6.593
##                                  3rd Qu.: 8.000
##                                  Max.   :98.000
##  ALCOHOL  DM       HTN      CAD      PRIOR.CMP CKD            HB
##  0:9390   0:6802   0:5250   0:3282   0:8544    0:9274   Min.   : 3.
00
##  1: 735   1:3323   1:4875   1:6843   1:1581    1: 851   1st Qu.:10.
80
##                                                       Median :12.
50
##                                                       Mean   :12.
33
##                                                       3rd Qu.:13.
90
##                                                       Max.   :22.
00
##       TLC            PLATELETS          GLUCOSE           UREA
##  Min.   :  0.30   Min.   :   1.38   Min.   :  1.2   Min.   :  0.10
##  1st Qu.:  8.00   1st Qu.: 173.00   1st Qu.:106.0   1st Qu.: 25.00
##  Median : 10.20   Median : 226.00   Median :137.0   Median : 35.00
##  Mean   : 11.62   Mean   : 238.93   Mean   :164.3   Mean   : 47.52
##  3rd Qu.: 13.60   3rd Qu.: 288.00   3rd Qu.:196.0   3rd Qu.: 55.00
##  Max.   :261.00   Max.   :1111.00   Max.   :888.0   Max.   :450.00
##    CREATININE     RAISED.CARDIAC.ENZYMES       EF         SEVERE.ANA
EMIA
##  Min.   : 0.065   0:7815                 Min.   :14.00   Min.   :0.
00000
##  1st Qu.: 0.760   1:2310                 1st Qu.:32.00   1st Qu.:0.
00000
##  Median : 0.960                          Median :44.00   Median :0.
00000
##  Mean   : 1.303                          Mean   :44.06   Mean   :0.
01719
##  3rd Qu.: 1.390                          3rd Qu.:60.00   3rd Qu.:0.
00000
##  Max.   :15.630                          Max.   :60.00   Max.   :1.
00000
##  ANAEMIA  STABLE.ANGINA ACS      STEMI    HEART.FAILURE AKI
```

```
##  0:8433   0:9293        0:5995   0:8441   0:7217        0:7981
##  1:1692   1: 832        1:4130   1:1684   1:2908        1:2144
##
##
##
##
```

```
dim(dataClean1)
```

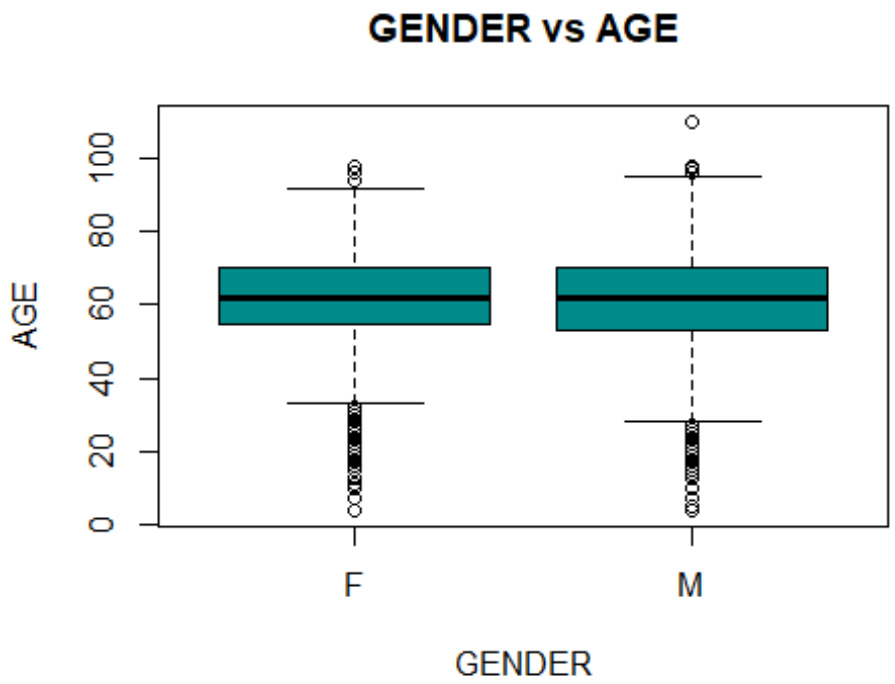
```
## [1] 10125    29
```

```
# Loading the png file for better visualization quality for cleaned da
taset

# d=head(dataClean1)
# knitr::kable(df, caption='Heading of cleaned dataset', align="lrcrcc
ccc")
# knitr::include_graphics("heatmap.png")
```

Observing some visualization of the dataset

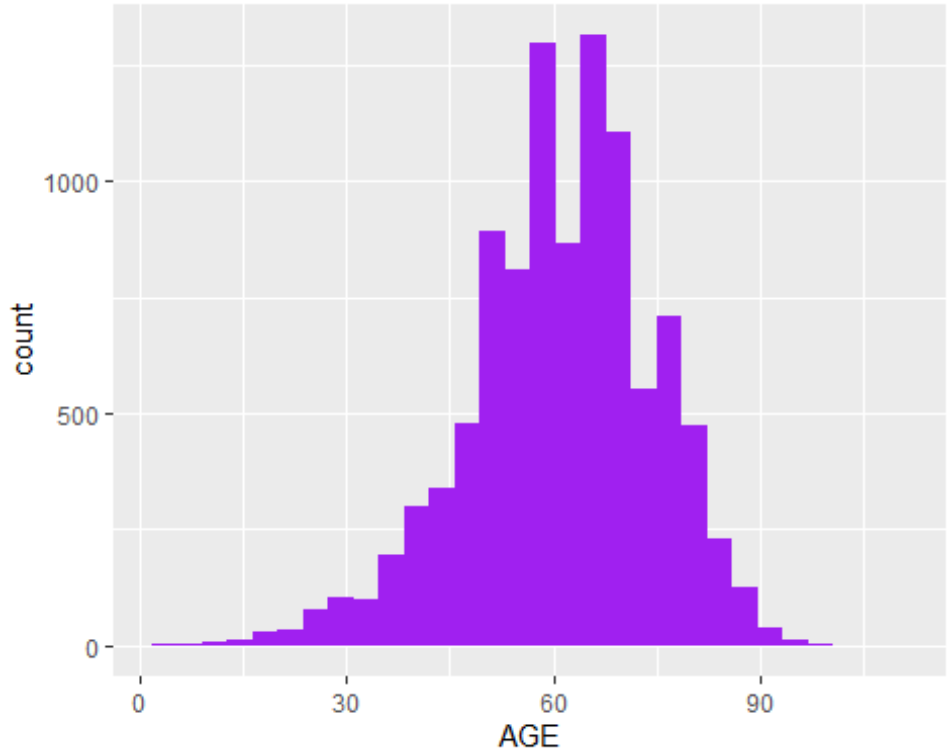
###Boxplots:

```
boxplot(AGE ~ GENDER, dataClean1, col = "dark cyan",main="GENDER vs AG
E")
```

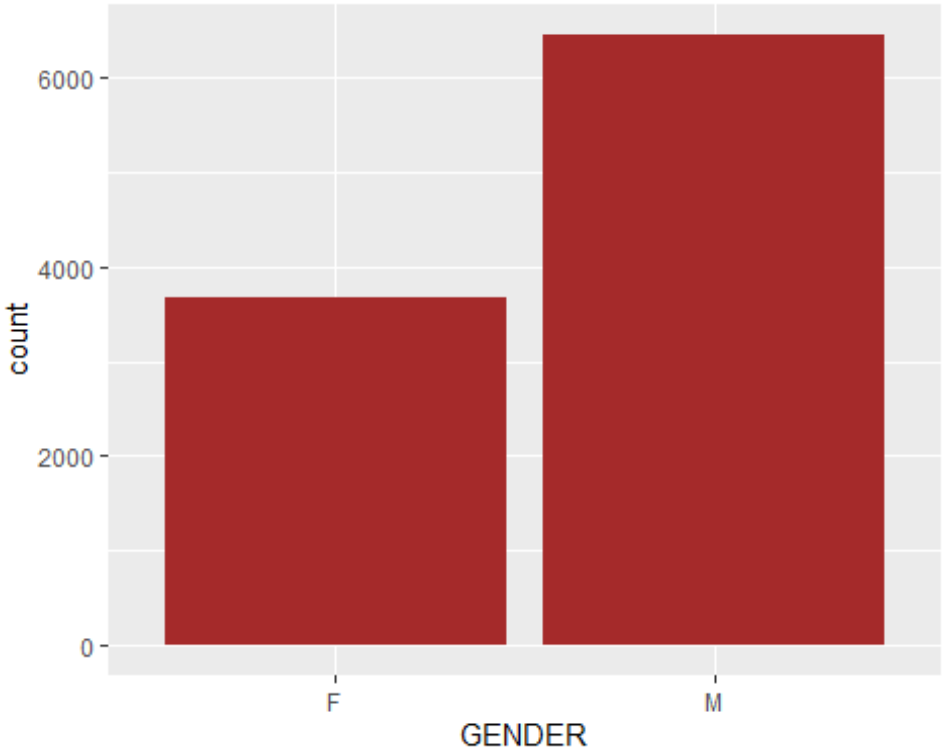


Barplots:

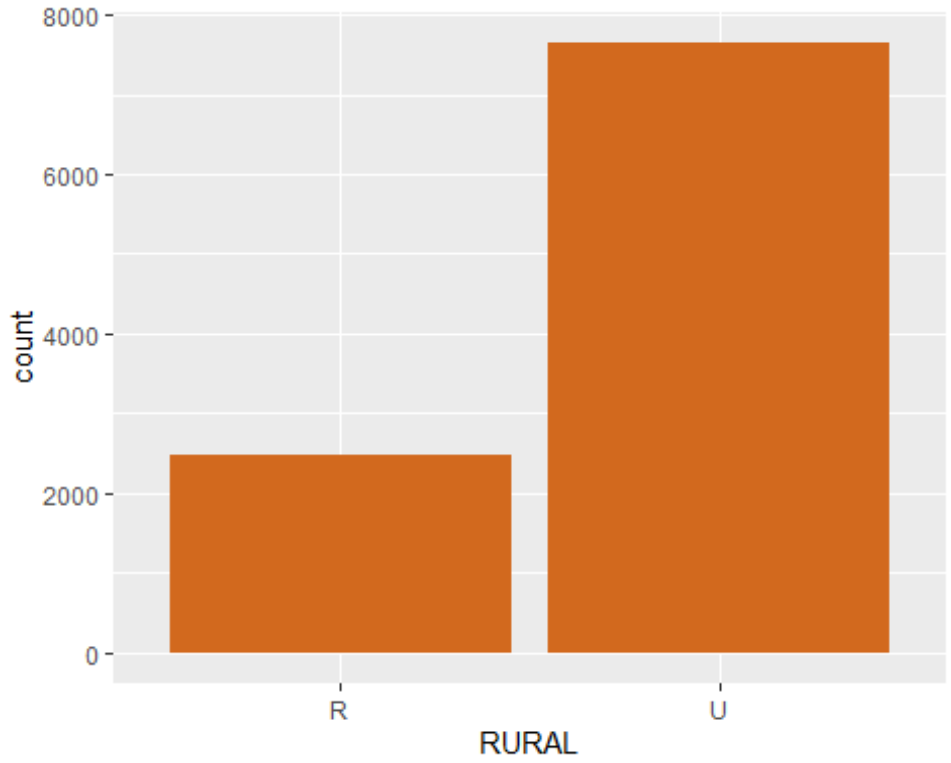
```
library(ggplot2)
ggplot(data = dataClean1, aes(x = AGE))+
  geom_bar(stat = "bin",fill = "purple")+theme_grey()
```




```
ggplot(dataClean1, aes(x=GENDER))+
  geom_bar(stat="count", fill="brown")+
  theme_grey()
```




```
ggplot(dataClean1, aes(x=RURAL))+
  geom_bar(stat="count", fill="chocolate")+
  theme_grey()
```



Put your respective analysis group chunk below.

```
# Just checking the cleanned dataset
summary(dataClean1)
```

```
##    MRD.No.                AGE           GENDER             RURAL
##  Length:10125       Min.   :  4.0   Length:10125       Length:10125
##  Class :character   1st Qu.: 53.0   Class :character   Class :chara
cter
##  Mode  :character   Median : 62.0   Mode  :character   Mode  :chara
cter
##                     Mean   : 61.1
##                     3rd Qu.: 70.0
##                     Max.   :110.0
##  TYPE.OF.ADMISSION.EMERGENCY.OPD DURATION.OF.STAY      OUTCOME
SMOKING
##  E:7236                          Min.   : 1.000   DAMA     : 446
0:9571
##  O:2889                          1st Qu.: 3.000   DISCHARGE:9086
1: 554
##                                  Median : 6.000   EXPIRY   : 593
##                                  Mean   : 6.593
##                                  3rd Qu.: 8.000
##                                  Max.   :98.000
##  ALCOHOL   DM       HTN      CAD      PRIOR.CMP CKD            HB
```

```
##  0:9390   0:6802   0:5250   0:3282   0:8544    0:9274   Min.   : 3.
00
##  1: 735   1:3323   1:4875   1:6843   1:1581    1: 851   1st Qu.:10.
80
##                                                         Median :12.
50
##                                                         Mean   :12.
33
##                                                         3rd Qu.:13.
90
##                                                         Max.   :22.
00
##       TLC            PLATELETS          GLUCOSE           UREA
##  Min.   :  0.30   Min.   :   1.38   Min.   :  1.2   Min.   :  0.10
##  1st Qu.:  8.00   1st Qu.: 173.00   1st Qu.:106.0   1st Qu.: 25.00
##  Median : 10.20   Median : 226.00   Median :137.0   Median : 35.00
##  Mean   : 11.62   Mean   : 238.93   Mean   :164.3   Mean   : 47.52
##  3rd Qu.: 13.60   3rd Qu.: 288.00   3rd Qu.:196.0   3rd Qu.: 55.00
##  Max.   :261.00   Max.   :1111.00   Max.   :888.0   Max.   :450.00
##    CREATININE     RAISED.CARDIAC.ENZYMES       EF         SEVERE.ANA
EMIA
##  Min.   : 0.065   0:7815                 Min.   :14.00   Min.   :0.
00000
##  1st Qu.: 0.760   1:2310                 1st Qu.:32.00   1st Qu.:0.
00000
##  Median : 0.960                          Median :44.00   Median :0.
00000
##  Mean   : 1.303                          Mean   :44.06   Mean   :0.
01719
##  3rd Qu.: 1.390                          3rd Qu.:60.00   3rd Qu.:0.
00000
##  Max.   :15.630                          Max.   :60.00   Max.   :1.
00000
##  ANAEMIA  STABLE.ANGINA ACS      STEMI    HEART.FAILURE AKI
##  0:8433   0:9293        0:5995   0:8441   0:7217        0:7981
##  1:1692   1: 832        1:4130   1:1684   1:2908        1:2144
##
##
##
##
```

```
names(dataClean1)
```

```
##  [1] "MRD.No."                          "AGE"
##  [3] "GENDER"                           "RURAL"
##  [5] "TYPE.OF.ADMISSION.EMERGENCY.OPD" "DURATION.OF.STAY"
```

```
##  [7] "OUTCOME"                          "SMOKING"
##  [9] "ALCOHOL"                          "DM"
## [11] "HTN"                              "CAD"
## [13] "PRIOR.CMP"                        "CKD"
## [15] "HB"                               "TLC"
## [17] "PLATELETS"                        "GLUCOSE"
## [19] "UREA"                             "CREATININE"
## [21] "RAISED.CARDIAC.ENZYMES"           "EF"
## [23] "SEVERE.ANAEMIA"                   "ANAEMIA"
## [25] "STABLE.ANGINA"                    "ACS"
## [27] "STEMI"                            "HEART.FAILURE"
## [29] "AKI"
```

PART II - SAMPLING

#Applying SRS on the clean dataset with a sample size 1/3 of the population.

```
set.seed(10)
#taking the required sample with a sample size of 3000
N=10125  #population size
n=3000 #sample size
idx=sample(1:N,size = n, replace = FALSE) #taking the sample
datasrs=dataClean1[idx,]
names(datasrs)
```

```
##  [1] "MRD.No."                          "AGE"
##  [3] "GENDER"                           "RURAL"
##  [5] "TYPE.OF.ADMISSION.EMERGENCY.OPD"  "DURATION.OF.STAY"
##  [7] "OUTCOME"                          "SMOKING"
##  [9] "ALCOHOL"                          "DM"
## [11] "HTN"                              "CAD"
## [13] "PRIOR.CMP"                        "CKD"
## [15] "HB"                               "TLC"
## [17] "PLATELETS"                        "GLUCOSE"
## [19] "UREA"                             "CREATININE"
## [21] "RAISED.CARDIAC.ENZYMES"           "EF"
## [23] "SEVERE.ANAEMIA"                   "ANAEMIA"
## [25] "STABLE.ANGINA"                    "ACS"
## [27] "STEMI"                            "HEART.FAILURE"
## [29] "AKI"
```

```
dim(datasrs)
```

```
## [1] 3000   29
```

Finding out the population average and standard deviation of the sample with Duration of Stay as variable of interest:

```
library(survey)
mydata<-data.frame(datasrs,pw=rep(N/n,n),fpc=rep(N,n))
svy<-svydesign(id=~0, strata = NULL, weights=~pw, data = mydata, fpc=~
fpc)
dossrs<-svymean(~DURATION.OF.STAY, svy)
dossrs
```

```
##                       mean     SE
## DURATION.OF.STAY 6.5097 0.0695
```

#####So the population average comes out to be 6.5097 and standard deviation comes out to be 0.0695 with a population size of 3000 and variable of interest as $"DURATION.OF.STAY"$.

```
#finding out the confidence interval for the duration of stay
confiddos<-confint(dossrs,level = 0.95, df=degf(svy))
confiddos
```

```
##                       2.5 %   97.5 %
## DURATION.OF.STAY 6.373452 6.645882
```

##Applying stratified sampling to compare the outcomes with SRS using the same sample size

Using the same variable of interest as duration of stay and using type of admission as the stratum.

Rechecking the individual count of type of admission that will be used as the strata.

```
table(dataClean1$TYPE.OF.ADMISSION.EMERGENCY.OPD)
```

```
##
##    E    O
## 7236 2889
```

It is observed that 2/3 of the patients are emergency and 1/3 are outpatients.

Using proportional allocation to find out the size of the sample for each stratum.

```
N=10125  #population size
n=3000 #sample size
psize<-table(dataClean1$TYPE.OF.ADMISSION.EMERGENCY.OPD)
palloc<-n*psize/N #determining the proportional allocation for each st
ratum
palloc
```

```
##
##    E    O
## 2144  856
```

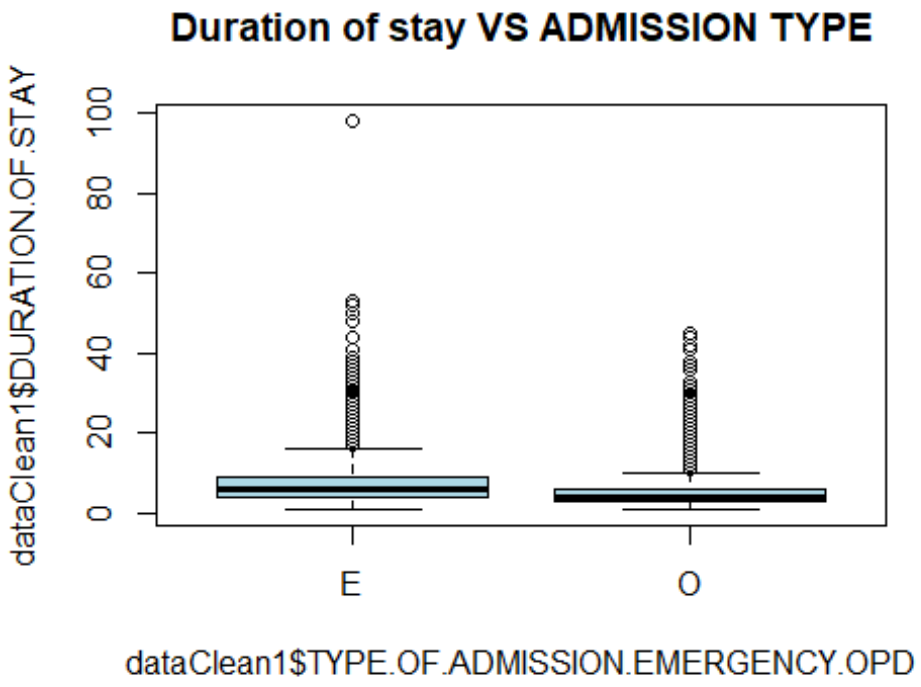
Rounding up number of sampled units in each stratum to the nearest integer.

```
intpalloc<-round(palloc) #round up the stratum proportions to the near
est integer
intpalloc

##
##    E    O
## 2144  856

sum(intpalloc)

## [1] 3000
```

Implementing SRS without replacement for type of admission with proportional allocation with n=3000 to create the sample.

```
set.seed(10)
library(sampling)
idx1<-sampling:::strata(dataClean1,stratanames=c("TYPE.OF.ADMISSION.EM
ERGENCY.OPD"),size=c(2144,856), method="srswor")
datastrat1<-getdata(dataClean1,idx1)
table(datastrat1$TYPE.OF.ADMISSION.EMERGENCY.OPD)

##
##    E    O
## 2144  856

dim(datastrat1)

## [1] 3000   32

# Checking that no probabilities are 0
sum(datastrat1$Prob<=0)

## [1] 0

#Calculating the sampling weights
datastrat1$sampwt<-1/datastrat1$Prob

# Checking that the sampling weights sum to the population sizes for e
ach stratum
tapply(datastrat1$sampwt,datastrat1$TYPE.OF.ADMISSION.EMERGENCY.OPD,su
m)

##    E    O
## 7236 2889

#Applying stratified sampling with variable of interest as duration of
stay and type of admission as stratum
datastrat1=data.frame(datastrat1, pw=datastrat1$sampwt, fpc=c(rep(7236
```

```
,2144),rep(2889,856)))
svy1<-svydesign(id=~1,strata = ~TYPE.OF.ADMISSION.EMERGENCY.OPD, weigh
ts = ~pw, data = datastrat1, fpc=~fpc)
dosstrat<-svymean(~DURATION.OF.STAY, svy1)
dosstrat

##                       mean     SE
## DURATION.OF.STAY 6.721 0.0749

boxplot(dataClean1$DURATION.OF.STAY~dataClean1$TYPE.OF.ADMISSION.EMERG
ENCY.OPD,main="Duration of stay VS ADMISSION TYPE",col="light blue")
```




```
#finding out the confidence interval
confiddos1<-confint(dosstrat,level = 0.95, df=degf(svy1))
confiddos1

##                        2.5 %   97.5 %
## DURATION.OF.STAY 6.574042 6.867958
```

#####So the population average comes out to be 6.721 and standard deviation comes out to be 0.0749 which is higher than SRS.

```
#Using Anova table for further comparison

summary(aov(formula=DURATION.OF.STAY~TYPE.OF.ADMISSION.EMERGENCY.OPD,
data = dataClean1))
```

```
##                                      Df Sum Sq Mean Sq F value Pr(>F)
## TYPE.OF.ADMISSION.EMERGENCY.OPD       1   7783    7783   344.2 <2e-16
***
## Residuals                         10123 228888      23
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(aov(formula=DURATION.OF.STAY~TYPE.OF.ADMISSION.EMERGENCY.OPD,
data = datastrat1))
```

```
##                                        Df Sum Sq Mean Sq F value Pr(>F)
## TYPE.OF.ADMISSION.EMERGENCY.OPD         1   1989  1989.4   83.07 <2e-16
***
## Residuals                           2998  71798    23.9
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Both the p-value for SRS and stratified sampling comes out lower.However, the SSB is very low compared to SSW to stratified sampling is not a good fit for this model.

```
#Comparing the sample mean and SE from SRS and stratified sampling wit
h the population mean and SE
popmean=sum((dataClean1$DURATION.OF.STAY)/N)
popsd=sd(dataClean1$DURATION.OF.STAY)
c(dossrs,dosstrat,popmean)
```

```
## DURATION.OF.STAY DURATION.OF.STAY
##         6.509667         6.721000         6.593481
```

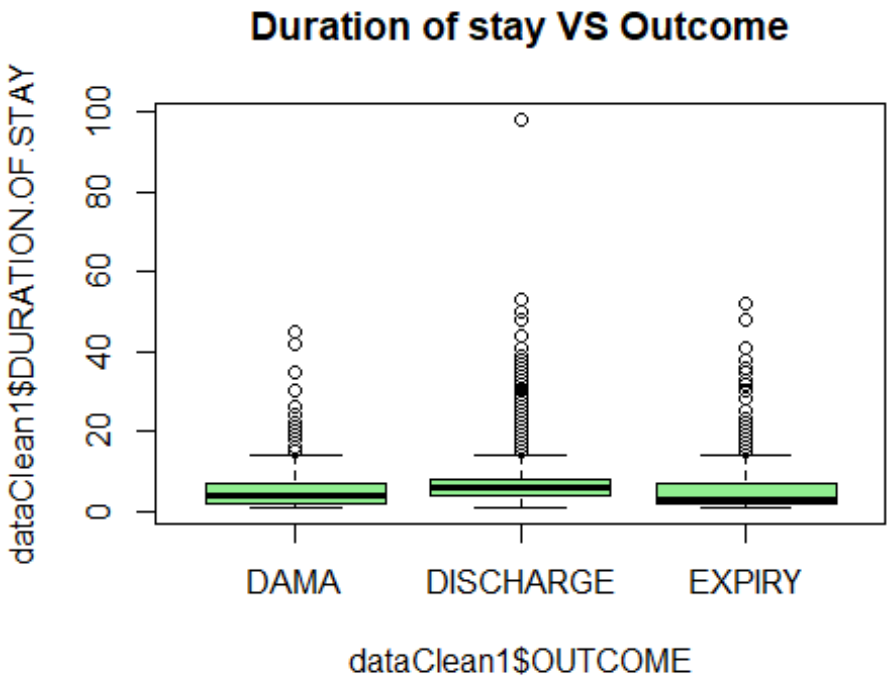
Rechecking the individual count of outcome that will be used as the strata.

```
table(dataClean1$OUTCOME)
```

```
##
##      DAMA DISCHARGE    EXPIRY
##       446      9086       593
```

It is observed that 1/2 of the patients have been discharged normally and the remaining half are shared between expiry and forceful discharge.

```
boxplot(dataClean1$DURATION.OF.STAY~dataClean1$OUTCOME,main="Duration
of stay VS Outcome",col="light green")
```



Sorting the outcome in alphabetical order for simplification purpose:

```
outsort<-dataClean1[order(dataClean1$OUTCOME),]
unique(outsort$OUTCOME)

## [1] DAMA      DISCHARGE EXPIRY
## Levels: DAMA DISCHARGE EXPIRY
```

Using proportional allocation to find out the size of the sample for each stratum.

```
N=10125  #population size
n=3000 #sample size
psize1<-table(outsort$OUTCOME)
palloc1<-n*psize1/N #determining the proportional allocation for each
stratum
palloc1

##
##      DAMA DISCHARGE    EXPIRY
##  132.1481 2692.1481  175.7037
```

Rounding up number of sampled units in each stratum to the nearest integer.

```
intpalloc1<-round(palloc1) #round up the stratum proportions to the ne
arest integer
intpalloc1
```

```
##
##      DAMA DISCHARGE    EXPIRY
##       132      2692       176
```

```
sum(intpalloc1)
```

```
## [1] 3000
```

Implementing SRS without replacement for each outcome with proportional allocation with n=3000 to create the sample.

```
set.seed(10)
library(sampling)
idx2<-sampling:::strata(outsort,stratanames=c("OUTCOME"),size=c(132,26
92,176), method="srswor")
datastrat2<-getdata(outsort,idx2)
table(datastrat2$OUTCOME)
```

```
##
##      DAMA DISCHARGE    EXPIRY
##       132      2692       176
```

```
dim(datastrat2)
```

```
## [1] 3000   32
```

```
# Checking that no probabilities are 0
sum(datastrat2$Prob<=0)
```

```
## [1] 0
```

```
#Calculating the sampling weights
datastrat2$sampwt<-1/datastrat2$Prob

# Checking that the sampling weights sum to the population sizes for e
ach stratum
tapply(datastrat2$sampwt,datastrat2$OUTCOME,sum)
```

```
##      DAMA DISCHARGE    EXPIRY
##       446      9086       593
```

```
#Applying stratified sampling with variable of interest as duration of
stay and OUTCOME as stratum
datastrat2=data.frame(datastrat2, pw=datastrat2$sampwt, fpc=c(rep(446,
132),rep(9086,2692),rep(593,176)))
svy2<-svydesign(id=~1,strata = ~OUTCOME, weights = ~pw, data = datastr
at2, fpc=~fpc)
dosstrat2<-svymean(~DURATION.OF.STAY, svy2)
dosstrat2
```

```
##                       mean     SE
## DURATION.OF.STAY 6.4741 0.0705
```

```
#finding out the confidence interval
confiddos2<-confint(dosstrat2,level = 0.95, df=degf(svy1))
confiddos2
```

```
##                       2.5 %   97.5 %
## DURATION.OF.STAY 6.335919 6.612222
```

#####So the population average comes out to be 6.4741 and standard deviation comes out to be 0.0705 which is higher than SRS.

```
#Using Anova table for further comparison

summary(aov(formula=DURATION.OF.STAY~OUTCOME, data = dataClean1))
```

```
##                Df Sum Sq Mean Sq F value   Pr(>F)
## OUTCOME         2    874   437.1   18.77 7.34e-09 ***
## Residuals   10122 235797    23.3
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(aov(formula=DURATION.OF.STAY~OUTCOME, data = datastrat2))
```

```
##                Df Sum Sq Mean Sq F value   Pr(>F)
## OUTCOME         2    412  206.23   9.747 6.03e-05 ***
## Residuals    2997  63412   21.16
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##Comparing the sample mean and SE from SRS and stratified sampling wi
th the population mean and SE
```

The p-value for both SRS and stratified sampling comes out lower. However, SSB is very low compared to SSW to stratified sampling is not a good fit for this model.

```
popmean=sum((dataClean1$DURATION.OF.STAY)/N)
popsd=sd(dataClean1$DURATION.OF.STAY)
c(dossrs,dosstrat,popmean)
```

```
## DURATION.OF.STAY DURATION.OF.STAY
##         6.509667         6.721000         6.593481
```

The overall conclusion that can be drawn from this section are the following: 1. SRS is better than stratified sampling as the SSB from the anova table is negligible compared to the SSW for both auxilary variable. 2. Both the sampling mean are close to the population mean 3. Only

quantitative variables could be used as the variable of interest and duration of stay providing the best option.

```
#Dropping Type of Admission as it will not be needed for further analy
sis
dataClean1$TYPE.OF.ADMISSION.EMERGENCY.OPD <-NULL
dim(dataClean1)

## [1] 10125    28

dataClean=dataClean1
summary(dataClean)

##    MRD.No.               AGE           GENDER             RURAL
##  Length:10125       Min.   :  4.0   Length:10125       Length:10125
##  Class :character   1st Qu.: 53.0   Class :character   Class :chara
cter
##  Mode  :character   Median : 62.0   Mode  :character   Mode  :chara
cter
##                     Mean   : 61.1
##                     3rd Qu.: 70.0
##                     Max.   :110.0
##  DURATION.OF.STAY      OUTCOME     SMOKING  ALCOHOL  DM       HTN
CAD
##  Min.   : 1.000   DAMA     : 446   0:9571   0:9390   0:6802   0:525
0   0:3282
##  1st Qu.: 3.000   DISCHARGE:9086   1: 554   1: 735   1:3323   1:487
5   1:6843
##  Median : 6.000   EXPIRY   : 593
##  Mean   : 6.593
##  3rd Qu.: 8.000
##  Max.   :98.000
##  PRIOR.CMP CKD            HB              TLC           PLATELETS
##  0:8544    0:9274   Min.   : 3.00   Min.   :  0.30   Min.   :   1.3
8
##  1:1581    1: 851   1st Qu.:10.80   1st Qu.:  8.00   1st Qu.: 173.0
0
##                     Median :12.50   Median : 10.20   Median : 226.0
0
##                     Mean   :12.33   Mean   : 11.62   Mean   : 238.9
3
##                     3rd Qu.:13.90   3rd Qu.: 13.60   3rd Qu.: 288.0
0
##                     Max.   :22.00   Max.   :261.00   Max.   :1111.0
0
##     GLUCOSE           UREA          CREATININE     RAISED.CARDIAC.E
NZYMES
```

```
##  Min.   :  1.2   Min.   :  0.10   Min.   : 0.065   0:7815
##  1st Qu.:106.0   1st Qu.: 25.00   1st Qu.: 0.760   1:2310
##  Median :137.0   Median : 35.00   Median : 0.960
##  Mean   :164.3   Mean   : 47.52   Mean   : 1.303
##  3rd Qu.:196.0   3rd Qu.: 55.00   3rd Qu.: 1.390
##  Max.   :888.0   Max.   :450.00   Max.   :15.630
##        EF        SEVERE.ANAEMIA    ANAEMIA  STABLE.ANGINA ACS
STEMI
##  Min.   :14.00   Min.   :0.00000   0:8433   0:9293        0:5995
0:8441
##  1st Qu.:32.00   1st Qu.:0.00000   1:1692   1: 832        1:4130
1:1684
##  Median :44.00   Median :0.00000
##  Mean   :44.06   Mean   :0.01719
##  3rd Qu.:60.00   3rd Qu.:0.00000
##  Max.   :60.00   Max.   :1.00000
##  HEART.FAILURE AKI
##  0:7217        0:7981
##  1:2908        1:2144
##
##
##
##
```

PART III _ CHECKING INDEPENDECE OF CATEGORICAL VARIABLES USING CONTIGENCE TABLE

```
library(fmsb)

categoricalNames <-(c( "HEART.FAILURE",   "STEMI", "AKI",  "ACS",
                       "RAISED.CARDIAC.ENZYMES", "ANAEMIA", "PRIOR.CMP
",
                       "CAD", "HTN", "DM", "ALCOHOL", "SMOKING", "CKD"
))

# Adding the categorical variables to check independence in a new data
frame
dataClean_OnlyCategorical <- dataClean[, categoricalNames]
namesCategorical <- names(dataClean_OnlyCategorical)

# Just to Check if everything is inorder
names(dataClean_OnlyCategorical)
```

```
##  [1] "HEART.FAILURE"          "STEMI"                  "AKI"
##  [4] "ACS"                    "RAISED.CARDIAC.ENZYMES" "ANAEMIA"
##  [7] "PRIOR.CMP"              "CAD"                    "HTN"
```

```
## [10] "DM"                       "ALCOHOL"                  "SMOKING"
## [13] "CKD"

dim(dataClean_OnlyCategorical)

## [1] 10125    13

#namesCategorical[1]


# Number of pairs to carry out the tests
numPairsCheck = length(namesCategorical)-1
#print(numPairsCheck)


# Creating matrix to store pvalue values
pvalueMatrix = matrix(0, length(namesCategorical), length(namesCategor
ical))


# Looping to quantify the p-value of Test Based on the differences (2x
2 Contingency tables) for all the pairs
for(i in seq(1:numPairsCheck)){
  for(j in seq(i:numPairsCheck)){

    print(i)
    print(j)

    # Taking the names of Categorical Variable to compare
    name1 = namesCategorical[i]
    name2 = namesCategorical[j+1]

    if(name1 != name2){
      #print(name1)
      #print(name2)

      #Creating a dataset with the two selected variables
      dataTest<- dataClean_OnlyCategorical[, c(name1,  name2)]
      tableTest <- table(dataTest)

      #print(tableTest)

      # Print Variables to check independence
      print(c(name1, name2))


     #Test based on the risk difference
      risk1 = riskdifference(tableTest[1,1], tableTest[2,1], tableTest
```

```
[1,1]+
                                tableTest[1,2],  tableTest[2,1]+tableTe
st[2,2],
                               conf.level = 0.95)

      print(risk1$p.value)

      #Storing pvalue
      pvalueMatrix[i,j+1] = risk1$p.value
      pvalueMatrix[j+1,i] = risk1$p.value

      if(risk1$p.value >= 0.05){ #Just to show the cases that are not
independent at 5% level
        print(c("The variables:", name1, name2, "are NOT independent !
", "p-value = ", risk1$p.value))
      }

    }

  }

}
```

```
## [1] 1
## [1] 1
## [1] "HEART.FAILURE" "STEMI"
##                  Cases People at risk         Risk
## Exposed   5.974000e+03   7.217000e+03 8.277678e-01
## Unexposed 2.467000e+03   2.908000e+03 8.483494e-01
## Total     8.441000e+03   1.012500e+04 8.336790e-01
## [1] 0.01008817
## [1] 1
## [1] 2
## [1] "HEART.FAILURE" "AKI"
##                  Cases People at risk         Risk
## Exposed   6.043000e+03   7.217000e+03 8.373285e-01
## Unexposed 1.938000e+03   2.908000e+03 6.664374e-01
## Total     7.981000e+03   1.012500e+04 7.882469e-01
## [1] 0
## [1] 1
## [1] 3
## [1] "HEART.FAILURE" "ACS"
##                  Cases People at risk         Risk
## Exposed   4.360000e+03   7.217000e+03 6.041291e-01
## Unexposed 1.635000e+03   2.908000e+03 5.622421e-01
## Total     5.995000e+03   1.012500e+04 5.920988e-01
```

```
## [1] 0.0001135319
## [1] 1
## [1] 4
## [1] "HEART.FAILURE"          "RAISED.CARDIAC.ENZYMES"
##                   Cases People at risk         Risk
## Exposed   5.826000e+03   7.217000e+03 8.072606e-01
## Unexposed 1.989000e+03   2.908000e+03 6.839752e-01
## Total     7.815000e+03   1.012500e+04 7.718519e-01
## [1] 0
## [1] 1
## [1] 5
## [1] "HEART.FAILURE" "ANAEMIA"
##                   Cases People at risk         Risk
## Exposed   6.200000e+03   7.217000e+03 8.590827e-01
## Unexposed 2.233000e+03   2.908000e+03 7.678817e-01
## Total     8.433000e+03   1.012500e+04 8.328889e-01
## [1] 0
## [1] 1
## [1] 6
## [1] "HEART.FAILURE" "PRIOR.CMP"
##                   Cases People at risk         Risk
## Exposed   6.601000e+03   7.217000e+03 9.146460e-01
## Unexposed 1.943000e+03   2.908000e+03 6.681568e-01
## Total     8.544000e+03   1.012500e+04 8.438519e-01
## [1] 0
## [1] 1
## [1] 7
## [1] "HEART.FAILURE" "CAD"
##                   Cases People at risk         Risk
## Exposed   2.286000e+03   7.217000e+03 3.167521e-01
## Unexposed 9.960000e+02   2.908000e+03 3.425034e-01
## Total     3.282000e+03   1.012500e+04 3.241481e-01
## [1] 0.01297268
## [1] 1
## [1] 8
## [1] "HEART.FAILURE" "HTN"
##                   Cases People at risk         Risk
## Exposed   3.771000e+03   7.217000e+03 5.225163e-01
## Unexposed 1.479000e+03   2.908000e+03 5.085970e-01
## Total     5.250000e+03   1.012500e+04 5.185185e-01
## [1] 0.2048202
## [1] "The variables:"        "HEART.FAILURE"         "HTN"
## [4] "are NOT independent !" "p-value = "            "0.204820159580
512"
## [1] 1
## [1] 9
```

```
## [1] "HEART.FAILURE" "DM"
##                   Cases People at risk         Risk
## Exposed   4.972000e+03   7.217000e+03 6.889289e-01
## Unexposed 1.830000e+03   2.908000e+03 6.292985e-01
## Total     6.802000e+03   1.012500e+04 6.718025e-01
## [1] 1.287407e-08
## [1] 1
## [1] 10
## [1] "HEART.FAILURE" "ALCOHOL"
##                   Cases People at risk         Risk
## Exposed   6.637000e+03   7.217000e+03 9.196342e-01
## Unexposed 2.753000e+03   2.908000e+03 9.466988e-01
## Total     9.390000e+03   1.012500e+04 9.274074e-01
## [1] 2.572918e-07
## [1] 1
## [1] 11
## [1] "HEART.FAILURE" "SMOKING"
##                   Cases People at risk         Risk
## Exposed   6.784000e+03   7.217000e+03 9.400028e-01
## Unexposed 2.787000e+03   2.908000e+03 9.583906e-01
## Total     9.571000e+03   1.012500e+04 9.452840e-01
## [1] 7.399081e-05
## [1] 1
## [1] 12
## [1] "HEART.FAILURE" "CKD"
##                   Cases People at risk         Risk
## Exposed   6.737000e+03   7.217000e+03 9.334904e-01
## Unexposed 2.537000e+03   2.908000e+03 8.724209e-01
## Total     9.274000e+03   1.012500e+04 9.159506e-01
## [1] 0
## [1] 2
## [1] 1
## [1] 2
## [1] 2
## [1] "STEMI" "AKI"
##                   Cases People at risk         Risk
## Exposed   6.575000e+03   8.441000e+03 7.789361e-01
## Unexposed 1.406000e+03   1.684000e+03 8.349169e-01
## Total     7.981000e+03   1.012500e+04 7.882469e-01
## [1] 3.090725e-08
## [1] 2
## [1] 3
## [1] "STEMI" "ACS"
##                   Cases People at risk         Risk
## Exposed   5.995000e+03   8.441000e+03 7.102239e-01
## Unexposed 0.000000e+00   1.684000e+03 0.000000e+00
```

```
## Total     5.995000e+03   1.012500e+04 5.920988e-01
## [1] 0
## [1] 2
## [1] 4
## [1] "STEMI"                  "RAISED.CARDIAC.ENZYMES"
##                  Cases People at risk         Risk
## Exposed   6.616000e+03   8.441000e+03 7.837934e-01
## Unexposed 1.199000e+03   1.684000e+03 7.119952e-01
## Total     7.815000e+03   1.012500e+04 7.718519e-01
## [1] 1.655191e-09
## [1] 2
## [1] 5
## [1] "STEMI"   "ANAEMIA"
##                   Cases People at risk         Risk
## Exposed   6.931000e+03   8.441000e+03 8.211112e-01
## Unexposed 1.502000e+03   1.684000e+03 8.919240e-01
## Total     8.433000e+03   1.012500e+04 8.328889e-01
## [1] 2.220446e-16
## [1] 2
## [1] 6
## [1] "STEMI"     "PRIOR.CMP"
##                   Cases People at risk         Risk
## Exposed   7.091000e+03   8.441000e+03 8.400663e-01
## Unexposed 1.453000e+03   1.684000e+03 8.628266e-01
## Total     8.544000e+03   1.012500e+04 8.438519e-01
## [1] 0.01422813
## [1] 2
## [1] 7
## [1] "STEMI" "CAD"
##                   Cases People at risk         Risk
## Exposed   3.115000e+03   8.441000e+03 3.690321e-01
## Unexposed 1.670000e+02   1.684000e+03 9.916865e-02
## Total     3.282000e+03   1.012500e+04 3.241481e-01
## [1] 0
## [1] 2
## [1] 8
## [1] "STEMI" "HTN"
##                   Cases People at risk         Risk
## Exposed   4.244000e+03   8.441000e+03 5.027840e-01
## Unexposed 1.006000e+03   1.684000e+03 5.973872e-01
## Total     5.250000e+03   1.012500e+04 5.185185e-01
## [1] 5.837553e-13
## [1] 2
## [1] 9
## [1] "STEMI" "DM"
##                   Cases People at risk         Risk
```

```
## Exposed   5.656000e+03   8.441000e+03 6.700628e-01
## Unexposed 1.146000e+03   1.684000e+03 6.805226e-01
## Total     6.802000e+03   1.012500e+04 6.718025e-01
## [1] 0.4012728
## [1] "The variables:"        "STEMI"                 "DM"
## [4] "are NOT independent !" "p-value = "            "0.401272804491
875"
## [1] 2
## [1] 10
## [1] "STEMI"   "ALCOHOL"
##                   Cases People at risk         Risk
## Exposed   7.876000e+03   8.441000e+03 9.330648e-01
## Unexposed 1.514000e+03   1.684000e+03 8.990499e-01
## Total     9.390000e+03   1.012500e+04 9.274074e-01
## [1] 1.394616e-05
## [1] 2
## [1] 11
## [1] "STEMI"   "SMOKING"
##                   Cases People at risk         Risk
## Exposed   8.034000e+03   8.441000e+03 9.517830e-01
## Unexposed 1.537000e+03   1.684000e+03 9.127078e-01
## Total     9.571000e+03   1.012500e+04 9.452840e-01
## [1] 7.441099e-08
## [1] 3
## [1] 1
## [1] "AKI"   "STEMI"
##                   Cases People at risk         Risk
## Exposed   6.575000e+03   7.981000e+03 8.238316e-01
## Unexposed 1.866000e+03   2.144000e+03 8.703358e-01
## Total     8.441000e+03   1.012500e+04 8.336790e-01
## [1] 3.275835e-08
## [1] 3
## [1] 2
## [1] 3
## [1] 3
## [1] "AKI" "ACS"
##                   Cases People at risk         Risk
## Exposed   4.661000e+03   7.981000e+03 5.840120e-01
## Unexposed 1.334000e+03   2.144000e+03 6.222015e-01
## Total     5.995000e+03   1.012500e+04 5.920988e-01
## [1] 0.001252337
## [1] 3
## [1] 4
## [1] "AKI"                    "RAISED.CARDIAC.ENZYMES"
##                   Cases People at risk         Risk
## Exposed   6.284000e+03   7.981000e+03 7.873700e-01
```

```
## Unexposed 1.531000e+03   2.144000e+03 7.140858e-01
## Total     7.815000e+03   1.012500e+04 7.718519e-01
## [1] 1.058775e-11
## [1] 3
## [1] 5
## [1] "AKI"     "ANAEMIA"
##                   Cases People at risk         Risk
## Exposed   7.049000e+03   7.981000e+03 8.832227e-01
## Unexposed 1.384000e+03   2.144000e+03 6.455224e-01
## Total     8.433000e+03   1.012500e+04 8.328889e-01
## [1] 0
## [1] 3
## [1] 6
## [1] "AKI"       "PRIOR.CMP"
##                   Cases People at risk         Risk
## Exposed   7.035000e+03   7.981000e+03 8.814685e-01
## Unexposed 1.509000e+03   2.144000e+03 7.038246e-01
## Total     8.544000e+03   1.012500e+04 8.438519e-01
## [1] 0
## [1] 3
## [1] 7
## [1] "AKI" "CAD"
##                   Cases People at risk         Risk
## Exposed   2.504000e+03   7.981000e+03 3.137451e-01
## Unexposed 7.780000e+02   2.144000e+03 3.628731e-01
## Total     3.282000e+03   1.012500e+04 3.241481e-01
## [1] 2.324325e-05
## [1] 3
## [1] 8
## [1] "AKI" "HTN"
##                   Cases People at risk         Risk
## Exposed   4.277000e+03   7.981000e+03 5.358978e-01
## Unexposed 9.730000e+02   2.144000e+03 4.538246e-01
## Total     5.250000e+03   1.012500e+04 5.185185e-01
## [1] 1.248268e-11
## [1] 3
## [1] 9
## [1] "AKI" "DM"
##                   Cases People at risk         Risk
## Exposed   5.622000e+03   7.981000e+03 7.044230e-01
## Unexposed 1.180000e+03   2.144000e+03 5.503731e-01
## Total     6.802000e+03   1.012500e+04 6.718025e-01
## [1] 0
## [1] 3
## [1] 10
## [1] "AKI"     "ALCOHOL"
```

```
##                  Cases People at risk         Risk
## Exposed   7.384000e+03   7.981000e+03 9.251973e-01
## Unexposed 2.006000e+03   2.144000e+03 9.356343e-01
## Total     9.390000e+03   1.012500e+04 9.274074e-01
## [1] 0.08517626
## [1] "The variables:"        "AKI"                   "ALCOHOL"
## [4] "are NOT independent !" "p-value = "            "0.085176258834
9421"
## [1] 4
## [1] 1
## [1] "ACS"   "STEMI"
##                  Cases People at risk         Risk
## Exposed   5.995000e+03   5.995000e+03 1.000000e+00
## Unexposed 2.446000e+03   4.130000e+03 5.922518e-01
## Total     8.441000e+03   1.012500e+04 8.336790e-01
## [1] 0
## [1] 4
## [1] 2
## [1] "ACS" "AKI"
##                  Cases People at risk         Risk
## Exposed   4.661000e+03   5.995000e+03 7.774812e-01
## Unexposed 3.320000e+03   4.130000e+03 8.038741e-01
## Total     7.981000e+03   1.012500e+04 7.882469e-01
## [1] 0.001265821
## [1] 4
## [1] 3
## [1] 4
## [1] 4
## [1] "ACS"                    "RAISED.CARDIAC.ENZYMES"
##                  Cases People at risk         Risk
## Exposed   5.477000e+03   5.995000e+03 9.135947e-01
## Unexposed 2.338000e+03   4.130000e+03 5.661017e-01
## Total     7.815000e+03   1.012500e+04 7.718519e-01
## [1] 0
## [1] 4
## [1] 5
## [1] "ACS"     "ANAEMIA"
##                  Cases People at risk         Risk
## Exposed   4.915000e+03   5.995000e+03 8.198499e-01
## Unexposed 3.518000e+03   4.130000e+03 8.518160e-01
## Total     8.433000e+03   1.012500e+04 8.328889e-01
## [1] 1.688755e-05
## [1] 4
## [1] 6
## [1] "ACS"       "PRIOR.CMP"
##                  Cases People at risk         Risk
```

```
## Exposed   4.957000e+03   5.995000e+03 8.268557e-01
## Unexposed 3.587000e+03   4.130000e+03 8.685230e-01
## Total     8.544000e+03   1.012500e+04 8.438519e-01
## [1] 6.455421e-09
## [1] 4
## [1] 7
## [1] "ACS" "CAD"
##                   Cases People at risk         Risk
## Exposed   2.473000e+03   5.995000e+03 4.125104e-01
## Unexposed 8.090000e+02   4.130000e+03 1.958838e-01
## Total     3.282000e+03   1.012500e+04 3.241481e-01
## [1] 0
## [1] 4
## [1] 8
## [1] "ACS" "HTN"
##                   Cases People at risk         Risk
## Exposed   3.086000e+03   5.995000e+03 5.147623e-01
## Unexposed 2.164000e+03   4.130000e+03 5.239709e-01
## Total     5.250000e+03   1.012500e+04 5.185185e-01
## [1] 0.3620164
## [1] "The variables:"        "ACS"                   "HTN"
## [4] "are NOT independent !" "p-value = "            "0.362016422687
757"
## [1] 4
## [1] 9
## [1] "ACS" "DM"
##                   Cases People at risk         Risk
## Exposed   4.050000e+03   5.995000e+03 6.755630e-01
## Unexposed 2.752000e+03   4.130000e+03 6.663438e-01
## Total     6.802000e+03   1.012500e+04 6.718025e-01
## [1] 0.3322117
## [1] "The variables:"        "ACS"                   "DM"
## [4] "are NOT independent !" "p-value = "            "0.332211719875
987"
## [1] 5
## [1] 1
## [1] "RAISED.CARDIAC.ENZYMES" "STEMI"
##                   Cases People at risk         Risk
## Exposed   6.616000e+03   7.815000e+03 8.465771e-01
## Unexposed 1.825000e+03   2.310000e+03 7.900433e-01
## Total     8.441000e+03   1.012500e+04 8.336790e-01
## [1] 1.833055e-09
## [1] 5
## [1] 2
## [1] "RAISED.CARDIAC.ENZYMES" "AKI"
##                   Cases People at risk         Risk
```

```
## Exposed   6.284000e+03   7.815000e+03 8.040947e-01
## Unexposed 1.697000e+03   2.310000e+03 7.346320e-01
## Total     7.981000e+03   1.012500e+04 7.882469e-01
## [1] 1.094924e-11
## [1] 5
## [1] 3
## [1] "RAISED.CARDIAC.ENZYMES" "ACS"
##                   Cases People at risk         Risk
## Exposed   5.477000e+03   7.815000e+03 7.008317e-01
## Unexposed 5.180000e+02   2.310000e+03 2.242424e-01
## Total     5.995000e+03   1.012500e+04 5.920988e-01
## [1] 0
## [1] 5
## [1] 4
## [1] 5
## [1] 5
## [1] "RAISED.CARDIAC.ENZYMES" "ANAEMIA"
##                   Cases People at risk         Risk
## Exposed   6.563000e+03   7.815000e+03 8.397953e-01
## Unexposed 1.870000e+03   2.310000e+03 8.095238e-01
## Total     8.433000e+03   1.012500e+04 8.328889e-01
## [1] 0.0009547112
## [1] 5
## [1] 6
## [1] "RAISED.CARDIAC.ENZYMES" "PRIOR.CMP"
##                   Cases People at risk         Risk
## Exposed   6.593000e+03   7.815000e+03 8.436340e-01
## Unexposed 1.951000e+03   2.310000e+03 8.445887e-01
## Total     8.544000e+03   1.012500e+04 8.438519e-01
## [1] 0.9114523
## [1] "The variables:"         "RAISED.CARDIAC.ENZYMES" "PRIOR.CMP"
## [4] "are NOT independent !"  "p-value = "             "0.9114523159
88431"
## [1] 5
## [1] 7
## [1] "RAISED.CARDIAC.ENZYMES" "CAD"
##                   Cases People at risk         Risk
## Exposed   2.686000e+03   7.815000e+03 3.436980e-01
## Unexposed 5.960000e+02   2.310000e+03 2.580087e-01
## Total     3.282000e+03   1.012500e+04 3.241481e-01
## [1] 4.440892e-16
## [1] 5
## [1] 8
## [1] "RAISED.CARDIAC.ENZYMES" "HTN"
##                   Cases People at risk         Risk
## Exposed   4.136000e+03   7.815000e+03 5.292386e-01
```

```
## Unexposed 1.114000e+03   2.310000e+03 4.822511e-01
## Total     5.250000e+03   1.012500e+04 5.185185e-01
## [1] 7.138793e-05
## [1] 6
## [1] 1
## [1] "ANAEMIA" "STEMI"
##                   Cases People at risk         Risk
## Exposed   6.931000e+03   8.433000e+03 8.218902e-01
## Unexposed 1.510000e+03   1.692000e+03 8.924350e-01
## Total     8.441000e+03   1.012500e+04 8.336790e-01
## [1] 2.220446e-16
## [1] 6
## [1] 2
## [1] "ANAEMIA" "AKI"
##                   Cases People at risk         Risk
## Exposed   7.049000e+03   8.433000e+03 8.358828e-01
## Unexposed 9.320000e+02   1.692000e+03 5.508274e-01
## Total     7.981000e+03   1.012500e+04 7.882469e-01
## [1] 0
## [1] 6
## [1] 3
## [1] "ANAEMIA" "ACS"
##                   Cases People at risk         Risk
## Exposed   4.915000e+03   8.433000e+03 5.828294e-01
## Unexposed 1.080000e+03   1.692000e+03 6.382979e-01
## Total     5.995000e+03   1.012500e+04 5.920988e-01
## [1] 1.599417e-05
## [1] 6
## [1] 4
## [1] "ANAEMIA"                "RAISED.CARDIAC.ENZYMES"
##                   Cases People at risk         Risk
## Exposed   6.563000e+03   8.433000e+03 7.782521e-01
## Unexposed 1.252000e+03   1.692000e+03 7.399527e-01
## Total     7.815000e+03   1.012500e+04 7.718519e-01
## [1] 0.0009456107
## [1] 6
## [1] 5
## [1] 6
## [1] 6
## [1] "ANAEMIA"   "PRIOR.CMP"
##                   Cases People at risk         Risk
## Exposed   7.166000e+03   8.433000e+03 8.497569e-01
## Unexposed 1.378000e+03   1.692000e+03 8.144208e-01
## Total     8.544000e+03   1.012500e+04 8.438519e-01
## [1] 0.0005456894
## [1] 6
```

```
## [1] 7
## [1] "ANAEMIA" "CAD"
##                  Cases People at risk         Risk
## Exposed   2.656000e+03   8.433000e+03 3.149532e-01
## Unexposed 6.260000e+02   1.692000e+03 3.699764e-01
## Total     3.282000e+03   1.012500e+04 3.241481e-01
## [1] 1.668604e-05
## [1] 7
## [1] 1
## [1] "PRIOR.CMP" "STEMI"
##                  Cases People at risk         Risk
## Exposed   7.091000e+03   8.544000e+03 8.299391e-01
## Unexposed 1.350000e+03   1.581000e+03 8.538899e-01
## Total     8.441000e+03   1.012500e+04 8.336790e-01
## [1] 0.01421728
## [1] 7
## [1] 2
## [1] "PRIOR.CMP" "AKI"
##                  Cases People at risk         Risk
## Exposed   7.035000e+03   8.544000e+03 8.233848e-01
## Unexposed 9.460000e+02   1.581000e+03 5.983555e-01
## Total     7.981000e+03   1.012500e+04 7.882469e-01
## [1] 0
## [1] 7
## [1] 3
## [1] "PRIOR.CMP" "ACS"
##                  Cases People at risk         Risk
## Exposed   4.957000e+03   8.544000e+03 5.801732e-01
## Unexposed 1.038000e+03   1.581000e+03 6.565465e-01
## Total     5.995000e+03   1.012500e+04 5.920988e-01
## [1] 5.279727e-09
## [1] 7
## [1] 4
## [1] "PRIOR.CMP"              "RAISED.CARDIAC.ENZYMES"
##                  Cases People at risk         Risk
## Exposed   6.593000e+03   8.544000e+03 7.716526e-01
## Unexposed 1.222000e+03   1.581000e+03 7.729285e-01
## Total     7.815000e+03   1.012500e+04 7.718519e-01
## [1] 0.9114522
## [1] "The variables:"         "PRIOR.CMP"              "RAISED.CARDI
AC.ENZYMES"
## [4] "are NOT independent !"  "p-value = "             "0.9114522130
97918"
## [1] 7
## [1] 5
## [1] "PRIOR.CMP" "ANAEMIA"
```

```
##                   Cases People at risk         Risk
## Exposed   7.166000e+03   8.544000e+03 8.387172e-01
## Unexposed 1.267000e+03   1.581000e+03 8.013915e-01
## Total     8.433000e+03   1.012500e+04 8.328889e-01
## [1] 0.0005440194
## [1] 7
## [1] 6
## [1] 8
## [1] 1
## [1] "CAD"   "STEMI"
##                   Cases People at risk         Risk
## Exposed   3.115000e+03   3.282000e+03 9.491164e-01
## Unexposed 5.326000e+03   6.843000e+03 7.783136e-01
## Total     8.441000e+03   1.012500e+04 8.336790e-01
## [1] 0
## [1] 8
## [1] 2
## [1] "CAD" "AKI"
##                   Cases People at risk         Risk
## Exposed   2.504000e+03   3.282000e+03 7.629494e-01
## Unexposed 5.477000e+03   6.843000e+03 8.003800e-01
## Total     7.981000e+03   1.012500e+04 7.882469e-01
## [1] 2.379996e-05
## [1] 8
## [1] 3
## [1] "CAD" "ACS"
##                   Cases People at risk         Risk
## Exposed   2.473000e+03   3.282000e+03 7.535040e-01
## Unexposed 3.522000e+03   6.843000e+03 5.146865e-01
## Total     5.995000e+03   1.012500e+04 5.920988e-01
## [1] 0
## [1] 8
## [1] 4
## [1] "CAD"                    "RAISED.CARDIAC.ENZYMES"
##                   Cases People at risk         Risk
## Exposed   2.686000e+03   3.282000e+03 8.184034e-01
## Unexposed 5.129000e+03   6.843000e+03 7.495251e-01
## Total     7.815000e+03   1.012500e+04 7.718519e-01
## [1] 6.661338e-16
## [1] 8
## [1] 5
## [1] "CAD"     "ANAEMIA"
##                   Cases People at risk         Risk
## Exposed   2.656000e+03   3.282000e+03 8.092626e-01
## Unexposed 5.777000e+03   6.843000e+03 8.442204e-01
## Total     8.433000e+03   1.012500e+04 8.328889e-01
```

```
## [1] 1.747886e-05
## [1] 9
## [1] 1
## [1] "HTN"   "STEMI"
##                  Cases People at risk         Risk
## Exposed   4.244000e+03   5.250000e+03 8.083810e-01
## Unexposed 4.197000e+03   4.875000e+03 8.609231e-01
## Total     8.441000e+03   1.012500e+04 8.336790e-01
## [1] 8.952838e-13
## [1] 9
## [1] 2
## [1] "HTN" "AKI"
##                  Cases People at risk         Risk
## Exposed   4.277000e+03   5.250000e+03 8.146667e-01
## Unexposed 3.704000e+03   4.875000e+03 7.597949e-01
## Total     7.981000e+03   1.012500e+04 7.882469e-01
## [1] 1.538414e-11
## [1] 9
## [1] 3
## [1] "HTN" "ACS"
##                  Cases People at risk         Risk
## Exposed   3.086000e+03   5.250000e+03 5.878095e-01
## Unexposed 2.909000e+03   4.875000e+03 5.967179e-01
## Total     5.995000e+03   1.012500e+04 5.920988e-01
## [1] 0.3620191
## [1] "The variables:"        "HTN"                   "ACS"
## [4] "are NOT independent !" "p-value = "            "0.362019081521
924"
## [1] 9
## [1] 4
## [1] "HTN"                    "RAISED.CARDIAC.ENZYMES"
##                  Cases People at risk         Risk
## Exposed   4.136000e+03   5.250000e+03 7.878095e-01
## Unexposed 3.679000e+03   4.875000e+03 7.546667e-01
## Total     7.815000e+03   1.012500e+04 7.718519e-01
## [1] 7.295238e-05
## [1] 10
## [1] 1
## [1] "DM"    "STEMI"
##                  Cases People at risk         Risk
## Exposed   5.656000e+03   6.802000e+03 8.315201e-01
## Unexposed 2.785000e+03   3.323000e+03 8.380981e-01
## Total     8.441000e+03   1.012500e+04 8.336790e-01
## [1] 0.4013166
## [1] "The variables:"        "DM"                    "STEMI"
## [4] "are NOT independent !" "p-value = "            "0.401316645694
```

```
662"
## [1] 10
## [1] 2
## [1] "DM"  "AKI"
##                  Cases People at risk         Risk
## Exposed   5.622000e+03   6.802000e+03 8.265216e-01
## Unexposed 2.359000e+03   3.323000e+03 7.099007e-01
## Total     7.981000e+03   1.012500e+04 7.882469e-01
## [1] 0
## [1] 10
## [1] 3
## [1] "DM"  "ACS"
##                  Cases People at risk         Risk
## Exposed   4.050000e+03   6.802000e+03 5.954131e-01
## Unexposed 1.945000e+03   3.323000e+03 5.853145e-01
## Total     5.995000e+03   1.012500e+04 5.920988e-01
## [1] 0.3322029
## [1] "The variables:"        "DM"                    "ACS"
## [4] "are NOT independent !" "p-value = "            "0.332202852139
091"
## [1] 11
## [1] 1
## [1] "ALCOHOL" "STEMI"
##                  Cases People at risk         Risk
## Exposed   7.876000e+03   9.390000e+03 8.387646e-01
## Unexposed 5.650000e+02   7.350000e+02 7.687075e-01
## Total     8.441000e+03   1.012500e+04 8.336790e-01
## [1] 1.208819e-05
## [1] 11
## [1] 2
## [1] "ALCOHOL" "AKI"
##                  Cases People at risk         Risk
## Exposed   7.384000e+03   9.390000e+03 7.863685e-01
## Unexposed 5.970000e+02   7.350000e+02 8.122449e-01
## Total     7.981000e+03   1.012500e+04 7.882469e-01
## [1] 0.08477142
## [1] "The variables:"        "ALCOHOL"               "AKI"
## [4] "are NOT independent !" "p-value = "            "0.084771422089
4678"
## [1] 12
## [1] 1
## [1] "SMOKING" "STEMI"
##                  Cases People at risk         Risk
## Exposed   8.034000e+03   9.571000e+03 8.394107e-01
## Unexposed 4.070000e+02   5.540000e+02 7.346570e-01
## Total     8.441000e+03   1.012500e+04 8.336790e-01
```

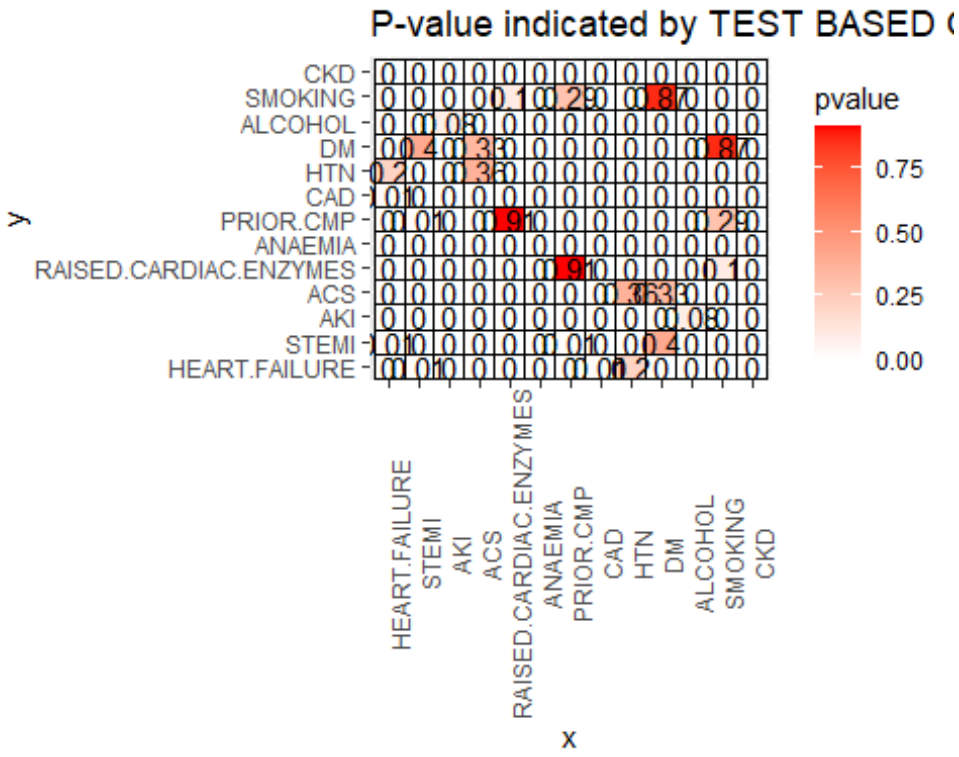
```
## [1] 4.352847e-08
## [1] 12
## [1] 2
## [1] "SMOKING" "AKI"
##                   Cases People at risk         Risk
## Exposed   7.504000e+03   9.571000e+03 7.840351e-01
## Unexposed 4.770000e+02   5.540000e+02 8.610108e-01
## Total     7.981000e+03   1.012500e+04 7.882469e-01
## [1] 4.772438e-07
## [1] 12
## [1] 3
## [1] "SMOKING" "ACS"
##                   Cases People at risk         Risk
## Exposed   5.713000e+03   9.571000e+03 5.969073e-01
## Unexposed 2.820000e+02   5.540000e+02 5.090253e-01
## Total     5.995000e+03   1.012500e+04 5.920988e-01
## [1] 5.649662e-05
## [1] 12
## [1] 4
## [1] "SMOKING"                "RAISED.CARDIAC.ENZYMES"
##                   Cases People at risk         Risk
## Exposed   7.404000e+03   9.571000e+03 7.735869e-01
## Unexposed 4.110000e+02   5.540000e+02 7.418773e-01
## Total     7.815000e+03   1.012500e+04 7.718519e-01
## [1] 0.09648714
## [1] "The variables:"         "SMOKING"                "RAISED.CARDI
AC.ENZYMES"
## [4] "are NOT independent !"  "p-value = "             "0.0964871415
049153"
## [1] 12
## [1] 5
## [1] "SMOKING" "ANAEMIA"
##                   Cases People at risk         Risk
## Exposed   7.942000e+03   9.571000e+03 8.297983e-01
## Unexposed 4.910000e+02   5.540000e+02 8.862816e-01
## Total     8.433000e+03   1.012500e+04 8.328889e-01
## [1] 5.636799e-05
## [1] 12
## [1] 6
## [1] "SMOKING"   "PRIOR.CMP"
##                   Cases People at risk         Risk
## Exposed   8.068000e+03   9.571000e+03 8.429631e-01
## Unexposed 4.760000e+02   5.540000e+02 8.592058e-01
## Total     8.544000e+03   1.012500e+04 8.438519e-01
## [1] 0.2864485
## [1] "The variables:"        "SMOKING"               "PRIOR.CMP"
```

```
## [4] "are NOT independent !" "p-value = "            "0.286448470985
319"
## [1] 12
## [1] 7
## [1] "SMOKING" "CAD"
##                  Cases People at risk         Risk
## Exposed   3.137000e+03   9.571000e+03 3.277609e-01
## Unexposed 1.450000e+02   5.540000e+02 2.617329e-01
## Total     3.282000e+03   1.012500e+04 3.241481e-01
## [1] 0.0006164428
## [1] 12
## [1] 8
## [1] "SMOKING" "HTN"
##                  Cases People at risk         Risk
## Exposed   4.894000e+03   9.571000e+03 5.113363e-01
## Unexposed 3.560000e+02   5.540000e+02 6.425993e-01
## Total     5.250000e+03   1.012500e+04 5.185185e-01
## [1] 4.026754e-10
## [1] 12
## [1] 9
## [1] "SMOKING" "DM"
##                  Cases People at risk         Risk
## Exposed   6.428000e+03   9.571000e+03 6.716122e-01
## Unexposed 3.740000e+02   5.540000e+02 6.750903e-01
## Total     6.802000e+03   1.012500e+04 6.718025e-01
## [1] 0.8650716
## [1] "The variables:"        "SMOKING"              "DM"
## [4] "are NOT independent !" "p-value = "           "0.865071555648
884"
## [1] 12
## [1] 10
## [1] "SMOKING" "ALCOHOL"
##                  Cases People at risk         Risk
## Exposed   9.060000e+03   9.571000e+03 9.466095e-01
## Unexposed 3.300000e+02   5.540000e+02 5.956679e-01
## Total     9.390000e+03   1.012500e+04 9.274074e-01
## [1] 0
## [1] 12
## [1] 11
## [1] 12
## [1] 12
## [1] "SMOKING" "CKD"
##                  Cases People at risk         Risk
## Exposed   8.746000e+03   9.571000e+03 9.138021e-01
## Unexposed 5.280000e+02   5.540000e+02 9.530686e-01
```

```
## Total     9.274000e+03   1.012500e+04 9.159506e-01
## [1] 3.141058e-05

# Adding names to the rows and columns of the matrix
row.names(pvalueMatrix) <- categoricalNames
colnames(pvalueMatrix) <- categoricalNames
pvalueMatrix

##                         HEART.FAILURE        STEMI          AKI
ACS
## HEART.FAILURE           0.000000e+00 1.008817e-02 0.000000e+00 1.13
5319e-04
## STEMI                   1.008817e-02 0.000000e+00 3.275835e-08 0.00
0000e+00
## AKI                     0.000000e+00 3.275835e-08 0.000000e+00 1.26
5821e-03
## ACS                     1.135319e-04 0.000000e+00 1.265821e-03 0.00
0000e+00
## RAISED.CARDIAC.ENZYMES  0.000000e+00 1.833055e-09 1.094924e-11 0.00
0000e+00
## ANAEMIA                 0.000000e+00 2.220446e-16 0.000000e+00 1.59
9417e-05
## PRIOR.CMP               0.000000e+00 1.421728e-02 0.000000e+00 5.27
9727e-09
## CAD                     1.297268e-02 0.000000e+00 2.379996e-05 0.00
0000e+00
## HTN                     2.048202e-01 8.952838e-13 1.538414e-11 3.62
0191e-01
## DM                      1.287407e-08 4.013166e-01 0.000000e+00 3.32
2029e-01
## ALCOHOL                 2.572918e-07 1.208819e-05 8.477142e-02 0.00
0000e+00
## SMOKING                 7.399081e-05 4.352847e-08 4.772438e-07 5.64
9662e-05
## CKD                     0.000000e+00 0.000000e+00 0.000000e+00 0.00
0000e+00
##                         RAISED.CARDIAC.ENZYMES      ANAEMIA    PRIOR
.CMP
## HEART.FAILURE                     0.000000e+00 0.000000e+00 0.000000
e+00
## STEMI                             1.833055e-09 2.220446e-16 1.421728
e-02
## AKI                               1.094924e-11 0.000000e+00 0.000000
e+00
## ACS                               0.000000e+00 1.599417e-05 5.279727
e-09
```

```
## RAISED.CARDIAC.ENZYMES           0.000000e+00 9.456107e-04 9.114522
e-01
## ANAEMIA                          9.456107e-04 0.000000e+00 5.440194
e-04
## PRIOR.CMP                        9.114522e-01 5.440194e-04 0.000000
e+00
## CAD                              6.661338e-16 1.747886e-05 0.000000
e+00
## HTN                              7.295238e-05 0.000000e+00 0.000000
e+00
## DM                               0.000000e+00 0.000000e+00 0.000000
e+00
## ALCOHOL                          0.000000e+00 0.000000e+00 0.000000
e+00
## SMOKING                          9.648714e-02 5.636799e-05 2.864485
e-01
## CKD                              0.000000e+00 0.000000e+00 0.000000
e+00
##                                       CAD          HTN           DM
ALCOHOL
## HEART.FAILURE          1.297268e-02 2.048202e-01 1.287407e-08 2.572
918e-07
## STEMI                  0.000000e+00 8.952838e-13 4.013166e-01 1.208
819e-05
## AKI                    2.379996e-05 1.538414e-11 0.000000e+00 8.477
142e-02
## ACS                    0.000000e+00 3.620191e-01 3.322029e-01 0.000
000e+00
## RAISED.CARDIAC.ENZYMES 6.661338e-16 7.295238e-05 0.000000e+00 0.000
000e+00
## ANAEMIA                1.747886e-05 0.000000e+00 0.000000e+00 0.000
000e+00
## PRIOR.CMP              0.000000e+00 0.000000e+00 0.000000e+00 0.000
000e+00
## CAD                    0.000000e+00 0.000000e+00 0.000000e+00 0.000
000e+00
## HTN                    0.000000e+00 0.000000e+00 0.000000e+00 0.000
000e+00
## DM                     0.000000e+00 0.000000e+00 0.000000e+00 0.000
000e+00
## ALCOHOL                0.000000e+00 0.000000e+00 0.000000e+00 0.000
000e+00
## SMOKING                6.164428e-04 4.026754e-10 8.650716e-01 0.000
000e+00
## CKD                    0.000000e+00 0.000000e+00 0.000000e+00 0.000
000e+00
```

```
##                               SMOKING          CKD
## HEART.FAILURE           7.399081e-05 0.000000e+00
## STEMI                   4.352847e-08 0.000000e+00
## AKI                     4.772438e-07 0.000000e+00
## ACS                     5.649662e-05 0.000000e+00
## RAISED.CARDIAC.ENZYMES 9.648714e-02 0.000000e+00
## ANAEMIA                 5.636799e-05 0.000000e+00
## PRIOR.CMP               2.864485e-01 0.000000e+00
## CAD                     6.164428e-04 0.000000e+00
## HTN                     4.026754e-10 0.000000e+00
## DM                      8.650716e-01 0.000000e+00
## ALCOHOL                 0.000000e+00 0.000000e+00
## SMOKING                 0.000000e+00 3.141058e-05
## CKD                     3.141058e-05 0.000000e+00

#Plotting Heatmap with p-values
library(reshape)
library(ggplot2)

#Transfor Matrix
df<-melt(pvalueMatrix)

colnames(df)<-c("x","y", "pvalue")
#head(df)

#Plotting heatmap
ggplot(df, aes(x = x, y = y, fill = pvalue))+
  geom_tile(color = "black") +
  scale_fill_gradient(low = "white", high = "red") +
  geom_text(aes(label=round(pvalue,2)), color = "black", size = 4) +
  theme(axis.text.x=element_text(angle=90))+
  ggtitle("P-value indicated by TEST BASED ON THE DIFFERENCE")
```




```
  coord_fixed()
```

```
## <ggproto object: Class CoordFixed, CoordCartesian, Coord, gg>
##     aspect: function
##     backtransform_range: function
##     clip: on
##     default: FALSE
##     distance: function
##     expand: TRUE
##     is_free: function
##     is_linear: function
##     labels: function
##     limits: list
##     modify_scales: function
##     range: function
##     ratio: 1
##     render_axis_h: function
##     render_axis_v: function
##     render_bg: function
##     render_fg: function
##     setup_data: function
##     setup_layout: function
##     setup_panel_guides: function
##     setup_panel_params: function
##     setup_params: function
```

```
##     train_panel_guides: function
##     transform: function
##     super:  <ggproto object: Class CoordFixed, CoordCartesian, Coor
d, gg>

#Saving the heatmap in a file (better resolution)
ggsave("heatmap.png", width = 16, height = 10)

# Loading the heatmap png file for better visualization quality

knitr::kable(df, caption='CONTIGENCE TABLE', align="lrcrccccc")
```

CONTIGENCE TABLE

x	y	pvalue
HEART.FAILURE	HEART.FAILURE	0.0000000
STEMI	HEART.FAILURE	0.0100882
AKI	HEART.FAILURE	0.0000000
ACS	HEART.FAILURE	0.0001135
RAISED.CARDIAC.ENZYMES	HEART.FAILURE	0.0000000
ANAEMIA	HEART.FAILURE	0.0000000
PRIOR.CMP	HEART.FAILURE	0.0000000
CAD	HEART.FAILURE	0.0129727
HTN	HEART.FAILURE	0.2048202
DM	HEART.FAILURE	0.0000000
ALCOHOL	HEART.FAILURE	0.0000003
SMOKING	HEART.FAILURE	0.0000740
CKD	HEART.FAILURE	0.0000000
HEART.FAILURE	STEMI	0.0100882
STEMI	STEMI	0.0000000
AKI	STEMI	0.0000000
ACS	STEMI	0.0000000
RAISED.CARDIAC.ENZYMES	STEMI	0.0000000
ANAEMIA	STEMI	0.0000000
PRIOR.CMP	STEMI	0.0142173
CAD	STEMI	0.0000000
HTN	STEMI	0.0000000
DM	STEMI	0.4013166
ALCOHOL	STEMI	0.0000121

SMOKING	STEMI	0.0000000
CKD	STEMI	0.0000000
HEART.FAILURE	AKI	0.0000000
STEMI	AKI	0.0000000
AKI	AKI	0.0000000
ACS	AKI	0.0012658
RAISED.CARDIAC.ENZYMES	AKI	0.0000000
ANAEMIA	AKI	0.0000000
PRIOR.CMP	AKI	0.0000000
CAD	AKI	0.0000238
HTN	AKI	0.0000000
DM	AKI	0.0000000
ALCOHOL	AKI	0.0847714
SMOKING	AKI	0.0000005
CKD	AKI	0.0000000
HEART.FAILURE	ACS	0.0001135
STEMI	ACS	0.0000000
AKI	ACS	0.0012658
ACS	ACS	0.0000000
RAISED.CARDIAC.ENZYMES	ACS	0.0000000
ANAEMIA	ACS	0.0000160
PRIOR.CMP	ACS	0.0000000
CAD	ACS	0.0000000
HTN	ACS	0.3620191
DM	ACS	0.3322029
ALCOHOL	ACS	0.0000000
SMOKING	ACS	0.0000565
CKD	ACS	0.0000000
HEART.FAILURE	RAISED.CARDIAC.ENZYMES	0.0000000
STEMI	RAISED.CARDIAC.ENZYMES	0.0000000
AKI	RAISED.CARDIAC.ENZYMES	0.0000000
ACS	RAISED.CARDIAC.ENZYMES	0.0000000
RAISED.CARDIAC.ENZYMES	RAISED.CARDIAC.ENZYMES	0.0000000
ANAEMIA	RAISED.CARDIAC.ENZYMES	0.0009456
PRIOR.CMP	RAISED.CARDIAC.ENZYMES	0.9114522

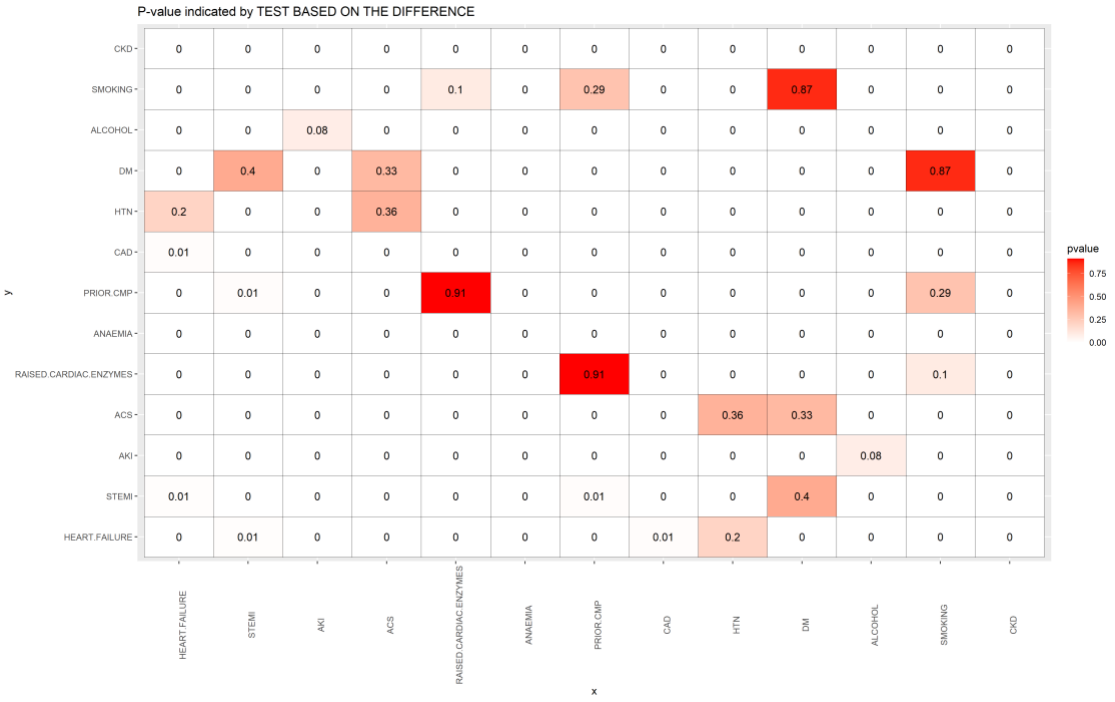
CAD	RAISED.CARDIAC.ENZYMES	0.0000000
HTN	RAISED.CARDIAC.ENZYMES	0.0000730
DM	RAISED.CARDIAC.ENZYMES	0.0000000
ALCOHOL	RAISED.CARDIAC.ENZYMES	0.0000000
SMOKING	RAISED.CARDIAC.ENZYMES	0.0964871
CKD	RAISED.CARDIAC.ENZYMES	0.0000000
HEART.FAILURE	ANAEMIA	0.0000000
STEMI	ANAEMIA	0.0000000
AKI	ANAEMIA	0.0000000
ACS	ANAEMIA	0.0000160
RAISED.CARDIAC.ENZYMES	ANAEMIA	0.0009456
ANAEMIA	ANAEMIA	0.0000000
PRIOR.CMP	ANAEMIA	0.0005440
CAD	ANAEMIA	0.0000175
HTN	ANAEMIA	0.0000000
DM	ANAEMIA	0.0000000
ALCOHOL	ANAEMIA	0.0000000
SMOKING	ANAEMIA	0.0000564
CKD	ANAEMIA	0.0000000
HEART.FAILURE	PRIOR.CMP	0.0000000
STEMI	PRIOR.CMP	0.0142173
AKI	PRIOR.CMP	0.0000000
ACS	PRIOR.CMP	0.0000000
RAISED.CARDIAC.ENZYMES	PRIOR.CMP	0.9114522
ANAEMIA	PRIOR.CMP	0.0005440
PRIOR.CMP	PRIOR.CMP	0.0000000
CAD	PRIOR.CMP	0.0000000
HTN	PRIOR.CMP	0.0000000
DM	PRIOR.CMP	0.0000000
ALCOHOL	PRIOR.CMP	0.0000000
SMOKING	PRIOR.CMP	0.2864485
CKD	PRIOR.CMP	0.0000000
HEART.FAILURE	CAD	0.0129727
STEMI	CAD	0.0000000
AKI	CAD	0.0000238

ACS	CAD	0.0000000
RAISED.CARDIAC.ENZYMES	CAD	0.0000000
ANAEMIA	CAD	0.0000175
PRIOR.CMP	CAD	0.0000000
CAD	CAD	0.0000000
HTN	CAD	0.0000000
DM	CAD	0.0000000
ALCOHOL	CAD	0.0000000
SMOKING	CAD	0.0006164
CKD	CAD	0.0000000
HEART.FAILURE	HTN	0.2048202
STEMI	HTN	0.0000000
AKI	HTN	0.0000000
ACS	HTN	0.3620191
RAISED.CARDIAC.ENZYMES	HTN	0.0000730
ANAEMIA	HTN	0.0000000
PRIOR.CMP	HTN	0.0000000
CAD	HTN	0.0000000
HTN	HTN	0.0000000
DM	HTN	0.0000000
ALCOHOL	HTN	0.0000000
SMOKING	HTN	0.0000000
CKD	HTN	0.0000000
HEART.FAILURE	DM	0.0000000
STEMI	DM	0.4013166
AKI	DM	0.0000000
ACS	DM	0.3322029
RAISED.CARDIAC.ENZYMES	DM	0.0000000
ANAEMIA	DM	0.0000000
PRIOR.CMP	DM	0.0000000
CAD	DM	0.0000000
HTN	DM	0.0000000
DM	DM	0.0000000
ALCOHOL	DM	0.0000000
SMOKING	DM	0.8650716

CKD	DM	0.0000000
HEART.FAILURE	ALCOHOL	0.0000003
STEMI	ALCOHOL	0.0000121
AKI	ALCOHOL	0.0847714
ACS	ALCOHOL	0.0000000
RAISED.CARDIAC.ENZYMES	ALCOHOL	0.0000000
ANAEMIA	ALCOHOL	0.0000000
PRIOR.CMP	ALCOHOL	0.0000000
CAD	ALCOHOL	0.0000000
HTN	ALCOHOL	0.0000000
DM	ALCOHOL	0.0000000
ALCOHOL	ALCOHOL	0.0000000
SMOKING	ALCOHOL	0.0000000
CKD	ALCOHOL	0.0000000
HEART.FAILURE	SMOKING	0.0000740
STEMI	SMOKING	0.0000000
AKI	SMOKING	0.0000005
ACS	SMOKING	0.0000565
RAISED.CARDIAC.ENZYMES	SMOKING	0.0964871
ANAEMIA	SMOKING	0.0000564
PRIOR.CMP	SMOKING	0.2864485
CAD	SMOKING	0.0006164
HTN	SMOKING	0.0000000
DM	SMOKING	0.8650716
ALCOHOL	SMOKING	0.0000000
SMOKING	SMOKING	0.0000000
CKD	SMOKING	0.0000314
HEART.FAILURE	CKD	0.0000000
STEMI	CKD	0.0000000
AKI	CKD	0.0000000
ACS	CKD	0.0000000
RAISED.CARDIAC.ENZYMES	CKD	0.0000000
ANAEMIA	CKD	0.0000000
PRIOR.CMP	CKD	0.0000000
CAD	CKD	0.0000000

HTN	CKD	0.0000000
DM	CKD	0.0000000
ALCOHOL	CKD	0.0000000
SMOKING	CKD	0.0000314
CKD	CKD	0.0000000

```
knitr::include_graphics("heatmap.png")
```



Based on the heatmap:

1) For HEART.FAILURE, only HTN indicated that there is NOT association (p-value = 0.2 > 0.05) at 5$ level, the other ones are dependent and could take into account into the models. However, the other ones should be independent among them. So, we can collect from the Heat Map: 1.1) STEMI - Only DM is independent 1.2) AKI - only ALCOHOL is independent 1.3) ACS - only HTN and DM are independent 1.4) RAISED CARDIAC ENZIMES - only PRIOR.CMP and SMOKING are independent 1.5) ANAEMIA and CKD - all of them are dependent 1.6) ALCOHOL - only AKI is independent 1.7) SMOKING - only RAISED>CARDIAX.ENZIMES, PRIOR.CMP and DM are independent

Based on this, there are different possibilities of combination of categorical variables for HEART.FAILURE modelling: a. HEART.FAILURE in function of RAISED.CARDIAC.ENZYMES, PRIOR.CMP and SMOKING b. HEART.FAILURE in function of RAISED.CARDIAC.ENZYMES and PRIOR.CMP c. HEART.FAILURE in function of RAISED.CARDIAC.ENZYMES and SMOKING d. HEART.FAILURE in function of PRIOR.CMP and SMOKING e. HEART.FAILURE in function of STEMI and DM f. HEART.FAILURE in function of AKI and ALCOHOL g. HEART.FAILURE in function of ACS and DM

2) For AKI, only ALCOHOL indicated that there is NOT association (p-value = 0.08 > 0.05) at 5$ level, the other ones are dependent and could take into account into the models. However, the other ones should be independent among them. For this reason and based on Heatmap, the possibilities of models are as following: 2.1) HEART.FAILURE - Only HTN is independent 2.2) STEMI - Only DM is independent 2.3) ACS - Only HTN is independent 2.4) RAISED CARDIAC ENZIMES - only PRIOR.CMP and ALCOHOL are independent 2.5) ANAEMIA and CKD - all of them are dependent

Based on this, there are different possibilities for AKI modelling: a. AKI in function of STEMI and DM b. AKI in function of ACS and DM c. AKI in function of ACS and HTN d. AKI in function of RAISED.CARDIAC.ENZYMES, PRIOR.CMP and SMOKING e. AKI in function of RAISED.CARDIAC.ENZYME and, PRIOR.CMP f. AKI in function of RAISED.CARDIAC.ENZYMES and SMOKING g. AKI in function of PRIOR.CMP and SMOKING h. AKI in function of HEART.FAILURE and HTN

In summary, we could find by the test of independence by applying test based on the differences using 2x2 Table Contingency the dependent explanatory variables of our response variables, at the same time these explanatory variables being independent among them.

PART IV - MODELLING AND PREDICTION

IV.1 - Categorical Response Variables (HEART.FAILURE and AKI)

IV.1.1 - HEART.FAILURE

For HEart Failure, it was defined to use all the quantitative variables and the categorical variables and Gender (based on EDA) and RAISED.CARDIAC ENZYMES, PRIOR.CMP and SMOKING (one of the possibility of models obtained with independence analysis with Contingency Table)

#(A) Evaluating HEART.FAILURE IN FUNCTION OF OTHER VARIABLES (only Quantitatives + GENDER + RAISED.CARDIAC ENZYMES + PRIOR.CMP + SMOKING)

Sampling (75% train part, 25% test part)

```
# Set a seed for random number generation
set.seed(10)

#Considering stratified sampling to separate each
library(sampling)

#Checking the order and total of each Gender
unique(dataClean$HEART.FAILURE)

## [1] 1 0
## Levels: 0 1
```

```
table(dataClean$HEART.FAILUREs) # >>>>>>>>>>>>>>need to check the s

## < table of extent 0 >

# 75% with HEAT.FAILURE = 1 and 0
n_HEART_Y = round(0.75*nrow(dataClean[dataClean$HEART.FAILURE == "1",]
))
n_HEART_N = round(0.75*nrow(dataClean[dataClean$HEART.FAILURE == "0",]
))


# srswor = Simple Random Sampling without replacement for each strata:
7% of Total for training
idx<-sampling::strata(dataClean, stratanames = ("HEART.FAILURE"), size
=c(n_HEART_Y,n_HEART_N), method="srswor")

#idx_unit

train=dataClean[idx$ID_unit,]

# Checking if the stratified sample was done correctly in Train part
table(train$HEART.FAILURE)

##
##    0    1
## 5413 2181

# Creating the test part (25% of Total, other rows not selected from t
raininng part)
test=dataClean[-idx$ID_unit,]

table(test$HEART.FAILURE)

##
##    0    1
## 1804  727

contrasts(train$HEART.FAILURE)

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

The stratified sampling separated the total population as expected.

#(A.1) LOGISTIC REGRESSION RAISED CARDIAC ENZIMES - only PRIOR.CMP and ALCOHOL

```
library(ISLR)
```

```
#Creating the the logistic regression model based on train part
HEART_logistic<-glm(HEART.FAILURE~factor(GENDER)+AGE+GLUCOSE+HB+TLC+PL
ATELETS+UREA+CREATININE+EF+factor(RAISED.CARDIAC.ENZYMES)+factor(PRIOR
.CMP)+
                      factor(SMOKING),family=binomial, data=train)

summary(HEART_logistic)

##
## Call:
## glm(formula = HEART.FAILURE ~ factor(GENDER) + AGE + GLUCOSE +
##     HB + TLC + PLATELETS + UREA + CREATININE + EF + factor(RAISED.C
ARDIAC.ENZYMES) +
##     factor(PRIOR.CMP) + factor(SMOKING), family = binomial, data =
train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.6565  -0.7346  -0.4786   0.8514   2.7395
##
## Coefficients:
##                                   Estimate Std. Error z value Pr(>|
z|)
## (Intercept)                        0.9412779  0.3076927   3.059 0.002
220 **
## factor(GENDER)M                   -0.3755875  0.0634067  -5.923 3.15e
-09 ***
## AGE                                0.0151023  0.0023756   6.357 2.05e
-10 ***
## GLUCOSE                            0.0012053  0.0003197   3.770 0.000
163 ***
## HB                                -0.0783684  0.0140360  -5.583 2.36e
-08 ***
## TLC                                0.0154811  0.0044585   3.472 0.000
516 ***
## PLATELETS                         -0.0003877  0.0002878  -1.347 0.178
044
## UREA                               0.0066479  0.0011014   6.036 1.58e
-09 ***
## CREATININE                        -0.0982801  0.0350772  -2.802 0.005
081 **
## EF                                -0.0554185  0.0029041 -19.083  < 2e
-16 ***
## factor(RAISED.CARDIAC.ENZYMES)1    0.4327435  0.0652899   6.628 3.40e
-11 ***
## factor(PRIOR.CMP)1                 0.5820183  0.0871285   6.680 2.39e
```

```
-11 ***
## factor(SMOKING)1               -0.3204094  0.1421573  -2.254 0.024
202 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 9107.1  on 7593  degrees of freedom
## Residual deviance: 7496.5  on 7581  degrees of freedom
## AIC: 7522.5
##
## Number of Fisher Scoring iterations: 4
```

Except PLATELETS (p-value > 0.05), all the other variables are significant at 5% level.

Checking the coefficient to see if there is some 0 at 95% Confidence Interval.

```
# Calculating the confidence interval on the coefficients
confint(HEART_logistic, level = 0.95)
```

```
##                                       2.5 %        97.5 %
## (Intercept)                      0.3380351900  1.5443919831
## factor(GENDER)M                 -0.4999546664 -0.2513675853
## AGE                              0.0104621944  0.0197758048
## GLUCOSE                          0.0005780096  0.0018316580
## HB                              -0.1058905190 -0.0508597872
## TLC                              0.0069267176  0.0243862091
## PLATELETS                       -0.0009542849  0.0001744202
## UREA                             0.0045023370  0.0088224100
## CREATININE                      -0.1679815306 -0.0302947647
## EF                              -0.0611389269 -0.0497527336
## factor(RAISED.CARDIAC.ENZYMES)1  0.3046014661  0.5605689273
## factor(PRIOR.CMP)1               0.4115782911  0.7531665960
## factor(SMOKING)1                -0.6042808020 -0.0464456706
```

Only PLATELETS with ZERO between UPPER and LOWER bounds of 95% Confidence Interval. Smoking indicated negative coefficient that indicates that this helps to reduce HEART.FAILURE that is not what is expected. So they should be removed.

Redoing Logistic Regression without PLATELETS and SMOKING.

```
# Calculating the confidence interval on the coefficients
confint(HEART_logistic, level = 0.95)
```

```
##                                       2.5 %        97.5 %
## (Intercept)                      0.3380351900  1.5443919831
## factor(GENDER)M                 -0.4999546664 -0.2513675853
```

```
## AGE                                 0.0104621944  0.0197758048
## GLUCOSE                             0.0005780096  0.0018316580
## HB                                 -0.1058905190 -0.0508597872
## TLC                                 0.0069267176  0.0243862091
## PLATELETS                          -0.0009542849  0.0001744202
## UREA                                0.0045023370  0.0088224100
## CREATININE                         -0.1679815306 -0.0302947647
## EF                                 -0.0611389269 -0.0497527336
## factor(RAISED.CARDIAC.ENZYMES)1     0.3046014661  0.5605689273
## factor(PRIOR.CMP)1                  0.4115782911  0.7531665960
## factor(SMOKING)1                   -0.6042808020 -0.0464456706
```

Now, all the variables are significant at 5% level and without 0 in any coefficient at 95% Confidence Interval.

Based on the coefficients, we can conclude that the probability of HEART.FAILURE: - decreases if the patient gender is male - decreases with HB, CREATININE and EF - increases with AGE, GLUCOSE, TLC and UREA; - increases if the patient has RAISED.CARDIAC.ENZYMES and PRIOR.CMP

Checking if there is problem of Multicolinearity in the training set.

```
#Checking Multicollinearity
library(car)

vif(HEART_logistic)
```

```
##                  factor(GENDER)                            AGE
##                        1.171097                       1.065967
##                         GLUCOSE                             HB
##                        1.048712                       1.271743
##                             TLC                      PLATELETS
##                        1.080857                       1.071593
##                            UREA                     CREATININE
##                        2.380428                       2.288321
##                              EF factor(RAISED.CARDIAC.ENZYMES)
##                        1.595727                       1.047283
##               factor(PRIOR.CMP)                 factor(SMOKING)
##                        1.559485                       1.037736
```

It was detected moderate collinearity between CREATININE and UREA, but it was not severe (VIF < 5, based on DATA 603), for this reason we adopt to keep them.

Applying Test part to the fitted logistic regression model

```
#Based on the model fitted based on the training data, predict the HEA
RT.FAILURE (Y) based on test data
Prob.predict_logistic<-predict(HEART_logistic,test,type="response")
```

```
testSize = nrow(test)
testSize
```

```
## [1] 2531
```

```
HEART_FAILURE.predict=rep("0", testSize)

HEART_FAILURE.predict[Prob.predict_logistic >= 0.5]="1"

# Checking the HEART_FAILURE prediction of Y=1 and N = 0 in the test d
ata
table(HEART_FAILURE.predict)
```

```
## HEART_FAILURE.predict
##    0    1
## 2077  454
```

```
#Comparing what the tests responses with the actual
actual=test$HEART.FAILURE
tablePred=table(HEART_FAILURE.predict,actual)

tablePred
```

```
##                       actual
## HEART_FAILURE.predict    0    1
##                     0 1637  440
##                     1  167  287
```

1636 “N” were predicted correctly, whereas 437 were wrong. In terms of “Y”, 290 were predicted correctly, whereas 168 were wrong. This considering p=0.5.

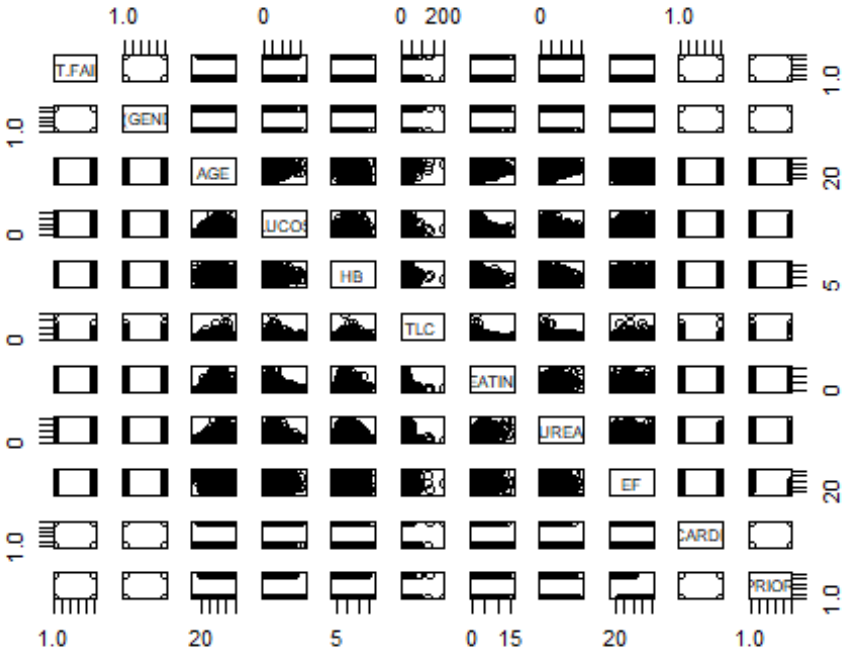
```
# Misclassification Ratio
mis_ratio = (tablePred[1, 2]+tablePred[2, 1])/(nrow(test))
mis_ratio
```

```
## [1] 0.2398262
```

The misclassification ratio is 0.230936 (23.1%), in other words, 76.9% were predicted correctly in the test part. This was obtained considering p=0.5.

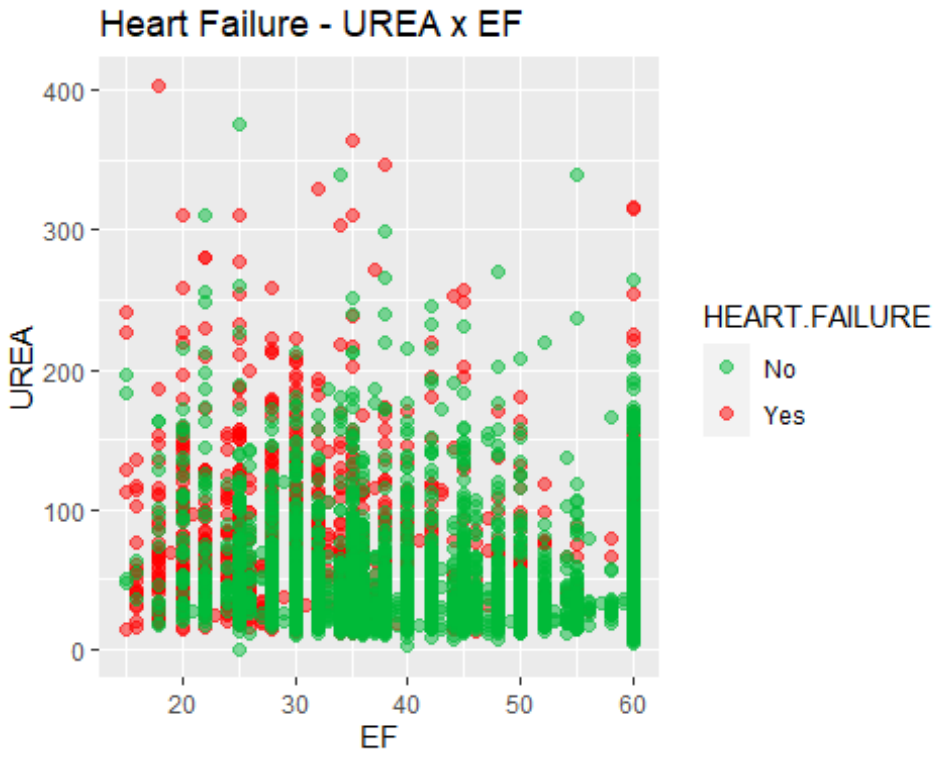
Checking the pairs.

```
pairs(~HEART.FAILURE+factor(GENDER)+AGE+GLUCOSE+HB+TLC+CREATININE+UREA
+EF+factor(RAISED.CARDIAC.ENZYMES)+factor(PRIOR.CMP), data=train)
```



Plotting only the most important variables (based on Classification Tree)

```
#Generating plots with regions
ggplot(data=train, aes(x=EF, y=UREA, color=HEART.FAILURE))+
  geom_point(size=2, alpha=0.5)+
  scale_color_manual(labels=c("No", "Yes"), values = c("#00BA38","red"
))+
  ggtitle("Heart Failure - UREA x EF")
```



Visualizing the probability plots for the main variables (UREA and EF) for different GENDER and assuming all the others as mean/median values.

```
#Calculating Means of Explanatory variables
#~HEART.FAILURE+factor(GENDER)+AGE+GLUCOSE+HB+TLC+CREATININE+UREA+EF+factor(RAISED.CARDIAC.ENZYMES)+factor(PRIOR.CMP)
medianTrainAGE = median(train$AGE)
medianTrainAGE

## [1] 62

meanTrainGLUCOSE = mean(train$GLUCOSE)
meanTrainGLUCOSE

## [1] 164.4254

meanTrainHB = mean(train$HB)
meanTrainHB

## [1] 12.35308

meanTrainTLC = mean(train$TLC)
meanTrainTLC

## [1] 11.63563
```

```
meanTrainCREATININE = mean(train$CREATININE)
meanTrainCREATININE
```

```
## [1] 1.304954
```

```
meanTrainUREA = mean(train$UREA)
meanTrainUREA
```

```
## [1] 47.58468
```

```
meanTrainEF = mean(train$EF)
meanTrainEF
```

```
## [1] 44.00532
```

```
medianTrainRAISED.CARDIAC.ENZYMES = median(as.integer(train$RAISED.CAR
DIAC.ENZYMES))
medianTrainRAISED.CARDIAC.ENZYMES
```

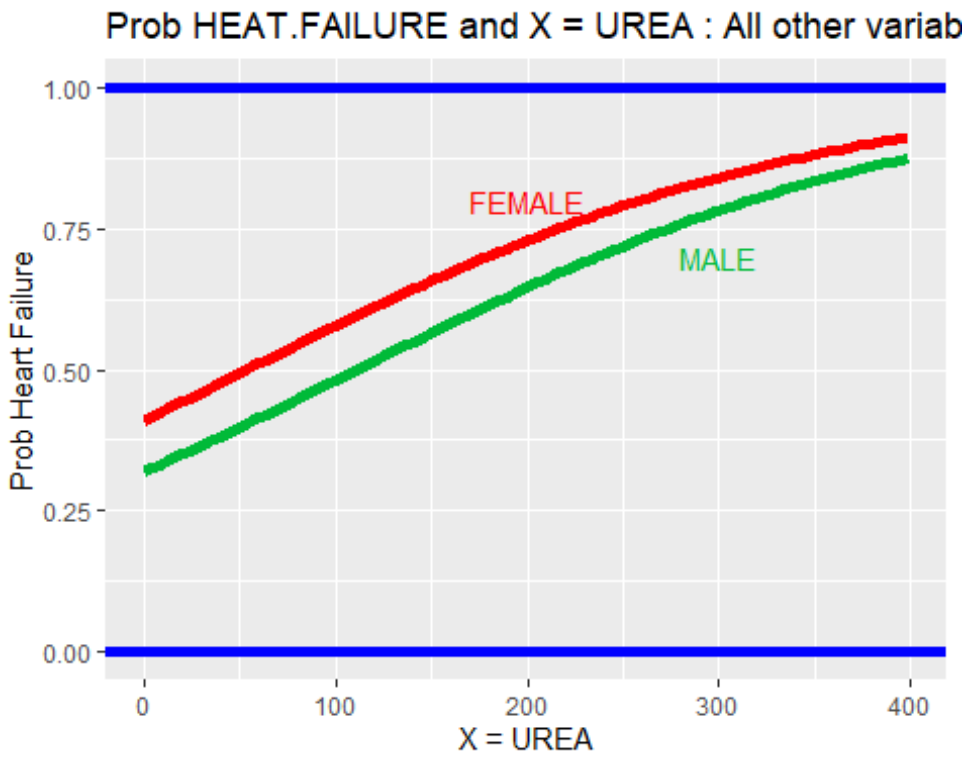
```
## [1] 1
```

```
medianTrainPRIOR.CMP = median(as.integer(train$PRIOR.CMP))
medianTrainPRIOR.CMP
```

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

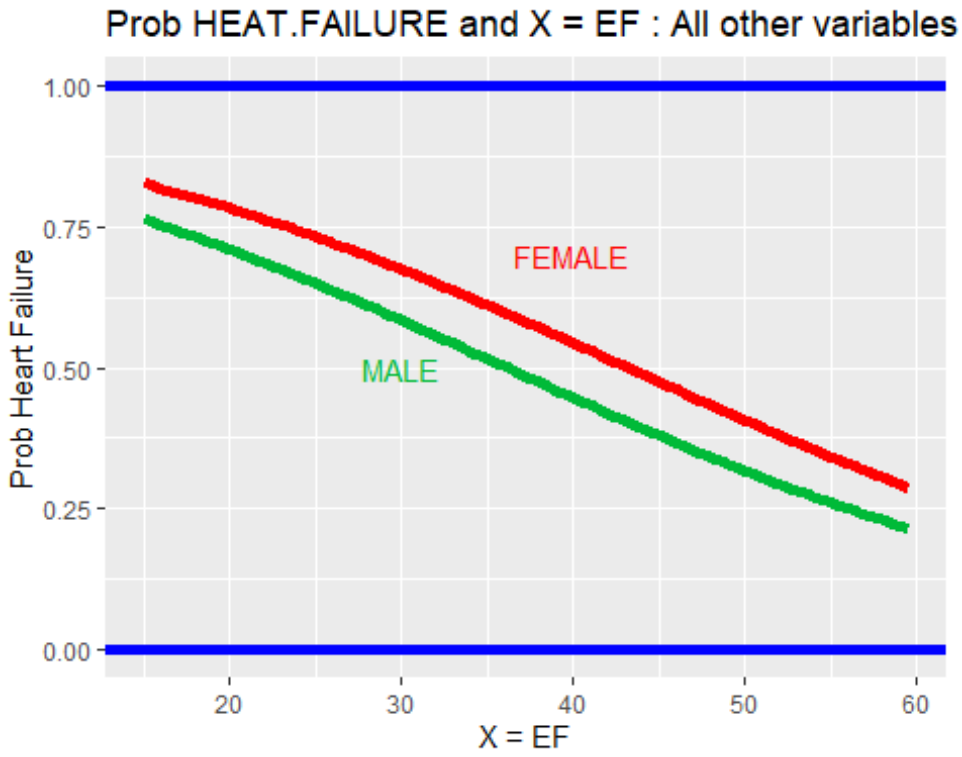
```
#Calculating probability of HEART FAILURE for each GENDER in function
of UREAM and keeping the othe parameters as median or mean

##### 1 - UREA
n = 100
pXvectorM=rep(0,n)
pXvectorF=rep(0,n)
valueV=rep(0,n)

count = 0
minV = min(train$UREA)
maxV = max(train$UREA)

for(i in seq(1:n+1)){
  valueV[i] = minV+(maxV-minV)*count/n

  # MAN
  genderV = 1
  expV = exp(0.7838871-0.390573*genderV+0.0156926*medianTrainAGE+0.001
2128*meanTrainGLUCOSE-0.0767528*meanTrainHB+
               0.0142416*meanTrainTLC-0.0966244*meanTrainCREATININE+0.
0067814*valueV[i]-0.0552717*meanTrainEF+
               0.4306312*medianTrainRAISED.CARDIAC.ENZYMES+0.5882911*m
```

```
edianTrainPRIOR.CMP)

  pXvectorM[i]=expV/(1+expV)

  # FEMALE
  genderV = 0
  expV = exp(0.7838871-0.390573*genderV+0.0156926*medianTrainAGE+0.001
2128*meanTrainGLUCOSE-0.0767528*meanTrainHB+
               0.0142416*meanTrainTLC-0.0966244*meanTrainCREATININE+0.
0067814*valueV[i]-0.0552717*meanTrainEF+
               0.4306312*medianTrainRAISED.CARDIAC.ENZYMES+0.5882911*m
edianTrainPRIOR.CMP)

  pXvectorF[i]=expV/(1+expV)


  count = count + 1
}

#pXvector


dfPlot <-data.frame(valueV,pXvectorM, pXvectorF)

#Generating UREA Probability plot
ggplot(data=dfPlot, aes(x=valueV) ) +
  geom_line(aes(y=pXvectorM),colour="#00BA38", size = 2)+
  geom_line(aes(y=pXvectorF),colour="red", size = 2)+
  annotate("text", x= 200, y=0.8, label = "FEMALE", color="Red")+
  annotate("text", x= 300, y=0.7, label = "MALE", color="#00BA38")+
  geom_hline(yintercept=0.0, color="blue", size=2)+
  geom_hline(yintercept=1, color="blue", size=2)+
  ggtitle("Prob HEAT.FAILURE and X = UREA : All other variables as mea
ns")+
  labs(x = "X = UREA", y = "Prob Heart Failure")
```




```
##### 2 - EF

n = 100
pXvectorM=rep(0,n)
pXvectorF=rep(0,n)
valueV=rep(0,n)

count = 0
minV = min(train$EF)
maxV = max(train$EF)

for(i in seq(1:n+1)){
  valueV[i] = minV+(maxV-minV)*count/n

  # MAN
  genderV = 1
  expV = exp(0.7838871-0.390573*genderV+0.0156926*medianTrainAGE+0.001
2128*meanTrainGLUCOSE-0.0767528*meanTrainHB+
               0.0142416*meanTrainTLC-0.0966244*meanTrainCREATININE+0.
0067814*meanTrainUREA-0.0552717*valueV[i]+
               0.4306312*medianTrainRAISED.CARDIAC.ENZYMES+0.5882911*m
edianTrainPRIOR.CMP)

  pXvectorM[i]=expV/(1+expV)
```

```
  # FEMALE
  genderV = 0
  expV = exp(0.7838871-0.390573*genderV+0.0156926*medianTrainAGE+0.001
2128*meanTrainGLUCOSE-0.0767528*meanTrainHB+
               0.0142416*meanTrainTLC-0.0966244*meanTrainCREATININE+0.
0067814*meanTrainUREA-0.0552717*valueV[i]+
               0.4306312*medianTrainRAISED.CARDIAC.ENZYMES+0.5882911*m
edianTrainPRIOR.CMP)

  pXvectorF[i]=expV/(1+expV)


  count = count + 1
}

#pXvector


dfPlot <-data.frame(valueV,pXvectorM, pXvectorF)

#Generating UREA Probability plot
ggplot(data=dfPlot, aes(x=valueV) ) +
  geom_line(aes(y=pXvectorM),colour="#00BA38", size = 2)+
  geom_line(aes(y=pXvectorF),colour="red", size = 2)+
  annotate("text", x= 40, y=0.7, label = "FEMALE", color="Red")+
  annotate("text", x= 30, y=0.5, label = "MALE", color="#00BA38")+
  geom_hline(yintercept=0.0, color="blue", size=2)+
  geom_hline(yintercept=1, color="blue", size=2)+
  ggtitle("Prob HEAT.FAILURE and X = EF : All other variables as means
")+
  labs(x = "X = EF", y = "Prob Heart Failure")
```



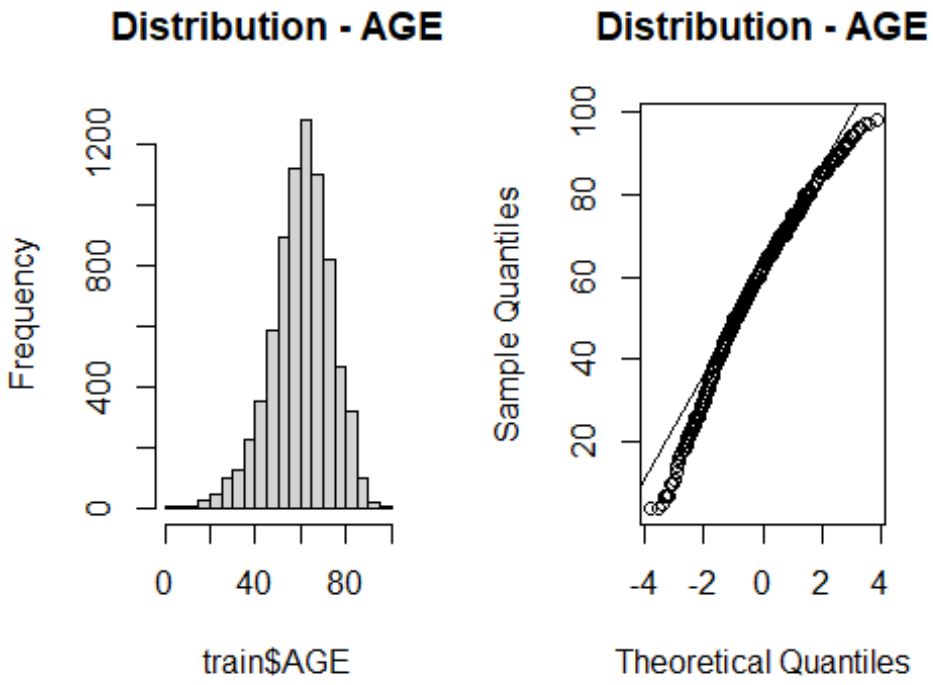
Based on the first plot in terms of UREA, the probability of FEMALE to have HEART.FAILURE varies between approximately 80% and 30% for values between the minimum and maximum UREA (between 0 and 400, respectively) values indicated in Train set keeping all the other relavant parameters as mean/median values in the train part. For MALE, HEART.FAILURE varies between approximately 30% and 87%, respectively.

For the second plot in terms of EF, the probability of FEMALE to have HEART.FAILURE varies between approximately 40% and 90% for values between the minimum and maximum EF (between 15 and 60, respectively) values indicated in Train set keeping all the other relavant parameters as mean/median values in the train part. For MALE, HEART.FAILURE varies between approximately 75 and 20%, respectively.

#(A.2) LINEAR DISCIMINATION ANALYSIS (LDA) For LDA, it is assumed that the quantitative variables follows a normal distribution, so we should check if the selected ones (AGE, GLUCOSE, HB, TLC, CREATININE, UREA and EF) are normally distributed.

```
# Checking if AGE is normally distributed by Kolmogorov-Smirnov Test
variableTest <- train$AGE
ks.test(train$AGE, "pnorm")

##
##  Asymptotic one-sample Kolmogorov-Smirnov test
##
## data:  train$AGE
```

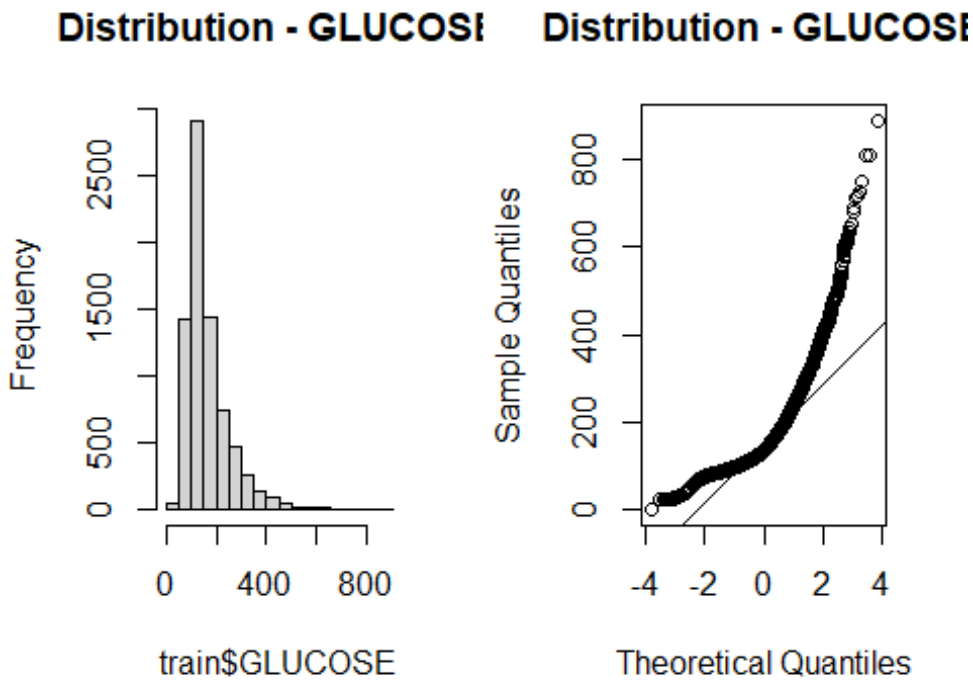
```
## D = 0.99997, p-value < 2.2e-16
## alternative hypothesis: two-sided

# Checking Shapiro.test (only accepts 5000 values)
shapiro.test(train$AGE[0:5000])

##
##  Shapiro-Wilk normality test
##
## data:  train$AGE[0:5000]
## W = 0.98511, p-value < 2.2e-16

#Define plot region
par(mfrow=c(1,2))

#Create histogram of the variable
hist(train$AGE, main="Distribution - AGE")
qqnorm(train$AGE, main="Distribution - AGE")
qqline(train$AGE)
```



Based on p-value (< 0.05), we should REJECT null hypothesis and conclude AGE is not normally distributed.

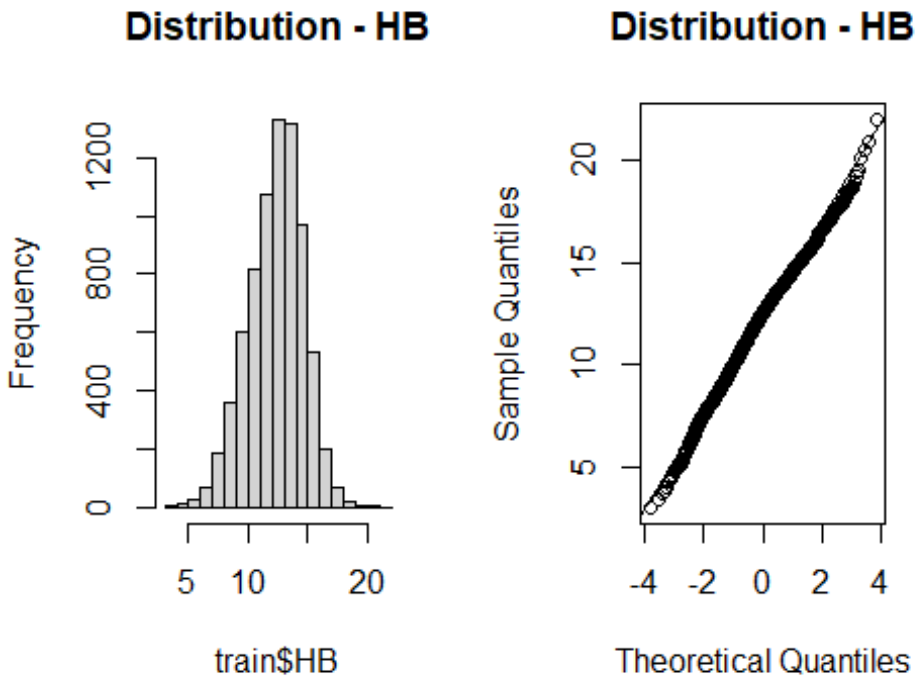
```
# Checking if GLUCOSE is normally distributed by Kolmogorov-Smirnov Te
st
variableTest <- train$GLUCOSE
ks.test(variableTest, "pnorm")
```

```
##
##  Asymptotic one-sample Kolmogorov-Smirnov test
##
## data:  variableTest
## D = 0.99987, p-value < 2.2e-16
## alternative hypothesis: two-sided

# Checking Shapiro.test (only accepts 5000 values)
shapiro.test(train$GLUCOSE[0:5000])

##
##  Shapiro-Wilk normality test
##
## data:  train$GLUCOSE[0:5000]
## W = 0.83045, p-value < 2.2e-16

#Define plot region
par(mfrow=c(1,2))

#Create histogram of the variable
hist(train$GLUCOSE, main="Distribution - GLUCOSE")
qqnorm(train$GLUCOSE, main="Distribution - GLUCOSE")
qqline(train$GLUCOSE)
```



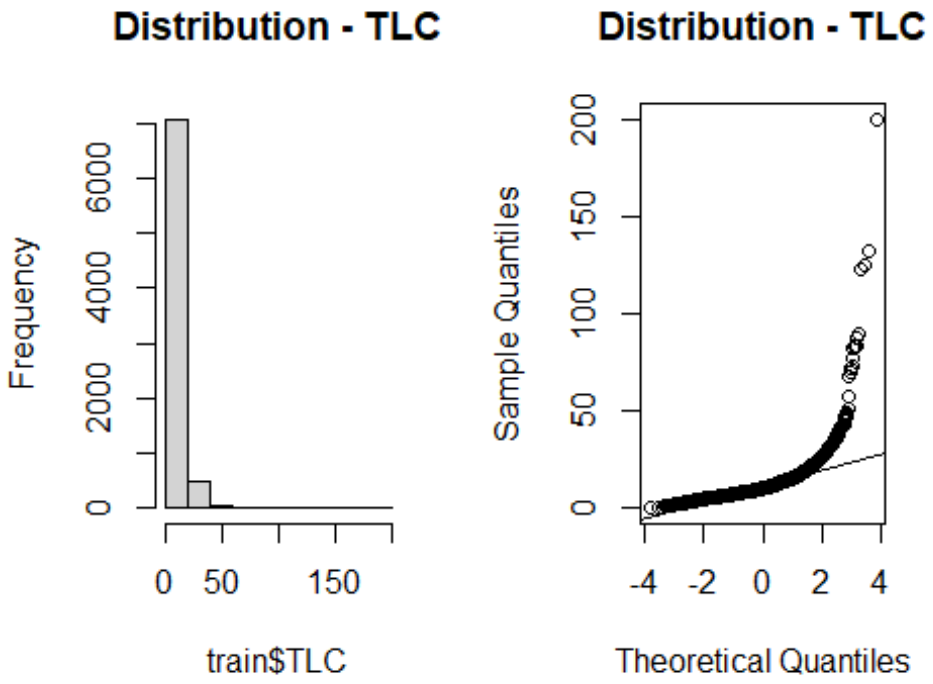
Based on p-value (< 0.05), we should REJECT null hypothesis and conclude GLUCOSE is not normally distributed.

```r
# Checking if HB is normally distributed by Kolmogorov-Smirnov Test
variableTest <- train$HB
ks.test(variableTest, "pnorm")

##
##  Asymptotic one-sample Kolmogorov-Smirnov test
##
## data:  variableTest
## D = 0.99963, p-value < 2.2e-16
## alternative hypothesis: two-sided

# Checking Shapiro.test (only accepts 5000 values)
shapiro.test(train$HB[0:5000])

##
##  Shapiro-Wilk normality test
##
## data:  train$HB[0:5000]
## W = 0.99549, p-value = 2.517e-11

#Define plot region
par(mfrow=c(1,2))

#Create histogram of the variable
hist(train$HB, main="Distribution - HB")
qqnorm(train$HB, main="Distribution - HB")
qqline(train$HB)
```



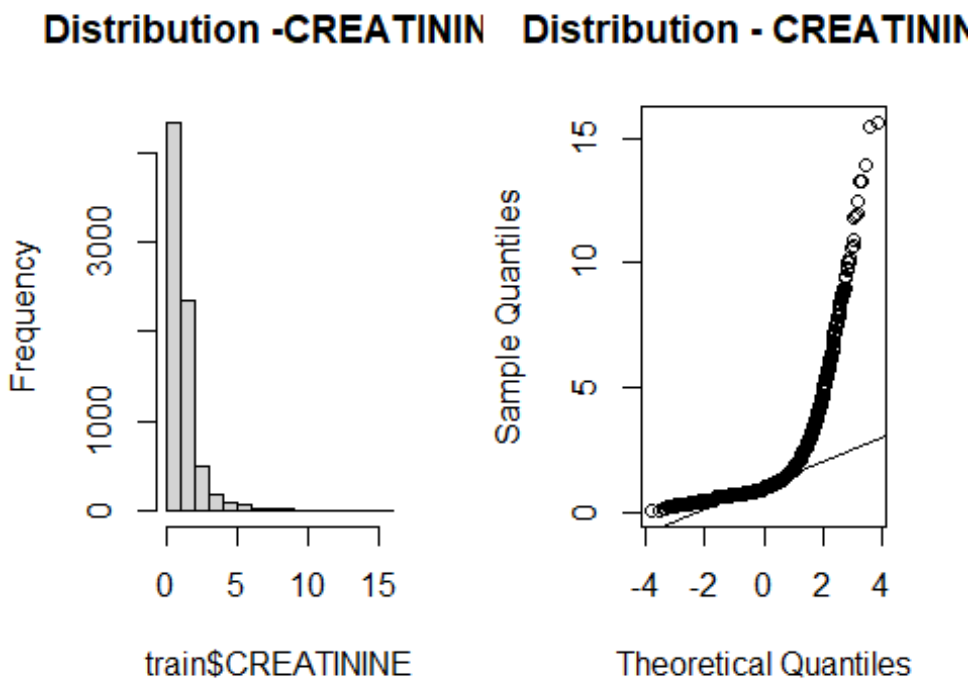
Based on p-value (< 0.05), we should REJECT null hypothesis and conclude HB is not normally distributed.

```
# Checking if TLC is normally distributed by Kolmogorov-Smirnov Test
variableTest <- train$TLC
ks.test(variableTest, "pnorm")

##
##  Asymptotic one-sample Kolmogorov-Smirnov test
##
## data:  variableTest
## D = 0.99418, p-value < 2.2e-16
## alternative hypothesis: two-sided

#Define plot region
par(mfrow=c(1,2))

#Create histogram of the variable
hist(train$TLC, main="Distribution - TLC")
qqnorm(train$TLC, main="Distribution - TLC")
qqline(train$TLC)
```



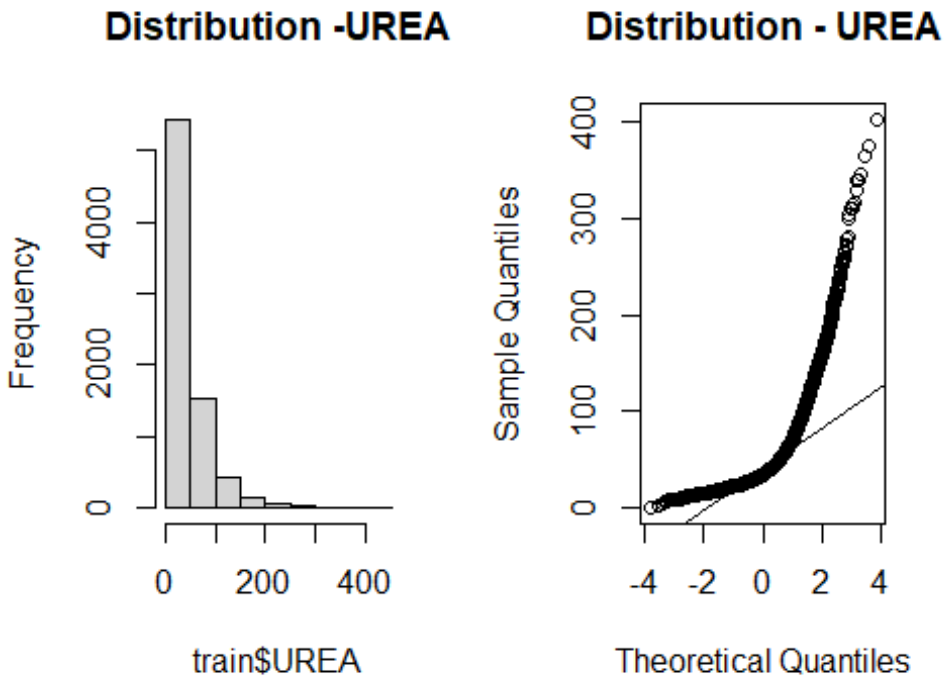
Based on p-value (< 0.05), we should REJECT null hypothesis and conclude TLC is not normally distributed.

```
# Checking if CREATININE is normally distributed by Kolmogorov-Smirnov Test
variableTest <- train$CREATININE
ks.test(variableTest, "pnorm")

##
##  Asymptotic one-sample Kolmogorov-Smirnov test
##
## data:  variableTest
## D = 0.67527, p-value < 2.2e-16
## alternative hypothesis: two-sided

#Define plot region
par(mfrow=c(1,2))

#Create histogram of the variable
hist(train$CREATININE, main="Distribution -CREATININE")
qqnorm(train$CREATININE, main="Distribution - CREATININE")
qqline(train$CREATININE)
```



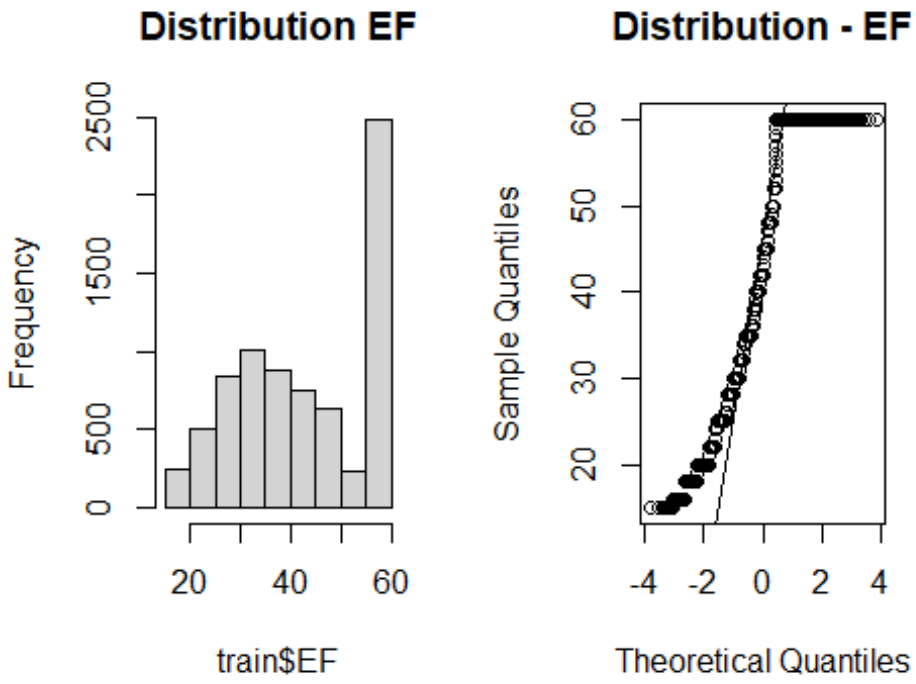
Based on p-value (< 0.05), we should REJECT null hypothesis and conclude CREATININE is not normally distributed.

```
# Checking if UREA is normally distributed by Kolmogorov-Smirnov Test
variableTest <- train$UREA
ks.test(variableTest, "pnorm")

##
##  Asymptotic one-sample Kolmogorov-Smirnov test
##
## data:  variableTest
## D = 0.99974, p-value < 2.2e-16
## alternative hypothesis: two-sided

#Define plot region
par(mfrow=c(1,2))

#Create histogram of the variable
hist(train$UREA, main="Distribution -UREA")
qqnorm(train$UREA, main="Distribution - UREA")
qqline(train$UREA)
```



Based on p-value (< 0.05), we should REJECT null hypothesis and conclude UREA is not normally distributed.

```
# Checking if EF is normally distributed by Kolmogorov-Smirnov Test
variableTest <- train$EF
ks.test(variableTest, "pnorm")

##
##  Asymptotic one-sample Kolmogorov-Smirnov test
##
## data:  variableTest
## D = 1, p-value < 2.2e-16
## alternative hypothesis: two-sided

#Define plot region
par(mfrow=c(1,2))

#Create histogram of the variable
hist(train$EF, main="Distribution EF")
qqnorm(train$EF, main="Distribution - EF")
qqline(train$EF)
```



Based on p-value (< 0.05), we should REJECT null hypothesis and conclude UREA is not normally distributed.

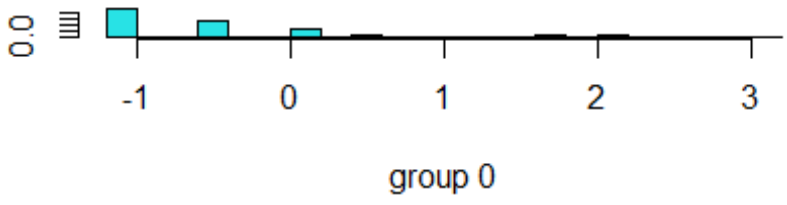
As it is required by LDA to have normality, we must not take into account none of the quantitative variables because the Kolmogorov-Smirnov test indicated that they are not normally distributed.

```
library(MASS)

#Creating the LDA model based on training part without CREATININE
HEART_lda.fit<-lda(HEART.FAILURE~factor(GENDER)+factor(RAISED.CARDIAC.
ENZYMES)+
                     factor(PRIOR.CMP), data = train)

HEART_lda.fit

## Call:
## lda(HEART.FAILURE ~ factor(GENDER) + factor(RAISED.CARDIAC.ENZYMES)
+
##     factor(PRIOR.CMP), data = train)
##
## Prior probabilities of groups:
##         0         1
## 0.7127996 0.2872004
##
## Group means:
##   factor(GENDER)M factor(RAISED.CARDIAC.ENZYMES)1 factor(PRIOR.CMP)
```

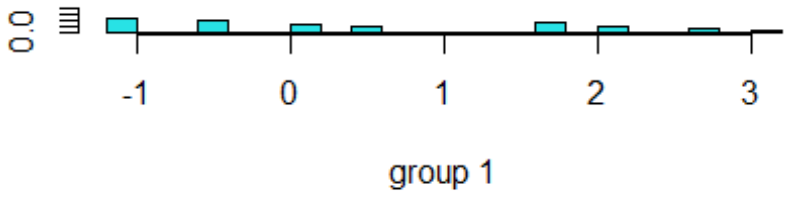
```
1
## 0       0.6545354                          0.1912064          0.085904
3
## 1       0.5910133                          0.3131591          0.335167
4
##
## Coefficients of linear discriminants:
##                                          LD1
## factor(GENDER)M                   -0.498434
## factor(RAISED.CARDIAC.ENZYMES)1    1.012301
## factor(PRIOR.CMP)1                 2.653732
```

The The prior probabilities are as expected when it was created the train part.The LDA output indicates that our prior probabilities are 0.7127996 and 0.2872004 or in other words, 71.3% of the training observations are patients who are not having HEART FAILURE and 28.7 % represent those that are having HEART FAILURE. It also provides the group means.

```
#Plotting visualize how are the distribution of "Y" and "N"
plot(HEART_lda.fit)
```



group 0



group 1

```
#Checking the prediction based on test part
HEART.pred<-predict(HEART_lda.fit,test)

#diabetes.pred
names(HEART.pred)
```

```
## [1] "class"     "posterior" "x"
```

```
#Plotting pairwise plot of the training set
#pairs(train)

# Checking the confusion table (Predicted versus real value in test pa
rt)
tablePred=table(HEART.pred$class, test$HEART.FAILURE)

tablePred
```

```
##
##        0    1
##   0 1653  493
##   1  151  234
```

1653 “N” were predicted correctly, whereas 493 were wrong. In terms of “Y”, 234 were predicted correctly, whereas 151 were wrong.. This was obtained assuming P(Y=pos|X1,…,Xp)≥0.5.

```
# Misclassification Ratio
mis_ratio = (tablePred[1, 2]+tablePred[2, 1])/(nrow(test))
mis_ratio
```

```
## [1] 0.2544449
```

The misclassification ratio is 0.25444449 (25.4%), in other words, 74.6% were predicted correctly the test part. This was obtained assuming P(Y=pos|X1,…,Xp)≥0.5.

#(A.3) QUADRATIC DISCRIMINATION ANALYSIS As QDA i not stric in terms of Normality, we added the quantitative variables for building the model.

```
# Creating the QDA model based on Train part (without CREATININE)
HEART_qda.fit<-qda(HEART.FAILURE~factor(GENDER)+AGE+GLUCOSE+HB+TLC+CRE
ATININE+UREA+
                     EF+factor(RAISED.CARDIAC.ENZYMES)+factor(PRIOR.CM
P), data = train)

HEART_qda.fit
```

```
## Call:
## qda(HEART.FAILURE ~ factor(GENDER) + AGE + GLUCOSE + HB + TLC +
##     CREATININE + UREA + EF + factor(RAISED.CARDIAC.ENZYMES) +
##     factor(PRIOR.CMP), data = train)
##
## Prior probabilities of groups:
##         0         1
## 0.7127996 0.2872004
```

```
##
## Group means:
##   factor(GENDER)M      AGE  GLUCOSE       HB      TLC CREATININE
UREA
## 0       0.6545354 60.04545 157.8815 12.57658 11.17966   1.200482 42
.06529
## 1       0.5910133 63.93994 180.6666 11.79837 12.76730   1.564241 61
.28317
##         EF factor(RAISED.CARDIAC.ENZYMES)1 factor(PRIOR.CMP)1
## 0 47.29012                       0.1912064          0.0859043
## 1 35.85282                       0.3131591          0.3351674
```

```
# Applying the model to the test data to predict the response variable
HEART.pred<-predict(HEART_qda.fit, test)$class

tablePred=table(HEART.pred, test$HEART.FAILURE)

tablePred

##
## HEART.pred    0    1
##          0 1546  402
##          1  258  325
```

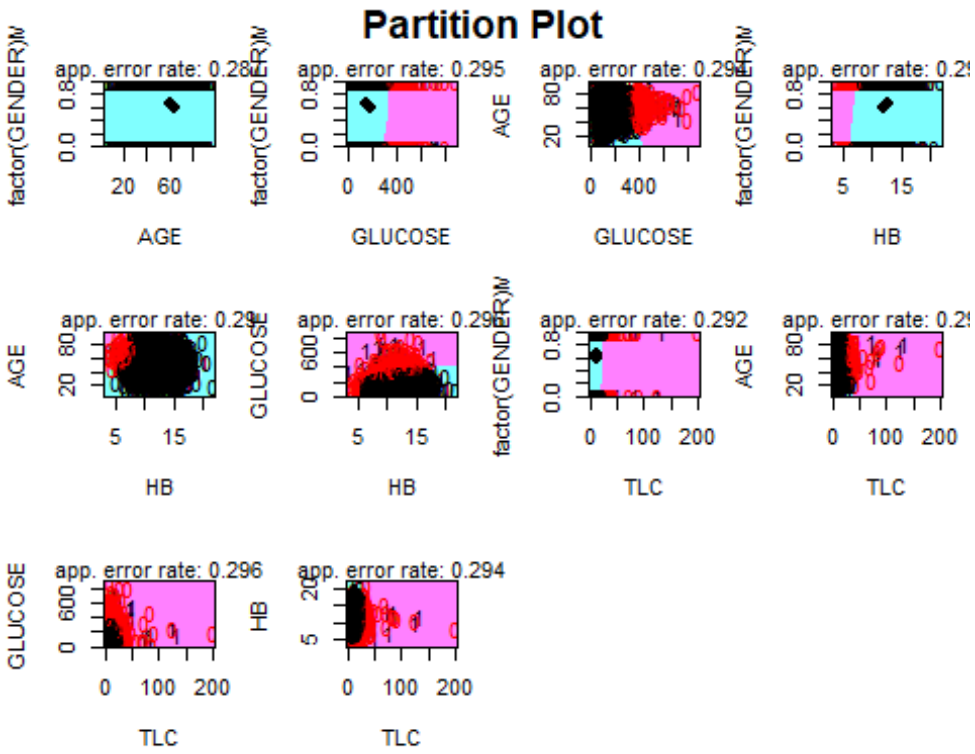
1546 “N” were predicted correctly, whereas 402 were wrong. In terms of “Y”,325 were predicted correctly, whereas 258 were wrong.. This was obtained assuming P(Y=pos|X1,…,Xp)≥0.5.

```
# Misclassification Ratio
mis_ratio = (tablePred[1, 2]+tablePred[2, 1])/(nrow(test))
mis_ratio

## [1] 0.2607665
```

The misclassification ratio is 0.2607665 (26.1), in other words, 73.9% were predicted correctly the test part. This was obtained assuming P(Y=pos|X1,…,Xp)≥0.5.

Checking the error of some pair parameters (with only some explanatory variablesseeing that shows the folowing messagewitheverything: Error in plot.new() : figure margins too large).

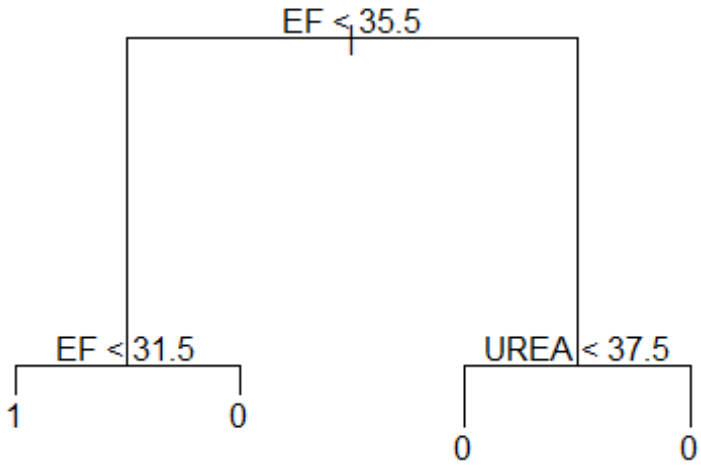
```
library(klaR)
partimat(HEART.FAILURE~factor(GENDER)+AGE+GLUCOSE+HB+TLC, data=train,
method="qda")
```



#(A.4) CLASSIFICATION TREE

```
library(tree)

# Doing the Classification Tree
HEART_tree.fit<-tree(HEART.FAILURE~factor(GENDER)+AGE+GLUCOSE+HB+TLC+C
REATININE+
                       UREA+EF+factor(RAISED.CARDIAC.ENZYMES)+factor(P
RIOR.CMP), train)


summary(HEART_tree.fit)

##
## Classification tree:
## tree(formula = HEART.FAILURE ~ factor(GENDER) + AGE + GLUCOSE +
##     HB + TLC + CREATININE + UREA + EF + factor(RAISED.CARDIAC.ENZYM
ES) +
##     factor(PRIOR.CMP), data = train)
## Variables actually used in tree construction:
## [1] "EF"   "UREA"
## Number of terminal nodes:  4
## Residual mean deviance:  1.019 = 7735 / 7590
## Misclassification error rate: 0.2447 = 1858 / 7594
```

```
#Plotting the tree
plot(HEART_tree.fit)
text(HEART_tree.fit ,pretty =0)
```



The tree only has 4 terminal nodes, with only 2 class (EF and UREA) as splitting rules. In addition, it is possible to see on the right side (EF > 37.5), if an observation falls there, it will always indicated NO heart failure.

Let us check the probability in each terminal node.

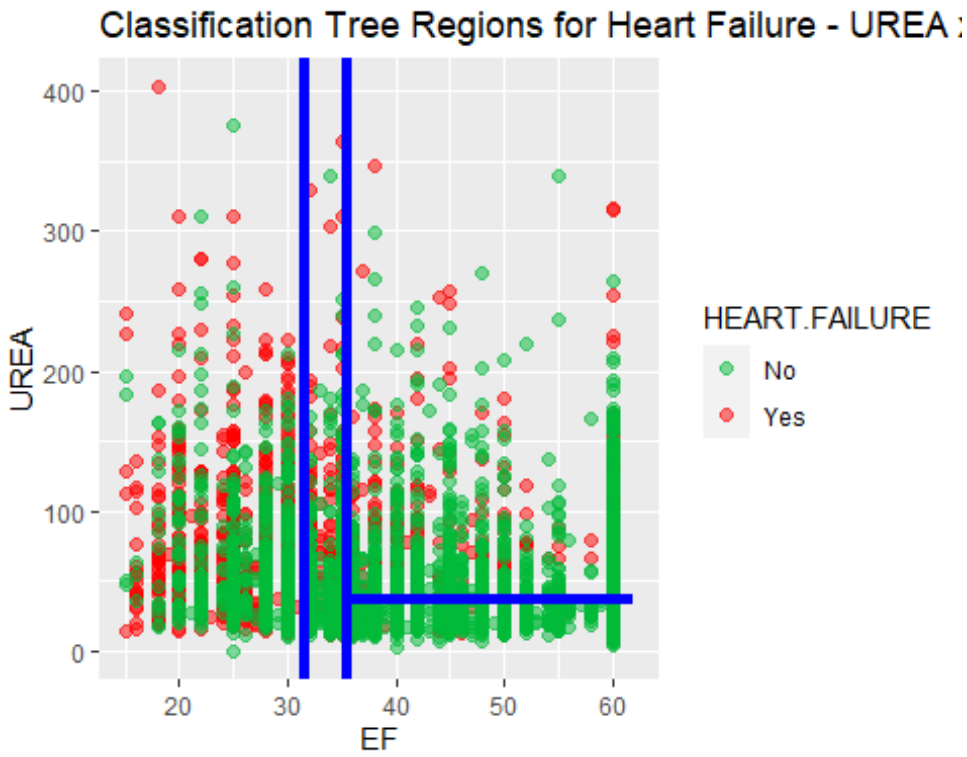
```
# Check the nodes of the tree
HEART_tree.fit
```

```
## node), split, n, deviance, yval, (yprob)
##       * denotes terminal node
##
## 1) root 7594 9107 0 ( 0.7128 0.2872 )
##   2) EF < 35.5 2605 3604 1 ( 0.4737 0.5263 )
##     4) EF < 31.5 1593 2142 1 ( 0.3986 0.6014 ) *
##     5) EF > 31.5 1012 1369 0 ( 0.5919 0.4081 ) *
##   3) EF > 35.5 4989 4426 0 ( 0.8376 0.1624 )
##     6) UREA < 37.5 3225 2185 0 ( 0.8936 0.1064 ) *
##     7) UREA > 37.5 1764 2039 0 ( 0.7353 0.2647 ) *
```

It is indicated that the probability of "Y" on the right side of the tree is significant smaller than "N", mainly in case of UREA < 34.5. For the left side, it is not too different the probabilities, and consequently it is possible the majority of wrong predictions happen there.

The plot below, we can visualize the tree regions with the train points there.

```
#Generating plots with regions
ggplot(data=train, aes(x=EF, y=UREA, color=HEART.FAILURE))+
  geom_point(size=2, alpha=0.5)+
  scale_color_manual(labels=c("No", "Yes"), values = c("#00BA38","red"
))+
  geom_vline(xintercept=35.5, color="blue", size=2)+
  geom_vline(xintercept=31.5, color="blue", size=2)+
  geom_segment(aes(x=35.5,y=37.5, xend=62,yend=37.5), color="blue", si
ze=2)+
  ggtitle("Classification Tree Regions for Heart Failure - UREA x EF")
```



Apply the tree to the test set, calculate the root square of the mean squared error (RMSE)

```
# Set a seed for random number generation
set.seed(10)

# Applying the unpruned tree to test part
HEART_tree.pred<-predict(HEART_tree.fit,test,type = "class")
tablePred=table(HEART_tree.pred,test$HEART.FAILURE)
```

```
tablePred
```

```
##
## HEART_tree.pred    0    1
##               0 1605  421
##               1  199  306
```

1605 “N” were predicted correctly, whereas 421 were wrong. In terms of “Y”, 306 were predicted correctly, whereas 199 were wrong.

```
# Misclassification Ratio
mis_ratio = (tablePred[1, 2]+tablePred[2, 1])/(nrow(test))
mis_ratio
```

```
## [1] 0.2449625
```

The misclassification ratio is 0.2449625 (24.5%), in other words, 76.5% were predicted correctly the test part.

It was not necessary to prune the tree seeing that just few terminal nodes were provided.

#(A.5) LOGISTIC REGRESSION with stratified 10-fold cross-validation

Randomly dividing the total clean observations in folds with approximatelly equal size as well as proportion of Heart Failure of “Y” and “N”.

```
library(caret)

# Set a seed for random number generation
set.seed(10)

# Creating 10 folds
folds<-createFolds(as.factor(dataClean$HEART.FAILURE), k=10)

#Checking the units in each fold
fold01<-dataClean[folds$Fold01,]
fold02<-dataClean[folds$Fold02,]
fold03<-dataClean[folds$Fold03,]
fold04<-dataClean[folds$Fold04,]
fold05<-dataClean[folds$Fold05,]
fold06<-dataClean[folds$Fold06,]
fold07<-dataClean[folds$Fold07,]
fold08<-dataClean[folds$Fold08,]
fold09<-dataClean[folds$Fold09,]
fold10<-dataClean[folds$Fold10,]
table(fold01$HEART.FAILURE)
```

```
##
##   0   1
## 721 291

table(fold02$HEART.FAILURE)

##
##   0   1
## 722 291

table(fold03$HEART.FAILURE)

##
##   0   1
## 722 291

table(fold04$HEART.FAILURE)

##
##   0   1
## 721 291

table(fold05$HEART.FAILURE)

##
##   0   1
## 721 291

table(fold06$HEART.FAILURE)

##
##   0   1
## 722 291

table(fold07$HEART.FAILURE)

##
##   0   1
## 722 291

table(fold08$HEART.FAILURE)

##
##   0   1
## 722 290

table(fold09$HEART.FAILURE)
```

```
##
##   0   1
## 722 290

table(fold10$HEART.FAILURE)

##
##   0   1
## 722 291

#folds
```

It is possible to verify that HEART.FAILURE responses were approximately equally distributed.

```
# Creating function to calculate misclassification rate for Logistic R
egression
library(MASS)

misclassification_LOGISTIC<-function(idx){
  # Select the other folders to training part
  trainFold<-dataClean[-idx,]

  # The current fold as the validation (test) part
  validationFold<-dataClean[idx,]

  #Fit the Logistic Regression model for the training Fold part
  fit_LOGISTICmodel<-glm(HEART.FAILURE~factor(GENDER)+AGE+GLUCOSE+HB+T
LC+CREATININE+
                           UREA+EF+factor(RAISED.CARDIAC.ENZYMES)+fact
or(PRIOR.CMP),
                         family=binomial, data=trainFold)


  # Applying the validation part to the fitted model and predict the H
EART FAILURE
  pred<-predict(fit_LOGISTICmodel,validationFold)

  HEART_FAILURE.predict=rep("0", nrow(validationFold))

  HEART_FAILURE.predict[pred >= 0.5]="1"


  #return the mean error of the prediction for the idx fold
  return(1-mean(HEART_FAILURE.predict==validationFold$HEART.FAILURE))
}

# Set a seed for random number generation
set.seed(10)
```

```r
# calculating the misclassification rate of each fold - Logistic Regre
ssion
mis_rate_Logistic=lapply(folds, misclassification_LOGISTIC)

# calculating the misclassification rate for each fold
#mis_rate_LOGISTIC

# Calculating the average misclassification rate
mean(as.numeric(mis_rate_Logistic))
```

```
## [1] 0.2568885
```

For Logistic Regression stratified with 10-folds, the misclassification rate is 0.256885 (25.7%).

#(A.6) LDA with stratified 10-fold cross-validation For LDA whch assumes normality of explanatory variables, based on previous analysis with Kolmogorov-Smirnov test, we could see that none of quantitative variables are normally distributed, and consequently we need to remove them.

```r
# Creating function to calculate misclassification rate for LDA
library(MASS)

misclassification_LDA<-function(idx){
  # Select the other folders to training part
  trainFold<-dataClean[-idx,]

  # The current fold as the validation (test) part
  validationFold<-dataClean[idx,]

  #Fit the LDA model for the training Fold part
  fit_LDAmodel<-lda(HEART.FAILURE~factor(GENDER)+factor(RAISED.CARDIAC
.ENZYMES)+
                      factor(PRIOR.CMP), data=trainFold)

  # Applying the validation part to the fitted model and predict the T
ype
  pred<-predict(fit_LDAmodel,validationFold)

  #return the mean error of the prediction for the idx fold
  return(1-mean(pred$class==validationFold$HEART.FAILURE))
}

# Set a seed for random number generation
set.seed(10)

# calculating the misclassification rate of each fold - LDA
mis_rate_LDA=lapply(folds, misclassification_LDA)
```

```
# calculating the misclassification rate for each fold
#mis_rate_LDA

# Calculating the average misclassification rate
mean(as.numeric(mis_rate_LDA))
```

```
## [1] 0.2527416
```

For LDA stratified with 10-folds, the misclassification rate is 0.2527416 (25.3%).

#(A.7) QDA with stratified 10-fold cross-validation

```
# Creating function to calculate misclassification rate for QDA
library(MASS)

misclassification_LDA<-function(idx){
  # Select the other folders to training part
  trainFold<-dataClean[-idx,]

  # The current fold as the validation (test) part
  validationFold<-dataClean[idx,]

  #Fit the QDA model for the training Fold part
  fit_QDAmodel<-qda(HEART.FAILURE~factor(GENDER)+AGE+GLUCOSE+HB+TLC+CR
EATININE+
                      UREA+EF+factor(RAISED.CARDIAC.ENZYMES)+factor(PR
IOR.CMP),
                    data=trainFold)

  # Applying the validation part to the fitted model and predict the T
ype
  pred<-predict(fit_QDAmodel,validationFold)

  #return the mean error of the prediction for the idx fold
  return(1-mean(pred$class==validationFold$HEART.FAILURE))
}

# Set a seed for random number generation
set.seed(10)

# calculating the misclassification rate of each fold - QDA
mis_rate_QDA=lapply(folds, misclassification_LDA)

# calculating the misclassification rate for each fold
#mis_rate_QDA
```

```
# Calculating the average misclassification rate
mean(as.numeric(mis_rate_QDA))
```

```
## [1] 0.2572843
```

For QDA stratified with 10-folds, the misclassification rate is 0.257284 (25.7%).

#(A.8) Classification Tree with stratified 10-fold cross-validation

```
# Creating function to calculate misclassification rate for Classifica
tion Tree
library(tree)

misclassification_TREE<-function(idx){
  # Select the other folders to training part
  trainFold<-dataClean[-idx,]

  # The current fold as the validation (test) part
  validationFold<-dataClean[idx,]

  #Fit the Tree model for the training Fold part
  fit_TREEmodel<-tree(HEART.FAILURE~factor(GENDER)+AGE+GLUCOSE+HB+TLC+
CREATININE+
                        UREA+EF+factor(RAISED.CARDIAC.ENZYMES)+factor(
PRIOR.CMP),
                      data=trainFold)

  # Applying the validation part to the fitted model and predict the T
ype
  pred<-predict(fit_TREEmodel,validationFold, type = "class")


  #return the mean error of the prediction for the idx fold
  return(1-mean(pred==validationFold$HEART.FAILURE))
}

# Set a seed for random number generation
set.seed(10)

# calculating the misclassification rate of each fold - Classification
Tree
mis_rate_ClassificationTREE=lapply(folds, misclassification_TREE)

# calculating the misclassification rate for each fold
#mis_rate_ClassificationTREE

# Calculating the average misclassification rate
mean(as.numeric(mis_rate_ClassificationTREE))
```

```
## [1] 0.2635064
```

For Classification tree considering stratified with 10-folds, the misclassification rate is 0.2635064 (26.4%).

#(A.9) LOGISTIC REGRESSION only with relevant variables (UREA and EF) indicated by CLASSIFICATION TREE (Professor's suggestion)

```
#Creating the the logistic regression model based on train part
HEART_logistic<-glm(HEART.FAILURE~UREA+EF, family=binomial, data=train
)

summary(HEART_logistic)

##
## Call:
## glm(formula = HEART.FAILURE ~ UREA + EF, family = binomial, data =
train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.3912  -0.8002  -0.4570   0.9735   2.2568
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  1.4250431  0.1066229   13.37   <2e-16 ***
## UREA         0.0080688  0.0007208   11.20   <2e-16 ***
## EF          -0.0660435  0.0023088  -28.61   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 9107.1  on 7593  degrees of freedom
## Residual deviance: 7785.0  on 7591  degrees of freedom
## AIC: 7791
##
## Number of Fisher Scoring iterations: 4
```

Applying Test part to the fitted logistic regression model

```
#Based on the model fitted based on the training data, predict the HEA
RT.FAILURE (Y) based on test data
Prob.predict_logistic<-predict(HEART_logistic,test,type="response")

testSize = nrow(test)
testSize
```

```
## [1] 2531
```

```
HEART_FAILURE.predict=rep("0", testSize)

HEART_FAILURE.predict[Prob.predict_logistic >= 0.5]="1"

# Checking the HEART_FAILURE prediction of Y=1 and N = 0 in the test d
ata
table(HEART_FAILURE.predict)
```

```
## HEART_FAILURE.predict
##    0    1
## 2165  366
```

```
#Comparing what the tests responses with the actual
actual=test$HEART.FAILURE
tablePred=table(HEART_FAILURE.predict,actual)

tablePred
```

```
##                       actual
## HEART_FAILURE.predict    0    1
##                     0 1675  490
##                     1  129  237
```

1675 “N” were predicted correctly, whereas 490 were wrong. In terms of “Y”, 237 were predicted correctly, whereas 129 were wrong. This considering p=0.5.

```
# Misclassification Ratio
mis_ratio = (tablePred[1, 2]+tablePred[2, 1])/(nrow(test))
mis_ratio
```

```
## [1] 0.2445674
```

The misclassification ratio is 0.2445674 (24.5%), in other words, 75.5% were predicted correctly in the test part. This was obtained considering p=0.5.

Considering Cross-validation

```
# Creating function to calculate misclassification rate for Logistic R
egression
library(MASS)

misclassification_LOGISTIC<-function(idx){
  # Select the other folders to training part
  trainFold<-dataClean[-idx,]

  # The current fold as the validation (test) part
  validationFold<-dataClean[idx,]
```

```
  #Fit the Logistic Regression model for the training Fold part
  fit_LOGISTICmodel<-glm(HEART.FAILURE~UREA+EF, family=binomial, data=
trainFold)


  # Applying the validation part to the fitted model and predict the H
EART FAILURE
  pred<-predict(fit_LOGISTICmodel,validationFold)

  HEART_FAILURE.predict=rep("0", nrow(validationFold))

  HEART_FAILURE.predict[pred >= 0.5]="1"


  #return the mean error of the prediction for the idx fold
  return(1-mean(HEART_FAILURE.predict==validationFold$HEART.FAILURE))
}

# Set a seed for random number generation
set.seed(10)

# calculating the misclassification rate of each fold - Logistic Regre
ssion
mis_rate_Logistic=lapply(folds, misclassification_LOGISTIC)

# calculating the misclassification rate for each fold
#mis_rate_LOGISTIC

# Calculating the average misclassification rate
mean(as.numeric(mis_rate_Logistic))

## [1] 0.2662715
```

The misclassification rate is 0.2662715 (26.2715%) considering Logistic Regression with only the variables indicated by Classification Tree, in other words, 73.6% were predicted correctly in the test part. This was obtained considering p=0.5.

#(A.10) SUMMARY oF HEART.FAILURE MODELS

The HEART.FAILURE models with the relevant explanatory variables are GENDER, AGE, HB, TLC, CREATININE, UREA, EF, RAISED.CARDIACENZYMES and PRIOR.CMP (Except LDAs, that just qualitative variables were taken into account due to no normality of the quantivative variables) presented the following misclassification rates for different statistical learning methods and using or not k-flod cross validation:

1) 75% Train part and 25% Prediction Part (no k-Fold Cross-Validation) LOGISTIC REGRESSION: Misclassification Rate = 23.9% LDA: Misclassification Rate = 25.4% QDA: Misclassification Rate =

26.1% Classifiation Tree: Misclassification Rate = 24.5% LOGISTIC REGRESSION with only UREA and EF: Misclassification Rate = 24.5%

2) 10–Fold Cross-Validation LOGISTIC REGRESSION: Misclassification Rate = 25.7% LDA: Misclassification Rate = 25.3% QDA: Misclassification Rate = 25.7% Classifiation Tree: Misclassification Rate = 26.4% LOGISTIC REGRESSION with only UREA and EF: Misclassification Rate = 26.6%

Based on the results above, LOGISTIC REGRESSION without Cross Validation that indicated the best performance (misclassification rate of 23.9%). However, the performance of others models were not too different. It is interesting to see the most relevant explanatory variables to take into account are EF and UREA, but considering also GENDER, AGE, HB, TLC, CREATININE, RAISED.CARDIACENZYMES and PRIOR.CMP can improve some models,mainly Logistic Regression.

IV.1.2 - AKI

For AKI, it was defined to use all the quantitative variables and the categorical variables and Gender (based on EDA) and RAISED.CARDIAC ENZYMES, PRIOR.CMP,(one of the possibility of models obtained with independence analysis with Contingency Table). And another set will be using STEMI, DM with Gender.

#Evaluating “AKI” IN FUNCTION OF OTHER VARIABLES (only Quantitatives + GENDER + STEMI and DM)

Sampling (75% train_AKI part, 25% test_AKI part)

```
# Set a seed for random number generation
set.seed(10)

#Considering stratified sampling to separate each
library(sampling)

#Checking the order and total of each Gender
unique(dataClean$AKI)

## [1] 0 1
## Levels: 0 1

table(dataClean$AKI) #>>>>>>>> s

##
##    0    1
## 7981 2144

# 75% with AKI = 1 and 0
n_AKI_Y = round(0.75*nrow(dataClean[dataClean$AKI == "1",]))
n_AKI_N = round(0.75*nrow(dataClean[dataClean$AKI == "0",]))
```

```
# srswor = Simple Random Sampling without replacement for each strata:
7% of Total for train_AKIing
idx_AKI<-sampling::strata(dataClean, stratanames = ("AKI"), size=c(n_A
KI_N,n_AKI_Y), method="srswor")
head(idx_AKI)
```

```
##   AKI ID_unit      Prob Stratum
## 1   0       1 0.7500313       1
## 2   0       2 0.7500313       1
## 5   0       5 0.7500313       1
## 6   0       6 0.7500313       1
## 8   0       8 0.7500313       1
## 9   0       9 0.7500313       1
```

```
tail(idx_AKI)
```

```
##       AKI ID_unit Prob Stratum
## 10085   1   10085 0.75       2
## 10089   1   10089 0.75       2
## 10100   1   10100 0.75       2
## 10107   1   10107 0.75       2
## 10123   1   10123 0.75       2
## 10124   1   10124 0.75       2
```

```
train_AKI=dataClean[idx_AKI$ID_unit,]

# Checking if the stratified sample was done correctly in train_AKI pa
rt
table(train_AKI$AKI)
```

```
##
##    0    1
## 5986 1608
```

```
# Creating the test_AKI part (25% of Total, other rows not selected fr
om train_AKIinng part)
test_AKI=dataClean[-idx_AKI$ID_unit,]

table(test_AKI$AKI)
```

```
##
##    0    1
## 1995  536
```

The stratified sampling separated the total population as expected.

#(B.1) LOGISTIC REGRESSION STEMI and DM

```r
library(ISLR)

#Creating the the logistic regression model based on train_AKI part
AKI_logistic<-glm(AKI~factor(GENDER)+AGE+GLUCOSE+HB+TLC+PLATELETS+UREA
+CREATININE
                  +EF+factor(STEMI)+factor(DM),family=binomial, data=t
rain_AKI)

summary(AKI_logistic)

##
## Call:
## glm(formula = AKI ~ factor(GENDER) + AGE + GLUCOSE + HB + TLC +
##     PLATELETS + UREA + CREATININE + EF + factor(STEMI) + factor(DM)
,
##     family = binomial, data = train_AKI)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -1.91423  -0.00001   0.00000   0.00000   2.75900
##
## Coefficients:
##                    Estimate Std. Error z value Pr(>|z|)
## (Intercept)     -7.163e+01  6.876e+00 -10.418  < 2e-16 ***
## factor(GENDER)M  2.010e-01  3.574e-01   0.562  0.57381
## AGE             -1.178e-02  1.381e-02  -0.853  0.39360
## GLUCOSE         -3.506e-03  1.587e-03  -2.210  0.02714 *
## HB               4.864e-02  7.683e-02   0.633  0.52668
## TLC              1.104e-02  1.380e-02   0.800  0.42345
## PLATELETS       -4.668e-04  1.624e-03  -0.288  0.77372
## UREA            -1.840e-03  6.666e-03  -0.276  0.78254
## CREATININE       4.861e+01  4.480e+00  10.849  < 2e-16 ***
## EF              -7.922e-03  1.127e-02  -0.703  0.48209
## factor(STEMI)1   2.676e-01  4.060e-01   0.659  0.50979
## factor(DM)1      9.513e-01  3.399e-01   2.798  0.00514 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 7840.97  on 7593  degrees of freedom
## Residual deviance:  299.23  on 7582  degrees of freedom
## AIC: 323.23
##
## Number of Fisher Scoring iterations: 13
```

Except CREATININE, GLUCOSE and DM (p-value < 0.05), all the other variables are not significant at 5% level due to (p-value > 0.05).

Checking the coefficient to see if there is some 0 at 95% Confidence Interval.

```
# Calculating the confidence interval on the coefficients
confint(AKI_logistic, level = 0.95)
```

```
##                          2.5 %        97.5 %
## (Intercept)     -86.712758255 -5.948998e+01
## factor(GENDER)M  -0.500048280  9.058572e-01
## AGE              -0.039178770  1.517540e-02
## GLUCOSE          -0.006730643 -4.703392e-04
## HB               -0.102856800  1.994492e-01
## TLC              -0.028050570  3.210482e-02
## PLATELETS        -0.003655352  2.700691e-03
## UREA             -0.015028705  1.108617e-02
## CREATININE       40.722370660  5.846763e+01
## EF               -0.030169173  1.416156e-02
## factor(STEMI)1   -0.518626137  1.079809e+00
## factor(DM)1       0.297630338  1.634966e+00
```

Except GLUCOSE, CREATININE and DM, all other variables with ZERO between UPPER and LOWER bounds of 95% Confidence Interval. So they should be removed.

Redoing Logistic Regression with only GLUCOSE, CREATININE and DM.

```
#Creating the the logistic regression model based on train_AKI part

AKI_logistic<-glm(AKI~CREATININE+GLUCOSE+factor(DM), family=binomial,
data=train_AKI)
summary(AKI_logistic)
```

```
##
## Call:
## glm(formula = AKI ~ CREATININE + GLUCOSE + factor(DM), family = bin
omial,
##     data = train_AKI)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -1.80962  -0.00001   0.00000   0.00000   2.79040
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -71.102440   6.553882 -10.849  < 2e-16 ***
## CREATININE   48.002438   4.397677  10.915  < 2e-16 ***
## GLUCOSE      -0.003436   0.001510  -2.275  0.02290 *
```

```
## factor(DM)1   0.856600   0.317553   2.698  0.00699 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 7840.97  on 7593  degrees of freedom
## Residual deviance:  304.16  on 7590  degrees of freedom
## AIC: 312.16
##
## Number of Fisher Scoring iterations: 13
```

```
# Calculating the confidence interval on the coefficients
confint(AKI_logistic, level = 0.95)
```

```
##                     2.5 %        97.5 %
## (Intercept) -85.552749319 -5.958838e+01
## CREATININE   40.268174639  5.768737e+01
## GLUCOSE      -0.006508921 -5.467129e-04
## factor(DM)1   0.245210247  1.494168e+00
```

We see, Creatinine, DM, glucose variables are significant and without 0 in any coefficient at 95% Confidence Interval

Checking if there is problem of Multicolinearity in the training set.

```
#Checking Multicollinearity
library(car)

vif(AKI_logistic)
```

```
## CREATININE    GLUCOSE factor(DM)
##   1.026479   1.119551   1.133526
```

It was detected no collinearity (or very very mild) between CREATININE, GLUCOSE and DM as VIF values are less than 2, So we are keeping them.

Applying test_AKI part to the fitted logistic regression model

```
#Based on the model fitted based on the train_AKIing data, predict the AKI (Y) based on test_AKI data
Prob.predict_logistic_AKI<-predict(AKI_logistic,test_AKI,type="response")

testSize_AKI = nrow(test_AKI)
testSize_AKI
```

```
## [1] 2531
```

```
AKI.predict=rep("0", testSize_AKI)

AKI.predict[Prob.predict_logistic_AKI >= 0.5]="1"

# Checking the HEART_FAILURE prediction of Y=1 and N = 0 in the test_A
KI data
table(AKI.predict)

## AKI.predict
##    0    1
## 1977  554

#Comparing what the test_AKI responses with the actual
actual_AKI=test_AKI$AKI
tablePred_AKI=table(AKI.predict,actual_AKI)

tablePred_AKI

##             actual_AKI
## AKI.predict    0    1
##           0 1973    4
##           1   22  532
```

Considering p=0.5, we see that, 1973 “0” were predicted correctly, whereas 4 were wrong. In terms of “1”, 532 were predicted correctly, whereas 22 were wrong.

```
# Misclassification Ratio
mis_ratio = (tablePred_AKI[1, 2]+tablePred_AKI[2, 1])/(nrow(test_AKI))
mis_ratio

## [1] 0.01027262
```

The misclassification ratio is 0.01027262 (1.03%), in other words, 98.97% were predicted correctly in the test_AKI part. This was obtained considering p=0.5.

Visualizing the probability plots of AKI for the main variables (GLUCOSE and CREATININE) for different DM assuming all the others as mean/median values.

```
#Calculating Means of Relevant Explanatory variables
medianTrainAGE = median(train_AKI$AGE)
medianTrainAGE

## [1] 62

meanTrainGLUCOSE = mean(train_AKI$GLUCOSE)
meanTrainGLUCOSE

## [1] 163.986
```

```r
meanTrainCREATININE = mean(train_AKI$CREATININE)
meanTrainCREATININE
```

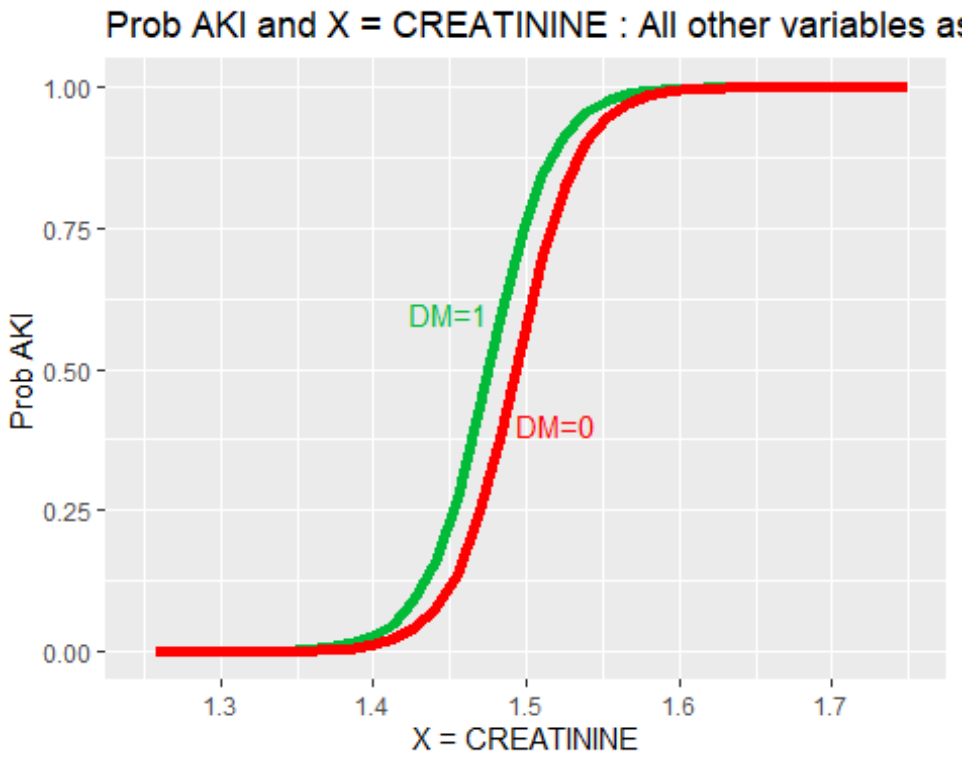
```
## [1] 1.303042
```

```r
medianTrainDM = median(as.integer(train_AKI$DM))
medianTrainDM
```

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

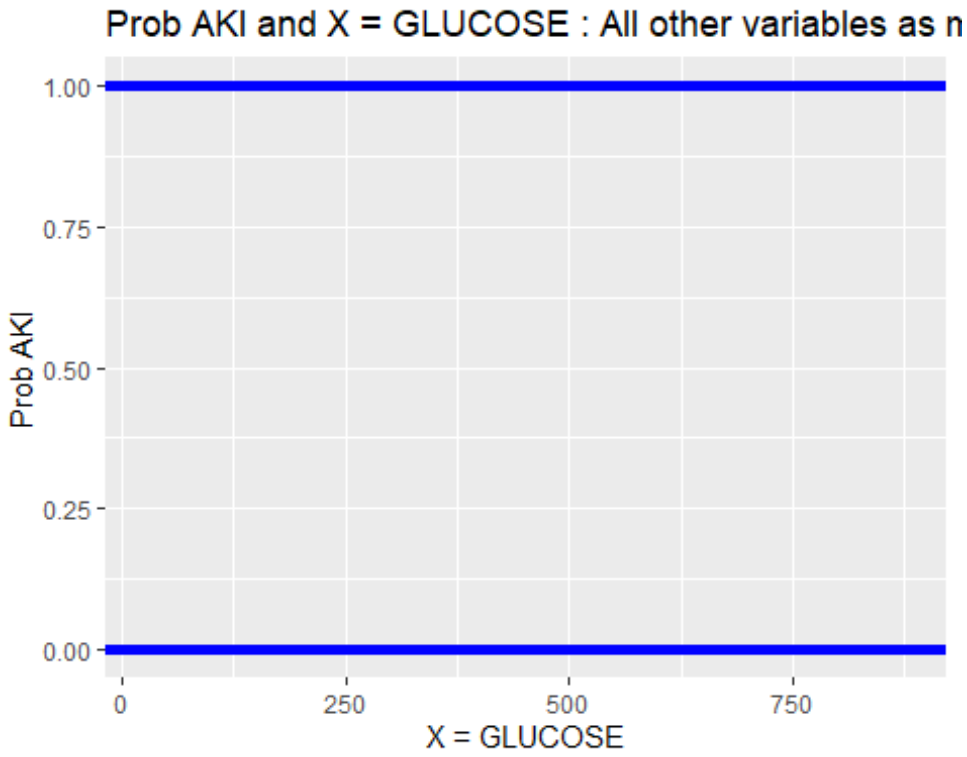
```r
#Calculating probability of AKI for each DM in function of UREAM and k
eeping the other parameters as median or mean

##### 1 - CREATININE
n = 1000
pXvectorM=rep(0,n)
pXvectorF=rep(0,n)
valueV=rep(0,n)

count = 0
minV = min(train_AKI$CREATININE)
maxV = max(train_AKI$CREATININE)


for(i in seq(1:n+1)){
  valueV[i] = minV+(maxV-minV)*count/n

  # DM = 1
  dmV = 1
  expV = exp(-71.102440-0.003436*meanTrainGLUCOSE+48.002438*valueV[i]+
0.856600*dmV)

  pXvectorM[i]=expV/(1+expV)

  # DM = 0
  dmV = 0
  expV = exp(-71.102440-0.003436*meanTrainGLUCOSE+48.002438*valueV[i]+
0.856600*dmV)

  pXvectorF[i]=expV/(1+expV)


  count = count + 1
}

#pXvector
```

```
dfPlot <-data.frame(valueV,pXvectorM, pXvectorF)

#Generating UREA Probability plot
ggplot(data=dfPlot, aes(x=valueV) ) +
  geom_line(aes(y=pXvectorM),colour="#00BA38", size = 2)+
  geom_line(aes(y=pXvectorF),colour="red", size = 2)+
  annotate("text", x= 1.52, y=0.4, label = "DM=0", color="Red")+
  annotate("text", x= 1.45, y=0.6, label = "DM=1", color="#00BA38")+
  #geom_hline(yintercept=0.0, color="blue", size=2)+
  #geom_hline(yintercept=1, color="blue", size=2)+
  xlim(1.25,1.75)+
  ggtitle("Prob AKI and X = CREATININE : All other variables as means"
)+
  labs(x = "X = CREATININE", y = "Prob AKI")
```




```
##### 2 - GLUCOSE

n = 100
pXvectorM=rep(0,n)
pXvectorF=rep(0,n)
valueV=rep(0,n)

count = 0
minV = min(train_AKI$GLUCOSE)
maxV = max(train_AKI$GLUCOSE)
```

```
for(i in seq(1:n+1)){
  valueV[i] = minV+(maxV-minV)*count/n

   # DM = 1
  dmV = 1
  expV = exp(-71.102440-0.003436*valueV[i]+48.002438*meanTrainCREATINI
NE+0.856600*dmV)

  pXvectorM[i]=expV/(1+expV)

  # DM = 0
  dmV = 0
  expV = exp(-71.102440-0.003436*valueV[i]+48.002438*meanTrainCREATINI
NE+0.856600*dmV)

  pXvectorF[i]=expV/(1+expV)


  count = count + 1
}

#pXvector


dfPlot <-data.frame(valueV,pXvectorM, pXvectorF)

#Generating UREA Probability plot
ggplot(data=dfPlot, aes(x=valueV) ) +
  geom_line(aes(y=pXvectorM),colour="#00BA38", size = 2)+
  geom_line(aes(y=pXvectorF),colour="red", size = 2)+
  #annotate("text", x= 40, y=0.7, label = "FEMALE", color="Red")+
  #annotate("text", x= 30, y=0.5, label = "MALE", color="#00BA38")+
  geom_hline(yintercept=0.0, color="blue", size=2)+
  geom_hline(yintercept=1, color="blue", size=2)+
  ggtitle("Prob AKI and X = GLUCOSE : All other variables as means")+
  labs(x = "X = GLUCOSE", y = "Prob AKI")
```



Based on the first plot in terms of CREATININE, the probability of having AKI is slightly different between having or no DM.No probability is expected for values of CREATININE lower than approximatelly 1.35, whereas 100% of probability of AKI for values approximately higher than 1.6. Between these values the probability follow this curve. The value obtaining here were assumed the mean of CREATININE in train part.

For the second plot in terms of GLUCOSE, the probability of AKI is zero considering the mean CREATININE value in train part.

#(B.2) LINEAR DISCIMINATION ANALYSIS (LDA)

To build the LDA model, we need to fulfil the assumptions that is variables need to be normally distributed. Therefore, we checked the normality of the quantitative variables in Heart Failure section and found all of them are not normally distributed as p value is less than 0.05 which reject null hypothesis. So, we exclude all quantitative variables.

As among our significant variables, both creatinine and glucose quantitative variables are not normally distributed, we cannot make the LDA model with only one categorical variable DM.

Therefore, It is not possible to build LDA model with only DM categorical significant variable and thus LDA model is not valid for this case.

However, we tried to build LDA model with suggested models with the categorical variables indicated by the independence test presented in item contingent table. So, we considered one of the suggested models categorical variables STEMI and DM to build the LDA model and do further analysis to see the result of misclassification rate. we tried another suggested model

with variables RAISED.CARDIAC.ENZYMES and PRIOR.CMP but we found misclassification rate is higher than this.

```
names (train_AKI)
```

```
##  [1] "MRD.No."                "AGE"                    "GENDER"
##  [4] "RURAL"                  "DURATION.OF.STAY"       "OUTCOME"
##  [7] "SMOKING"                "ALCOHOL"                "DM"
## [10] "HTN"                    "CAD"                    "PRIOR.CMP"
## [13] "CKD"                    "HB"                     "TLC"
## [16] "PLATELETS"              "GLUCOSE"                "UREA"
## [19] "CREATININE"             "RAISED.CARDIAC.ENZYMES" "EF"
## [22] "SEVERE.ANAEMIA"         "ANAEMIA"                "STABLE.ANGI
NA"
## [25] "ACS"                    "STEMI"                  "HEART.FAILU
RE"
## [28] "AKI"
```

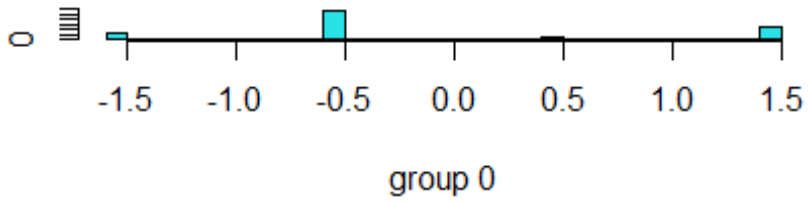
```
library(MASS)

#Creating the LDA model based on training part
# AKI_lda.fit<-lda(AKI~factor(GENDER)+AGE+GLUCOSE+HB+TLC+CREATININE+UR
EA+EF+
#                    factor(STEMI)+factor(DM), data = train_AKI)

AKI_lda.fit<-lda(AKI~
                   factor(STEMI)+factor(DM), data = train_AKI)
AKI_lda.fit
```

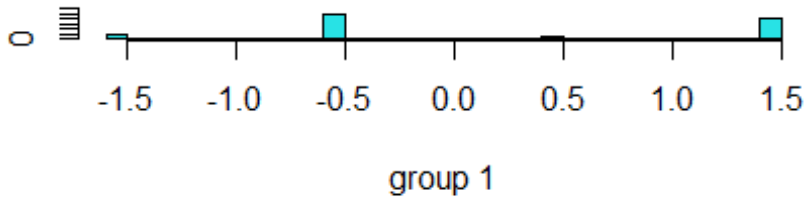
```
## Call:
## lda(AKI ~ factor(STEMI) + factor(DM), data = train_AKI)
##
## Prior probabilities of groups:
##         0         1
## 0.7882539 0.2117461
##
## Group means:
##   factor(STEMI)1 factor(DM)1
## 0      0.1772469   0.2943535
## 1      0.1293532   0.4465174
##
## Coefficients of linear discriminants:
##                       LD1
## factor(STEMI)1 -0.983984
## factor(DM)1     2.002377
```

The LDA output indicates that our prior probabilities are 0.7882539 and 0.2117461 or in other words, 78.8% of the training observations are patients who are not having ACUTE KIDNEY INJURY and 21.2 % represent those that are having ACUTE KIDNEY INJURY. It also provides the group means.

The The prior probabilities are as expected when it was created the train_AKI part.

```
#Plotting visualize how are the distribution of "Y" and "N"
plot(AKI_lda.fit)
```






```
#Checking the prediction based on test_AKI part
AKI.pred<-predict(AKI_lda.fit,test_AKI)

#diabetes.pred
names(AKI.pred)

## [1] "class"     "posterior" "x"

#Plotting pairwise plot of the training set
#pairs(train_AKI)

# Checking the confusion table (Predicted versus real value in test_AK
I part)
tablePred_AKI=table(AKI.pred$class, test_AKI$AKI)

tablePred_AKI
```

```
##
##        0    1
##   0 1995  536
##   1    0    0
```

1989 “N” were predicted correctly, whereas 218 were wrong. In terms of “Y”, 318 were predicted correctly, whereas 6 were wrong. This was obtained assuming P(Y=pos|X1,…,Xp)≥0.5.

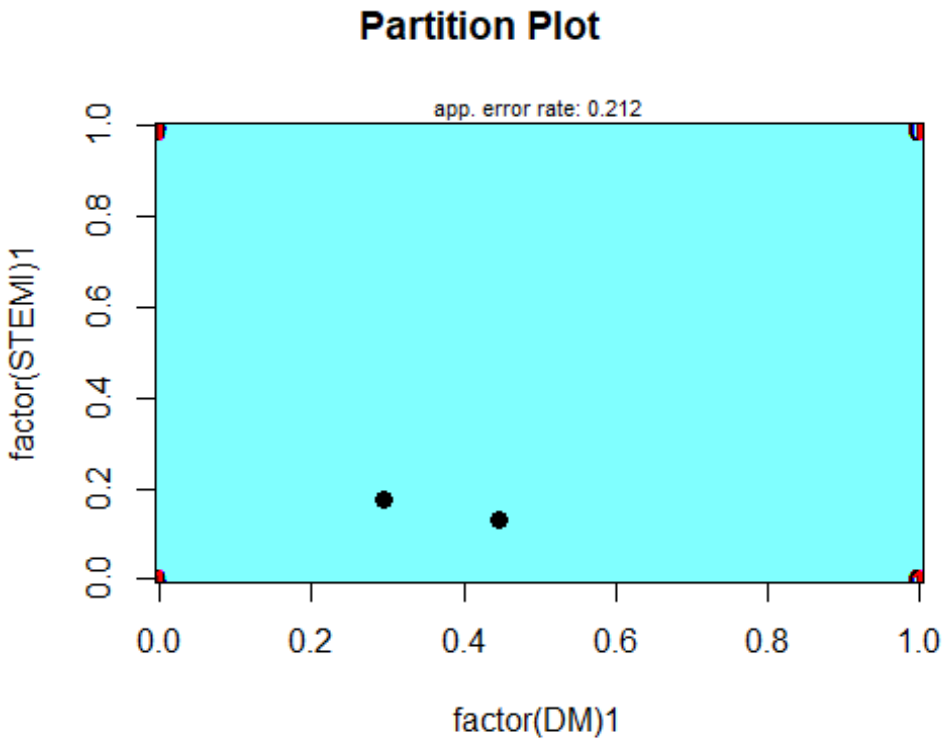
```
# Misclassification Ratio
mis_ratio = (tablePred_AKI[1, 2]+tablePred_AKI[2, 1])/(nrow(test_AKI))
mis_ratio
```

```
## [1] 0.211774
```

The misclassification ratio is 0.08850257 (8.9%), in other words, 91.1% were predicted correctly the test_AKI part. This was obtained assuming P(Y=pos|X1,…,Xp)≥0.5.

Checking the error of some pair parameters.

```
library(klaR)

# partimat(AKI~factor(GENDER)+AGE+GLUCOSE+HB+CREATININE+factor(STEMI)+
factor(DM),
#          data=train_AKI, method="lda")
partimat(AKI~factor(STEMI)+factor(DM),
         data=train_AKI, method="lda")
```



```
# graphics.off()
#  par("mar")
#  par(mar=c(1,1,1,1))
```

from the partiion plot, we see that DM and STEMI is showing error rate 0.212 which is similar to misclassification rate.

#(B.3) QUADRATIC DISCRIMINATION ANALYSIS

We have built QDA model with and without 10 k-fold cross validation (CV) to see the results. QDA analysis done considering significant variables only that is Creatinine, Glucose, and DM.

```
# Creating the QDA model based on train_AKI part
AKI_qda.fit<-qda(AKI~CREATININE+GLUCOSE+factor(DM), data = train_AKI)
AKI_qda.fit

## Call:
## qda(AKI ~ CREATININE + GLUCOSE + factor(DM), data = train_AKI)
##
## Prior probabilities of groups:
##         0         1
## 0.7882539 0.2117461
##
## Group means:
##   CREATININE  GLUCOSE factor(DM)1
## 0   0.893588 159.5743   0.2943535
## 1   2.827289 180.4093   0.4465174
```

For the model built without CV, we found the QDA model output indicates that 78.8% of the training observations are patients who are not having ACUTE KIDNEY INJURY and 21.2 % represent those that are having ACUTE KIDNEY INJURY. It also provided the group means of each variable.

```
# Applying the model to the test_AKI data to predict the response vari
able
AKI.pred<-predict(AKI_qda.fit, test_AKI)$class

tablePred_AKI=table(AKI.pred, test_AKI$AKI)

tablePred_AKI

##
## AKI.pred    0    1
##        0 1987   39
##        1    8  497
```

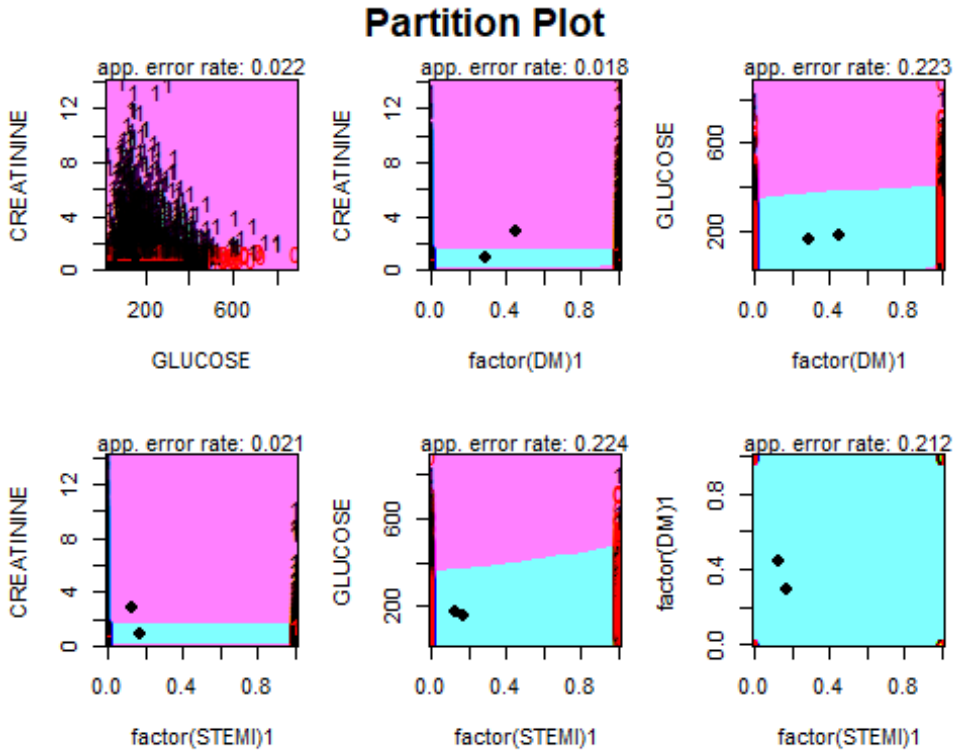
1987 “N” were predicted correctly, whereas 39 were wrong. In terms of “Y”,497 were predicted correctly, whereas 8 were wrong. This was obtained assuming $P(Y=pos|X1,\ldots,Xp)\geq 0.5$.

```
# Misclassification Ratio
mis_ratio = (tablePred_AKI[1, 2]+tablePred_AKI[2, 1])/(nrow(test_AKI))
mis_ratio

## [1] 0.01856974
```

The misclassification ratio is 0.01856974 (1.86%), in other words, 98.14% were predicted correctly the test_AKI part. This was obtained assuming P(Y=pos|X1,…,Xp)≥0.5.

We have drawn the partition plot to identify the lowest error rate for the associated variables. Here we included STEMI category variable along with significant ones due to LDA model was built considering STEMI and to see the error rate with others in QDA partition plot.

```
library(klaR)

partimat(AKI~CREATININE++GLUCOSE+factor(DM)+factor(STEMI),
         data=train_AKI, method="qda")
```



From the partition plot, we see that Creatinine and DM is showing lowest error rate that is 0.018.

We tried redoing QDA model with only these two variables to see the result and we found the misclassification rate is 0.01738443 (1.72%) which is slightly improved from actual QDA model rate 1.86%.

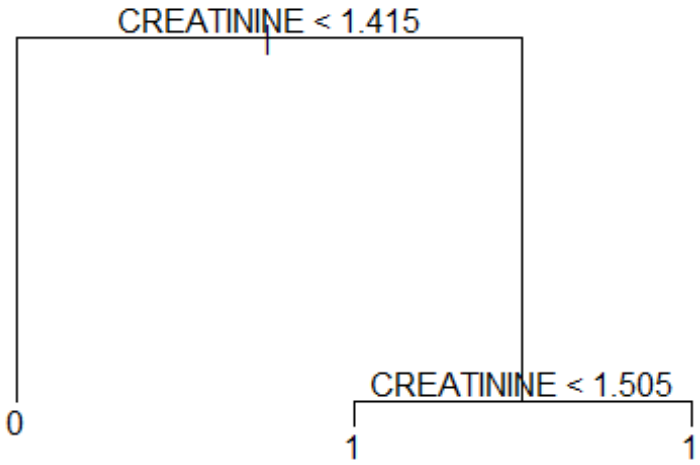
#(B.4) CLASSIFICATION TREE

```
library(tree)

# Doing the Classification Tree
AKI_tree.fit<-tree(AKI~CREATININE+GLUCOSE+factor(DM), train_AKI)


summary(AKI_tree.fit)

##
## Classification tree:
## tree(formula = AKI ~ CREATININE + GLUCOSE + factor(DM), data = trai
n_AKI)
## Variables actually used in tree construction:
## [1] "CREATININE"
## Number of terminal nodes:  3
## Residual mean deviance:  0.03923 = 297.8 / 7591
## Misclassification error rate: 0.0129 = 98 / 7594
```

From the model, we found the classification model only selected creatinine variable to construct the tree and the total number of terminal nodes selected by model is 3.

Further we plot the tree for better visualization and due to only 3 terminal nodes, we did not prune the tree as this showing the optimal view/result for the prediction.

```
#Plotting the tree
plot(AKI_tree.fit)
text(AKI_tree.fit ,pretty =0)
```



The tree only has 3 terminal nodes, with only 1 class (CREATININE) as splitting rules. In addition, it is possible to see on the right side (CREATININE < 1.505), if an observation falls there, it will always indicate Acute Kidney Injury.

Let us check the probability in each terminal node.

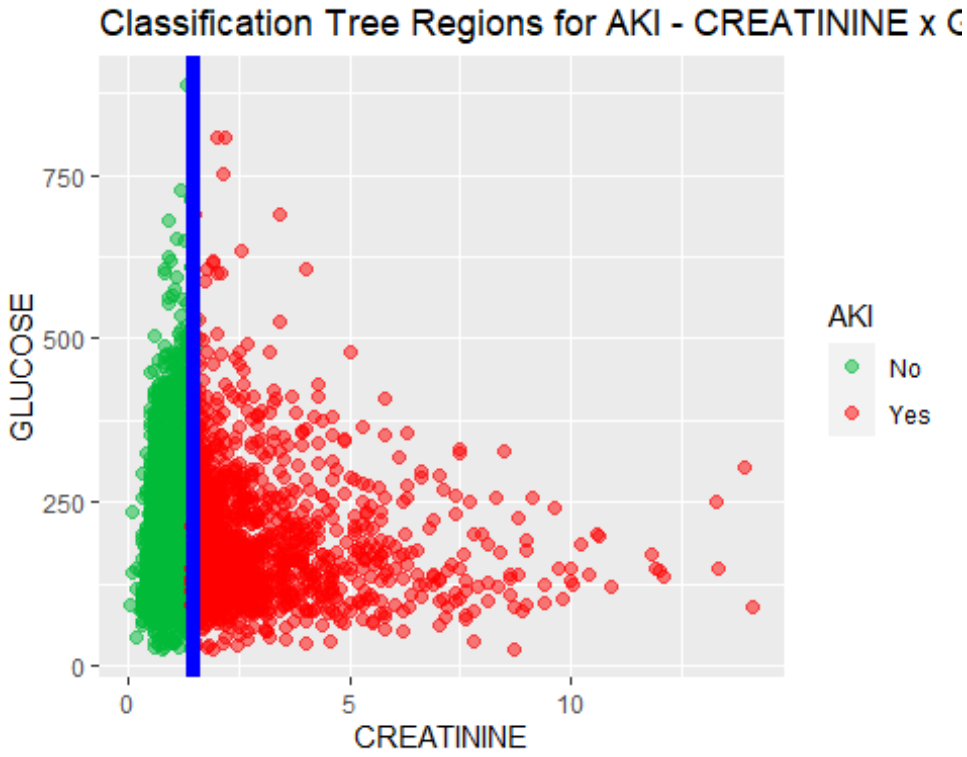
```
# Check the nodes of the tree
AKI_tree.fit
```

```
## node), split, n, deviance, yval, (yprob)
##       * denotes terminal node
##
## 1) root 7594 7841.00 0 ( 0.7882539 0.2117461 )
##   2) CREATININE < 1.415 5890   19.36 0 ( 0.9998302 0.0001698 ) *
##   3) CREATININE > 1.415 1704  744.40 1 ( 0.0569249 0.9430751 )
##     6) CREATININE < 1.505 201  278.40 1 ( 0.4825871 0.5174129 ) *
##     7) CREATININE > 1.505 1503    0.00 1 ( 0.0000000 1.0000000 ) *
```

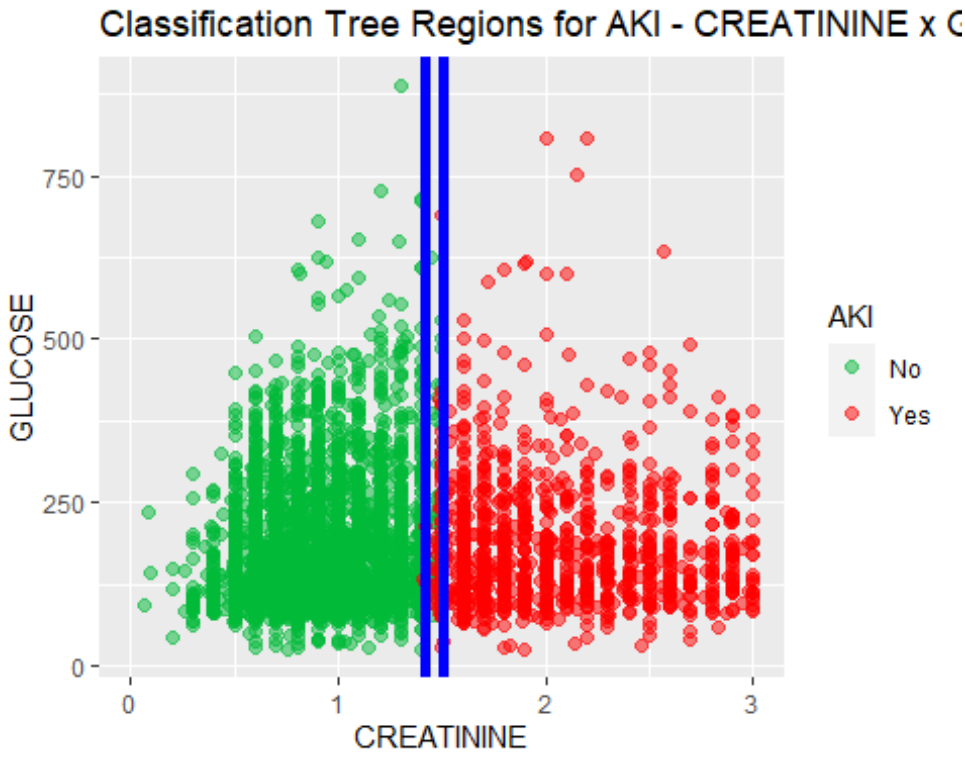
Here we see that, in case of CREATININE > 1.505 where a purity has reached to 100% of ”1" as prediction whereas CREATININE < 1.505 showing almost similar probability 48% and 51% for both ”0” and ”1, Consequently it is possible the majority of wrong predictions happen there. For the left side of tree where creatinine <1.4, the difference of probability is also significant, the “0” probability is showing 99.98% so probability of ”0" is very high.

```
#Generating plots with regions
ggplot(data=train_AKI, aes(x=CREATININE, y=GLUCOSE, color=AKI))+
```

```
  geom_point(size=2, alpha=0.5)+
  scale_color_manual(labels=c("No", "Yes"), values = c("#00BA38","red"
))+
  geom_vline(xintercept=1.505, color="blue", size=2)+
  geom_vline(xintercept=1.415, color="blue", size=2)+
  ggtitle("Classification Tree Regions for AKI - CREATININE x GLUCOSE"
)
```




```
ggplot(data=train_AKI, aes(x=CREATININE, y=GLUCOSE, color=AKI))+
  geom_point(size=2, alpha=0.5)+
  scale_color_manual(labels=c("No", "Yes"), values = c("#00BA38","red"
))+
  geom_vline(xintercept=1.505, color="blue", size=2)+
  geom_vline(xintercept=1.415, color="blue", size=2)+
  xlim(0,3)+
  ggtitle("Classification Tree Regions for AKI - CREATININE x GLUCOSE"
)
```



For further visualization, we have drawn the classification tree regions for Creatinine vs Glucose along with AKI for better visualization.

From this plot, we see that, AKI "1"-Yes and "0"-No is well divided into two regions. Both yes and no of AKI data is merged within blue line where creatinine value is around 1.5 where most of the wrong prediction were happened.

Apply the tree to the test_AKI set, calculate the root square of the mean squared error (RMSE)

```
# Set a seed for random number generation
set.seed(10)

# Applying the unpruned tree to test_AKI part
AKI_tree.pred<-predict(AKI_tree.fit,test_AKI,type = "class")
tablePred_AKI=table(AKI_tree.pred,test_AKI$AKI)

tablePred_AKI

##
## AKI_tree.pred    0    1
##             0 1962    0
##             1   33  536
```

1962 "N" were predicted correctly, whereas 0 were wrong. In terms of "Y", 536 were predicted correctly, whereas 33 were wrong.

```
# Misclassification Ratio
mis_ratio = (tablePred_AKI[1, 2]+tablePred_AKI[2, 1])/(nrow(test_AKI))
mis_ratio

## [1] 0.01303832
```

The misclassification ratio is 0.01303832 (1.3%), in other words, 98.7% were predicted correctly the test_AKI part.

It was not necessary to prune the tree seeing that just few nodes were provided.

#(B.5) LOGISTIC REGRESSION with stratified 10-fold cross-validation

Randomly dividing the total clean observations in folds with approximately equal size as well as proportion of AKI of “Y” and “N”.

```
library(caret)

# Set a seed for random number generation
set.seed(10)

# Creating 10 folds
folds_AKI<-createFolds(as.factor(dataClean$AKI), k=10)

#Checking the units in each fold
fold01_AKI<-dataClean[folds_AKI$Fold01,]
fold02_AKI<-dataClean[folds_AKI$Fold02,]
fold03_AKI<-dataClean[folds_AKI$Fold03,]
fold04_AKI<-dataClean[folds_AKI$Fold04,]
fold05_AKI<-dataClean[folds_AKI$Fold05,]
fold06_AKI<-dataClean[folds_AKI$Fold06,]
fold07_AKI<-dataClean[folds_AKI$Fold07,]
fold08_AKI<-dataClean[folds_AKI$Fold08,]
fold09_AKI<-dataClean[folds_AKI$Fold09,]
fold10_AKI<-dataClean[folds_AKI$Fold10,]
table(fold01_AKI$AKI)

##
##   0   1
## 798 215

table(fold02_AKI$AKI)

##
##   0   1
## 798 214

table(fold03_AKI$AKI)
```

```
##
##   0   1
## 798 215

table(fold04_AKI$AKI)

##
##   0   1
## 798 214

table(fold05_AKI$AKI)

##
##   0   1
## 798 214

table(fold06_AKI$AKI)

##
##   0   1
## 798 214

table(fold07_AKI$AKI)

##
##   0   1
## 798 214

table(fold08_AKI$AKI)

##
##   0   1
## 798 215

table(fold09_AKI$AKI)

##
##   0   1
## 799 214

table(fold10_AKI$AKI)

##
##   0   1
## 798 215

#folds_AKI
```

It is possible to verify that AKI responses were approximately equally distributed.

```r
# Creating function to calculate misclassification rate for Logistic R
egression
library(MASS)

misclassification_LOGISTIC_AKI<-function(idx_AKI){
  # Select the other folders to train_AKIing part
  trainFold_AKI<-dataClean[-idx_AKI,]

  # The current fold as the validation (test_AKI) part
  validationFold_AKI<-dataClean[idx_AKI,]

  #Fit the Logistic Regression model for the train_AKIing Fold part
  fit_LOGISTICmodel_AKI<-glm(AKI~CREATININE+GLUCOSE+factor(DM), family
=binomial,
                             data=trainFold_AKI)


  # Applying the validation part to the fitted model and predict the H
EART FAILURE
  pred_AKI<-predict(fit_LOGISTICmodel_AKI,validationFold_AKI)

  AKI.predict=rep("0", nrow(validationFold_AKI))

  AKI.predict[pred_AKI >= 0.5]="1"


  #return the mean error of the prediction for the idx_AKI fold
  return(1-mean(AKI.predict==validationFold_AKI$AKI))
}

# Set a seed for random number generation
set.seed(10)

# calculating the misclassification rate of each fold - Logistic Regre
ssion
mis_rate_Logistic_AKI=lapply(folds_AKI, misclassification_LOGISTIC_AKI
)

# calculating the misclassification rate for each fold
#mis_rate_LOGISTIC_AKI

# Calculating the average misclassification rate
mean(as.numeric(mis_rate_Logistic_AKI))

## [1] 0.01106144
```

For Logistic Regression stratified with 10-folds, the misclassification rate is 0.0106144 (1.11%).

#(B.6) LDA with stratified 10-fold cross-validation We are not able to consider any quantitative variable as it was verified previously that they are not normally distributed.

As mentioned earlier, among our significant variables, both creatinine and glucose quantitative variables are not normally distributed, we cannot make the LDA model with only one categorical variable DM.

Therefore, It is not possible to build LDA model with only DM categorical significant variable and thus LDA model is not valid for this case.

However, we tried here to build LDA model with suggested models with the categorical variables indicated by the independence test presented in item contingent table. So, we considered one of the suggested models categorical variables STEMI and DM to build the LDA model and do further analysis to see the result of misclassification rate.

```
# Creating function to calculate misclassification rate for LDA
library(MASS)

misclassification_LDA_AKI<-function(idx_AKI){
  # Select the other folders to train_AKIing part
  trainFold_AKI<-dataClean[-idx_AKI,]

  # The current fold as the validation (test_AKI) part
  validationFold_AKI<-dataClean[idx_AKI,]

  #Fit the LDA model for the train_AKIing Fold part
  #fit_LDAmodel_AKI<-lda(AKI~CREATININE+GLUCOSE+factor(DM), data=train
Fold_AKI)
  fit_LDAmodel_AKI<-lda(AKI~factor(STEMI)+factor(DM), data=trainFold_A
KI)

  # Applying the validation part to the fitted model and predict the T
ype
  pred_AKI<-predict(fit_LDAmodel_AKI,validationFold_AKI)

  #return the mean error of the prediction for the idx_AKI fold
  return(1-mean(pred_AKI$class==validationFold_AKI$AKI))
}

# Set a seed for random number generation
set.seed(10)

# calculating the misclassification rate of each fold - LDA
mis_rate_LDA_AKI=lapply(folds_AKI, misclassification_LDA_AKI)

# calculating the misclassification rate for each fold
#mis_rate_LDA_AKI
```

```r
# Calculating the average misclassification rate
mean(as.numeric(mis_rate_LDA_AKI))
```

```
## [1] 0.2117529
```

For LDA stratified with 10-folds, the misclassification rate is 0.2117529 (21.2%). To compare, we also built model with 10 k-fold CV and analysis the outcome where we found the misclassification rate is same.

#(B.7) QDA with stratified 10-fold cross-validation

```r
# Creating function to calculate misclassification rate for QDA
library(MASS)

misclassification_LDA_AKI<-function(idx_AKI){
  # Select the other folders to train_AKIing part
  trainFold_AKI<-dataClean[-idx_AKI,]

  # The current fold as the validation (test_AKI) part
  validationFold_AKI<-dataClean[idx_AKI,]

  #Fit the QDA model for the train_AKIing Fold part
  fit_QDAmodel_AKI<-qda(AKI~CREATININE+GLUCOSE+factor(DM), data=trainF
old_AKI)

  # Applying the validation part to the fitted model and predict the T
ype
  pred_AKI<-predict(fit_QDAmodel_AKI,validationFold_AKI)

  #return the mean error of the prediction for the idx_AKI fold
  return(1-mean(pred_AKI$class==validationFold_AKI$AKI))
}

# Set a seed for random number generation
set.seed(10)

# calculating the misclassification rate of each fold - QDA
mis_rate_QDA_AKI=lapply(folds_AKI, misclassification_LDA_AKI)

# calculating the misclassification rate for each fold
#mis_rate_QDA_AKI

# Calculating the average misclassification rate
mean(as.numeric(mis_rate_QDA_AKI))
```

```
## [1] 0.02340727
```

For QDA stratified with 10-folds, the misclassification rate is 0.02340727 (2.34%).

#(A.8) Classification Tree with stratified 10-fold cross-validation

```
# Creating function to calculate misclassification rate for Classifica
tion Tree
library(tree)

misclassification_TREE_AKI<-function(idx_AKI){
  # Select the other folders to train_AKIing part
  trainFold_AKI<-dataClean[-idx_AKI,]

  # The current fold as the validation (test_AKI) part
  validationFold_AKI<-dataClean[idx_AKI,]

  #Fit the Tree model for the train_AKIing Fold part
  fit_TREEmodel_AKI<-tree(AKI~CREATININE+GLUCOSE+factor(DM), data=trai
nFold_AKI)

  # Applying the validation part to the fitted model and predict the T
ype
  pred_AKI<-predict(fit_TREEmodel_AKI,validationFold_AKI, type = "clas
s")


  #return the mean error of the prediction for the idx_AKI fold
  return(1-mean(pred_AKI==validationFold_AKI$AKI))
}

# Set a seed for random number generation
set.seed(10)

# calculating the misclassification rate of each fold - Classification
Tree
mis_rate_ClassificationTREE_AKI=lapply(folds_AKI, misclassification_TR
EE_AKI)

# calculating the misclassification rate for each fold
#mis_rate_ClassificationTREE_AKI

# Calculating the average misclassification rate
mean(as.numeric(mis_rate_ClassificationTREE_AKI))

## [1] 0.01333358
```

For Classification tree considering stratified with 10-folds, the misclassification rate is 0.01333358 (1.33%).

#(B.9) SUMMARY oF AKI MODELS

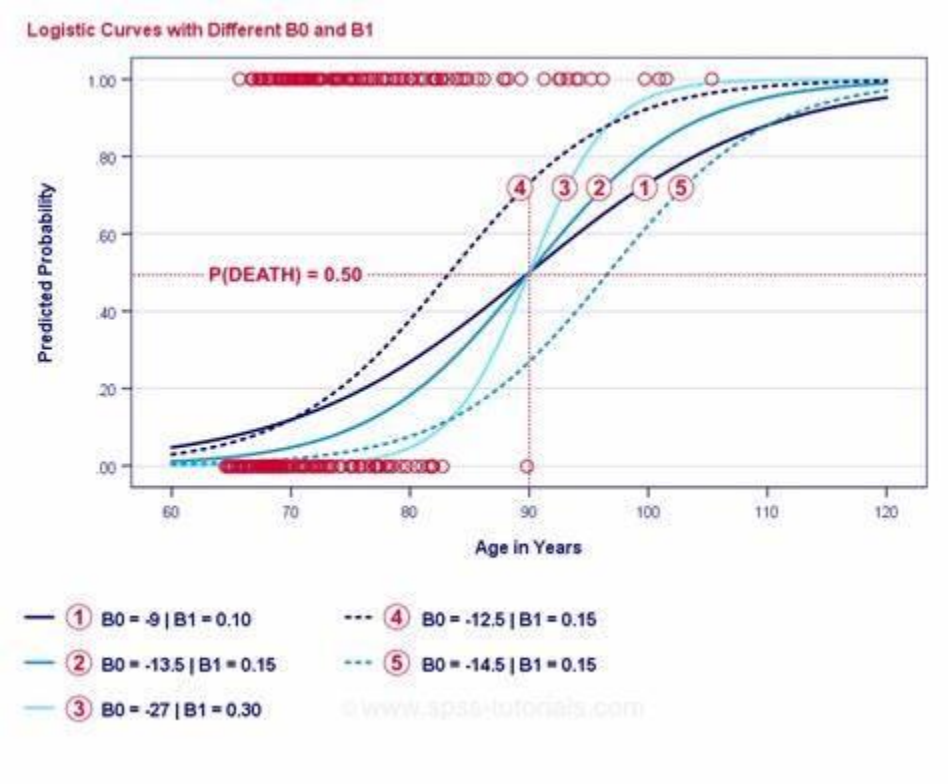
The summary results of the different models created to predict AKI not only using stratified sampling (75% train part, 25% test part), but also with 10 k-fold Stratified Cross-validation, which the results in term of misclassification rate are presented below-

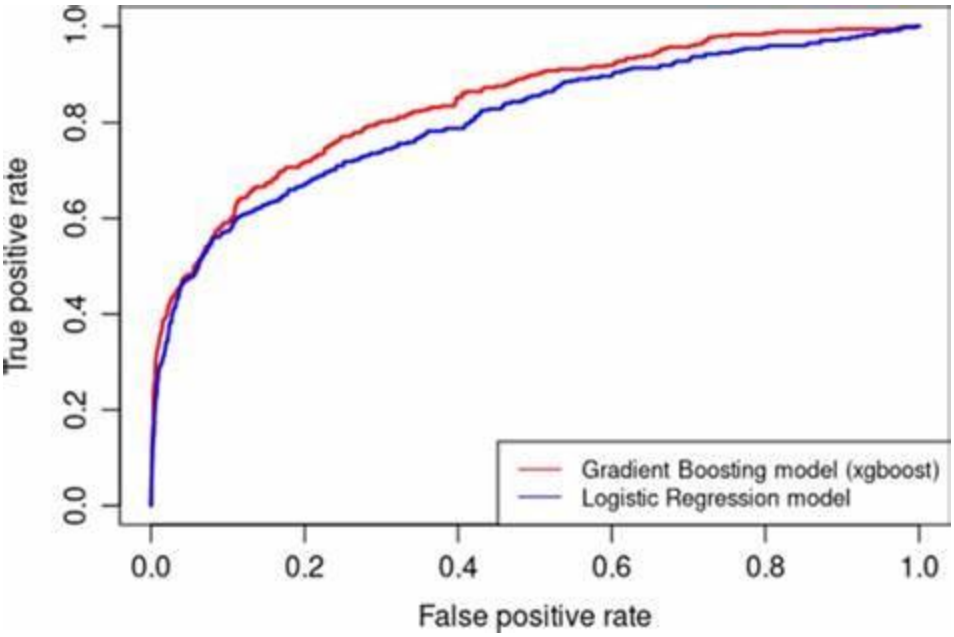
1) 75% train_AKI part and 25% Prediction Part (no k-Fold Cross-Validation) LOGISTIC REGRESSION: Misclassification Rate = 1.03% #LDA: Misclassification Rate = “It is not possible to use this with significant variables and at the same time because all the quantitative variables are not normally distributed” QDA: Misclassification Rate = 1.86% Classifiation Tree: Misclassification Rate = 1.3%

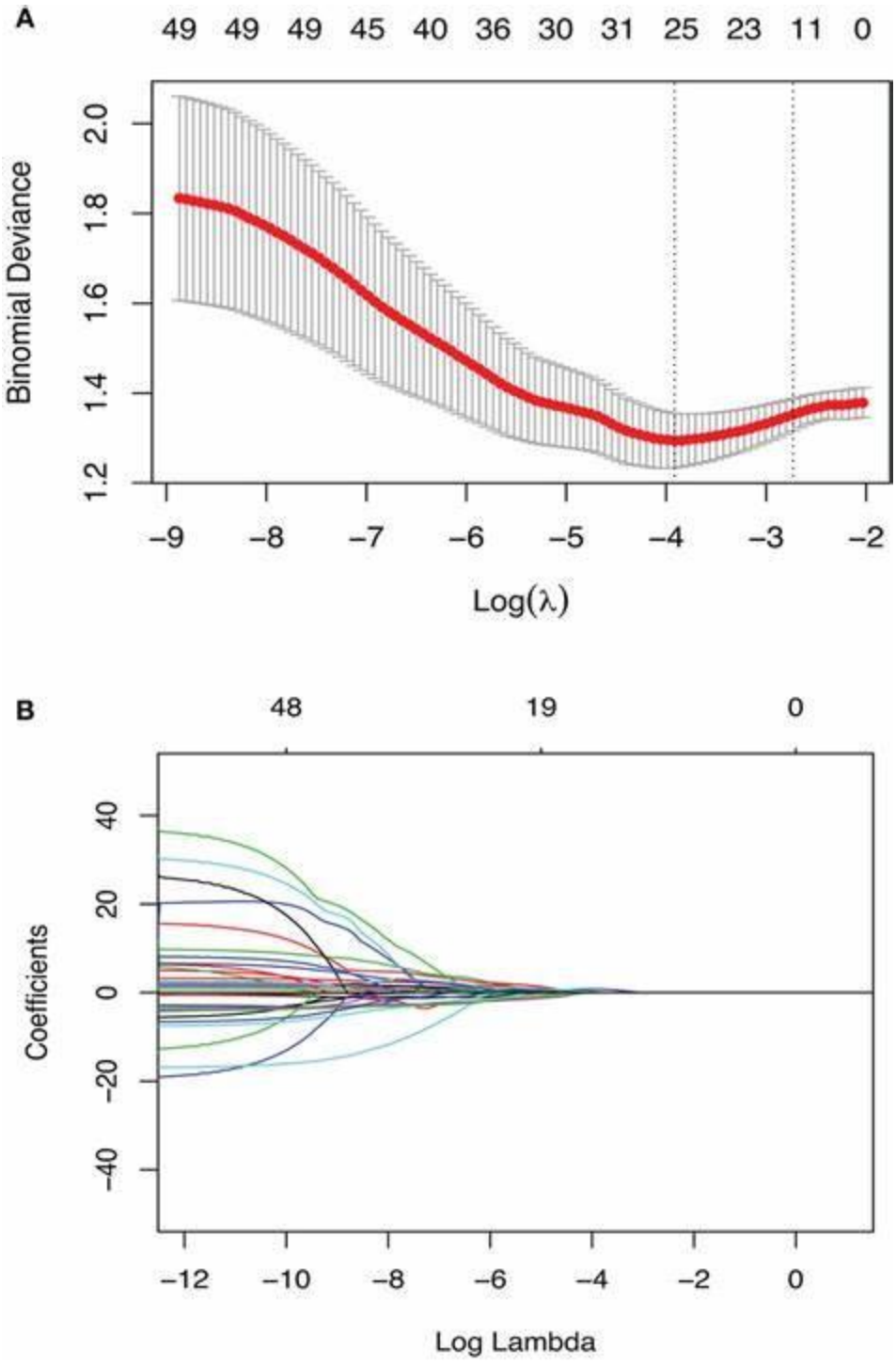
2) 10–Fold Cross-Validation LOGISTIC REGRESSION: Misclassification Rate = 1.11% #LDA: Misclassification Rate = “It is not possible to use this with significant variables and at the same time because all the quantitative variables are not normally distributed” QDA: Misclassification Rate = 2.34% Classifiation Tree: Misclassification Rate = 1.33%

NB: However, we attempted to build LDA model with suggested models with the categorical variables (STEMI and DM) as indicated by the independence test presented in contingent table section to see the difference and results and we found misclassification rate is 21.2% with and without CV cases which is higher than other models and does not predict AKI positive effectively.

Based on the results above,







the Logistic Regression model without stratified sampling indicated the lowest or best misclassification rate of 1.03% among all the generated models to predict AKI. However, we did not observe significant differences with logistic regression with 10 k-fold stratified cross validation and classification tree with and without cross validation.

```
head(dataClean)
```

```
##   MRD.No. AGE GENDER RURAL DURATION.OF.STAY   OUTCOME SMOKING ALCOH
OL DM HTN
## 1  234735  81      M     R                3 DISCHARGE       0
0  1   0
## 2  234696  65      M     R                5 DISCHARGE       0
1  0   1
## 4  234635  67      F     U                8 DISCHARGE       0
0  0   1
## 5  234486  60      F     U               23 DISCHARGE       0
0  0   1
## 6  234675  44      M     U               10 DISCHARGE       0
0  1   1
## 7  234563  56      F     U                6 DISCHARGE       0
0  1   1
##   CAD PRIOR.CMP CKD   HB  TLC PLATELETS GLUCOSE UREA CREATININE
## 1   0         0   0  9.5 16.1       337      80   34       0.90
## 2   1         0   0 13.7  9.0       149     112   18       0.90
## 4   1         0   0 12.8  9.9       286     130   27       0.60
## 5   0         1   0 13.6  9.1        26     144   55       1.25
## 6   1         1   0 13.5 22.3       322     217   51       0.90
## 7   1         1   0 13.3 12.6       166     277   28       0.60
##   RAISED.CARDIAC.ENZYMES EF SEVERE.ANAEMIA ANAEMIA STABLE.ANGINA AC
S STEMI
## 1                      1 35              0       1             0
1     0
## 2                      0 42              0       0             0
0     0
## 4                      0 42              0       0             0
0     0
## 5                      0 16              0       0             0
0     0
## 6                      0 25              0       0             0
1     0
## 7                      0 30              0       0             0
1     1
##   HEART.FAILURE AKI
## 1             1   0
## 2             0   0
## 4             0   0
## 5             0   0
## 6             1   0
## 7             1   0
```

IV.2 - Quantitative Response Variables (DURATION OF STAY IN HOSPITAL)

IV.2.1 - LINEAR REGRESSION IN PREDICTING DURATION OF STAY IN HOSPITAL

```
head(dataClean)
```

```
##   MRD.No. AGE GENDER RURAL DURATION.OF.STAY   OUTCOME SMOKING ALCOH
OL DM HTN
## 1  234735  81      M     R                3 DISCHARGE       0
0  1   0
## 2  234696  65      M     R                5 DISCHARGE       0
1  0   1
## 4  234635  67      F     U                8 DISCHARGE       0
0  0   1
## 5  234486  60      F     U               23 DISCHARGE       0
0  0   1
## 6  234675  44      M     U               10 DISCHARGE       0
0  1   1
## 7  234563  56      F     U                6 DISCHARGE       0
0  1   1
##   CAD PRIOR.CMP CKD   HB  TLC PLATELETS GLUCOSE UREA CREATININE
## 1   0         0   0  9.5 16.1       337      80   34       0.90
## 2   1         0   0 13.7  9.0       149     112   18       0.90
## 4   1         0   0 12.8  9.9       286     130   27       0.60
## 5   0         1   0 13.6  9.1        26     144   55       1.25
## 6   1         1   0 13.5 22.3       322     217   51       0.90
## 7   1         1   0 13.3 12.6       166     277   28       0.60
##   RAISED.CARDIAC.ENZYMES EF SEVERE.ANAEMIA ANAEMIA STABLE.ANGINA AC
S STEMI
## 1                      1 35              0       1             0
1     0
## 2                      0 42              0       0             0
0     0
## 4                      0 42              0       0             0
0     0
## 5                      0 16              0       0             0
0     0
## 6                      0 25              0       0             0
1     0
## 7                      0 30              0       0             0
1     1
##   HEART.FAILURE AKI
## 1             1   0
```

```
## 2             0   0
## 4             0   0
## 5             0   0
## 6             1   0
## 7             1   0
```

```
dim(dataClean)
```

```
## [1] 10125    28
```

```
summary(dataClean)
```

```
##    MRD.No.               AGE           GENDER             RURAL
##  Length:10125       Min.   :  4.0   Length:10125       Length:10125
##  Class :character   1st Qu.: 53.0   Class :character   Class :chara
cter
##  Mode  :character   Median : 62.0   Mode  :character   Mode  :chara
cter
##                     Mean   : 61.1
##                     3rd Qu.: 70.0
##                     Max.   :110.0
##  DURATION.OF.STAY      OUTCOME     SMOKING  ALCOHOL  DM       HTN
CAD
##  Min.   : 1.000   DAMA     : 446   0:9571   0:9390   0:6802   0:525
0   0:3282
##  1st Qu.: 3.000   DISCHARGE:9086   1: 554   1: 735   1:3323   1:487
5   1:6843
##  Median : 6.000   EXPIRY   : 593
##  Mean   : 6.593
##  3rd Qu.: 8.000
##  Max.   :98.000
##  PRIOR.CMP CKD            HB              TLC            PLATELETS
##  0:8544    0:9274   Min.   : 3.00   Min.   :  0.30   Min.   :   1.3
8
##  1:1581    1: 851   1st Qu.:10.80   1st Qu.:  8.00   1st Qu.: 173.0
0
##                     Median :12.50   Median : 10.20   Median : 226.0
0
##                     Mean   :12.33   Mean   : 11.62   Mean   : 238.9
3
##                     3rd Qu.:13.90   3rd Qu.: 13.60   3rd Qu.: 288.0
0
##                     Max.   :22.00   Max.   :261.00   Max.   :1111.0
0
##     GLUCOSE           UREA          CREATININE     RAISED.CARDIAC.E
NZYMES
##  Min.   :  1.2   Min.   :  0.10   Min.   : 0.065   0:7815
```

```
##  1st Qu.:106.0   1st Qu.: 25.00   1st Qu.: 0.760   1:2310
##  Median :137.0   Median : 35.00   Median : 0.960
##  Mean   :164.3   Mean   : 47.52   Mean   : 1.303
##  3rd Qu.:196.0   3rd Qu.: 55.00   3rd Qu.: 1.390
##  Max.   :888.0   Max.   :450.00   Max.   :15.630
##        EF        SEVERE.ANAEMIA    ANAEMIA  STABLE.ANGINA ACS
STEMI
##  Min.   :14.00   Min.   :0.00000   0:8433   0:9293        0:5995
0:8441
##  1st Qu.:32.00   1st Qu.:0.00000   1:1692   1: 832        1:4130
1:1684
##  Median :44.00   Median :0.00000
##  Mean   :44.06   Mean   :0.01719
##  3rd Qu.:60.00   3rd Qu.:0.00000
##  Max.   :60.00   Max.   :1.00000
##  HEART.FAILURE AKI
##  0:7217        0:7981
##  1:2908        1:2144
##
##
##
##
```

```
names(dataClean)
```

```
##  [1] "MRD.No."                "AGE"                    "GENDER"
##  [4] "RURAL"                  "DURATION.OF.STAY"       "OUTCOME"
##  [7] "SMOKING"                "ALCOHOL"                "DM"
## [10] "HTN"                    "CAD"                    "PRIOR.CMP"
## [13] "CKD"                    "HB"                     "TLC"
## [16] "PLATELETS"              "GLUCOSE"                "UREA"
## [19] "CREATININE"             "RAISED.CARDIAC.ENZYMES" "EF"
## [22] "SEVERE.ANAEMIA"         "ANAEMIA"                "STABLE.ANGI
NA"
## [25] "ACS"                    "STEMI"                  "HEART.FAILU
RE"
## [28] "AKI"
```

Creating the linear model using numerical variables and one categorical variable (Gender) to determine the coefficients and P-value in order to find out which ones are statistically relevant.

```
library(mctest)

linearModel1 <- lm(DURATION.OF.STAY~factor(GENDER)+AGE+HB+TLC+PLATELET
S+GLUCOSE+UREA+CREATININE, data = dataClean)

summary(linearModel1)
```

```
##
## Call:
## lm(formula = DURATION.OF.STAY ~ factor(GENDER) + AGE + HB + TLC +
##     PLATELETS + GLUCOSE + UREA + CREATININE, data = dataClean)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -15.280  -2.801  -0.844   1.529  89.521
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)      6.1511961  0.4271455  14.401  < 2e-16 ***
## factor(GENDER)M  0.2257423  0.1026968   2.198    0.028 *
## AGE              0.0164249  0.0036104   4.549 5.44e-06 ***
## HB              -0.2430356  0.0228688 -10.627  < 2e-16 ***
## TLC              0.0714295  0.0067349  10.606  < 2e-16 ***
## PLATELETS        0.0001566  0.0004690   0.334    0.738
## GLUCOSE          0.0037358  0.0005517   6.772 1.34e-11 ***
## UREA             0.0156683  0.0018748   8.357  < 2e-16 ***
## CREATININE       0.0510935  0.0604433   0.845    0.398
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.657 on 10116 degrees of freedom
## Multiple R-squared:  0.07289,    Adjusted R-squared:  0.07216
## F-statistic: 99.41 on 8 and 10116 DF,  p-value: < 2.2e-16
```

Removing PLATELETS and CREATININE from the model

```
linearModel2 <- lm(DURATION.OF.STAY~factor(GENDER)+AGE+HB+TLC+GLUCOSE+
UREA, data = dataClean)

summary(linearModel2)

##
## Call:
## lm(formula = DURATION.OF.STAY ~ factor(GENDER) + AGE + HB + TLC +
##     GLUCOSE + UREA, data = dataClean)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -15.320  -2.809  -0.844   1.523  89.475
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)      6.2491529  0.3934462  15.883  < 2e-16 ***
## factor(GENDER)M  0.2356628  0.1013095   2.326     0.02 *
```

```
## AGE              0.0162250  0.0036026   4.504 6.75e-06 ***
## HB              -0.2464693  0.0224853 -10.961  < 2e-16 ***
## TLC              0.0719215  0.0066435  10.826  < 2e-16 ***
## GLUCOSE          0.0037277  0.0005514   6.761 1.45e-11 ***
## UREA             0.0167179  0.0013300  12.570  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.657 on 10118 degrees of freedom
## Multiple R-squared:  0.07281,    Adjusted R-squared:  0.07226
## F-statistic: 132.4 on 6 and 10118 DF,  p-value: < 2.2e-16
```

Obviously GENDER has no effect on how long a patient will remain on admission. Therefore, removing GENDER from the linear model

```
linearModel3 <- lm(DURATION.OF.STAY~AGE+HB+TLC+GLUCOSE+UREA, data = da
taClean)

summary(linearModel3)

##
## Call:
## lm(formula = DURATION.OF.STAY ~ AGE + HB + TLC + GLUCOSE + UREA,
##     data = dataClean)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -15.205  -2.793  -0.846   1.525  89.332
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  6.1874430  0.3926364  15.759  < 2e-16 ***
## AGE          0.0163724  0.0036028   4.544 5.58e-06 ***
## HB          -0.2304180  0.0214050 -10.765  < 2e-16 ***
## TLC          0.0716079  0.0066435  10.779  < 2e-16 ***
## GLUCOSE      0.0036914  0.0005513   6.696 2.25e-11 ***
## UREA         0.0170265  0.0013236  12.863  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.658 on 10119 degrees of freedom
## Multiple R-squared:  0.07232,    Adjusted R-squared:  0.07186
## F-statistic: 157.8 on 5 and 10119 DF,  p-value: < 2.2e-16
```

With the low R-squared value, it means that the predictors can only account for 7% of the response function using this model.

Proceeding with this linear equation for further analysis that will test the appropriateness of the model.

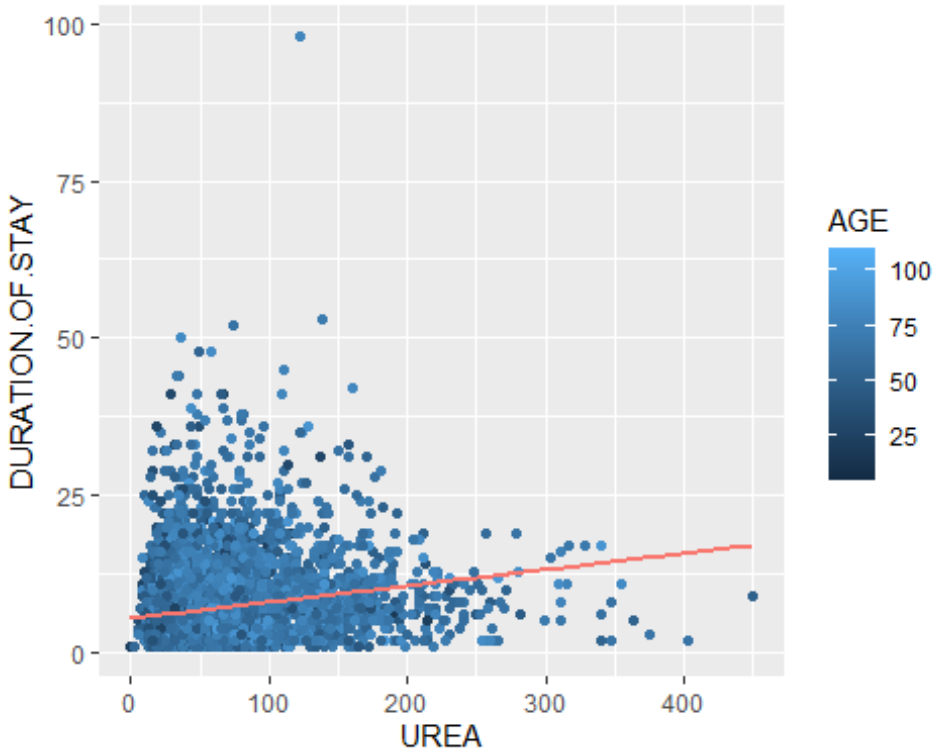
```
confint(linearModel3)
```

```
##                    2.5 %       97.5 %
## (Intercept)  5.41779782  6.957088274
## AGE          0.00931007  0.023434654
## HB          -0.27237604 -0.188459993
## TLC          0.05858525  0.084630572
## GLUCOSE      0.00261083  0.004771956
## UREA         0.01443193  0.019621129
```

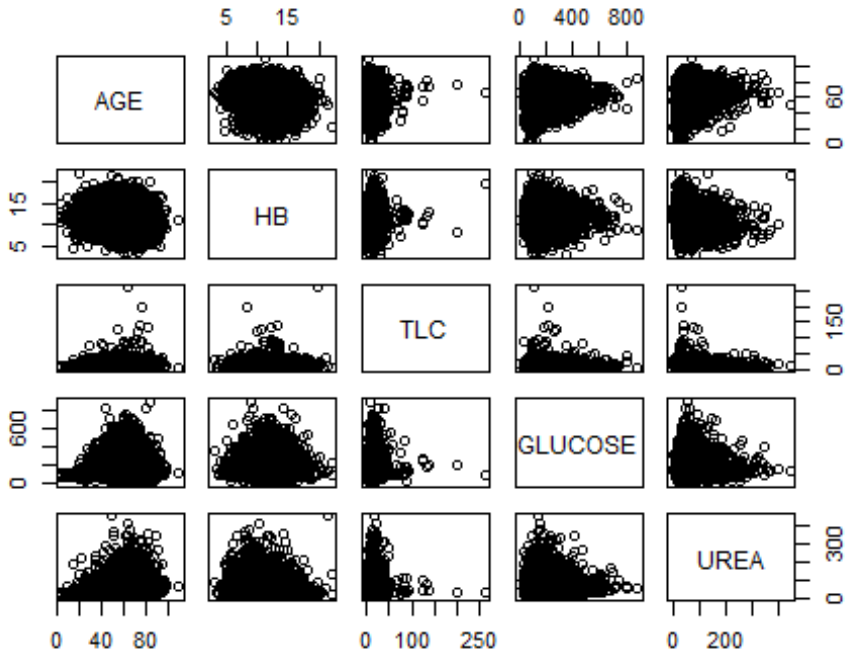
None of the independent variables crossed the zero mark between the low and high points in their 95% confidence interval, so we can utilize all in the model. Therefore age will vary by 0.012 to 0.025 for every day spent in admission, if HB, TLC, GLUCOSE and UREA are held constant.The same interpretation goes to other independent variables.

Plotting the relationship between one of the predictors and the response function

```
Duration = function(x){coef(linearModel3)[6]*x+coef(linearModel3)[5]*x
+coef(linearModel3)[4]*x+coef(linearModel3)[3]*x+coef(linearModel3)[2]
*x+coef(linearModel3)[1]}
ggplot(data=dataClean,mapping= aes(x=UREA,y=DURATION.OF.STAY,colour=AG
E))+geom_point()+ geom_smooth(method = "lm", se=FALSE, color=scales::h
ue_pal()(2)[1])
```



```
library(mctest)

pairs(~AGE+HB+TLC+GLUCOSE+UREA, data = dataClean)
```



The plot above shows minimal linear relationships. However, TLC seems to be linear with the rest of the independent variables.

The next will be to test for multicollinearity between the independent variables.

```
imcdiag(mod = linearModel3, method = "VIF")

##
## Call:
## imcdiag(mod = linearModel3, method = "VIF")
##
##
##  VIF Multicollinearity Diagnostics
##
##             VIF detection
## AGE     1.0690         0
## HB      1.1435         0
## TLC     1.0400         0
## GLUCOSE 1.0412         0
## UREA    1.1829         0
##
## NOTE:  VIF Method Failed to detect multicollinearity
```

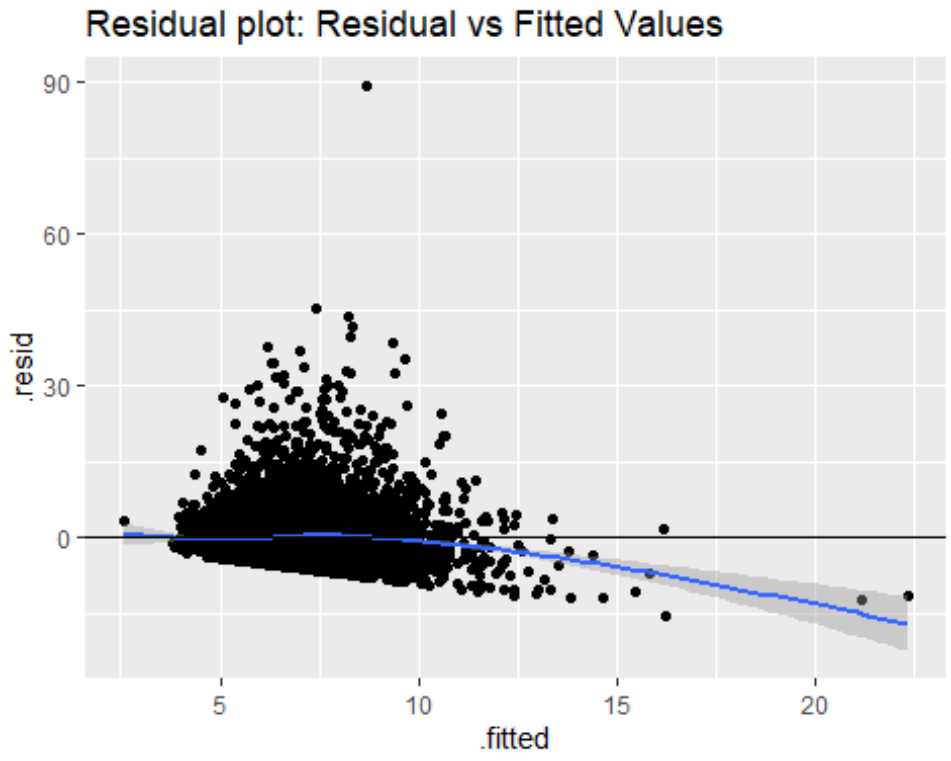
```
##
##
## 0 --> COLLINEARITY is not detected by the test
##
## ===================================
```

The above VIF test failed to detect multicollinearity.

The next will be to test for heteroscedasticity (non constant variance) based on the following assumptions:

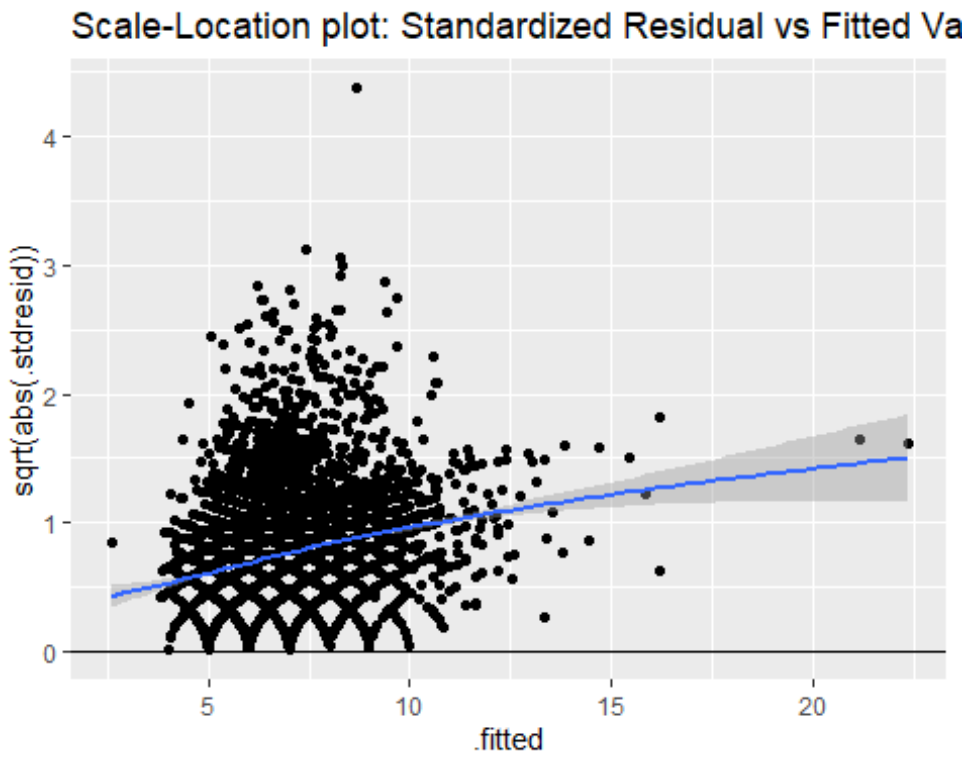
$$H_0: Heteroscedasticity\ is\ not\ present\ (Homoscedasticity)$$
$$H_a: Heteroscedasticity\ is\ present$$

```
ggplot(linearModel3, aes(x=.fitted, y=.resid)) + geom_point() + geom_s
mooth() + geom_hline(yintercept = 0) + ggtitle("Residual plot: Residua
l vs Fitted Values")
```



There seem to be some scatter in the residuals plot above as the residuals tend to deviate from a horizontal band. There is also evidence of varying spread of the residuals. Suspecting heteroscedasticity.

```
ggplot(linearModel3, aes(x=.fitted, y=sqrt(abs(.stdresid)))) + geom_po
int() + geom_smooth() + geom_hline(yintercept = 0) + ggtitle("Scale-Lo
cation plot: Standardized Residual vs Fitted Values")
```



There seem to be some scatter in the scaled location plot of the residuals above as the residuals tend to deviate from a horizontal band.

The next will be to perform the BPtest for heteroscedasticity.

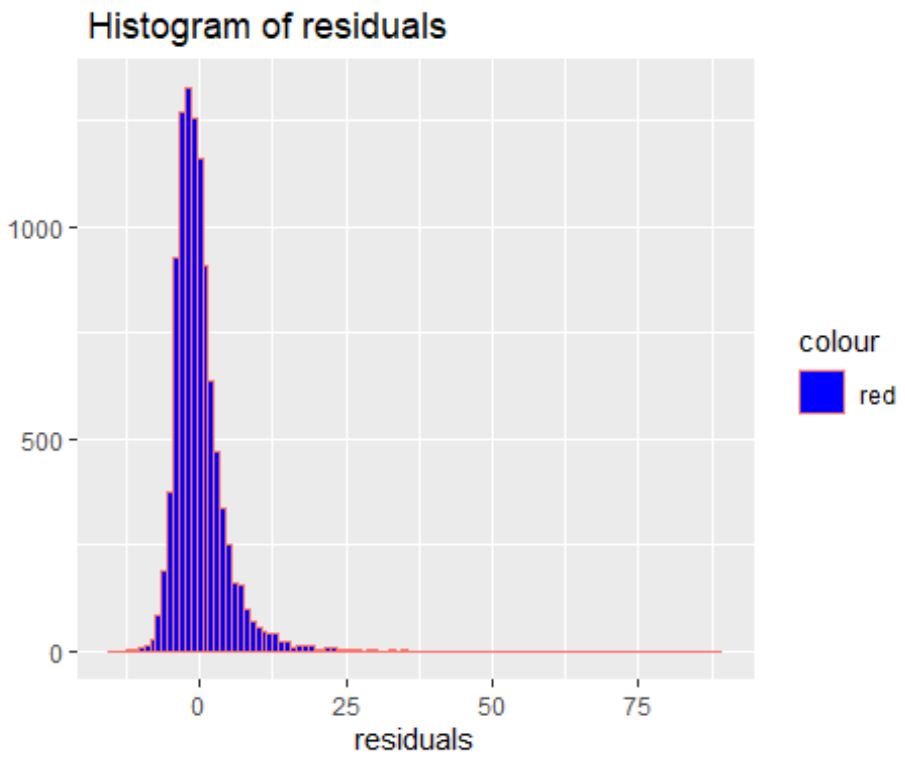
```
library(lmtest) # installing package

bptest(linearModel3)

##
##  studentized Breusch-Pagan test
##
## data:  linearModel3
## BP = 129.19, df = 5, p-value < 2.2e-16
```

With a P-value of 2.2e-16 which is very much less than 0.05, we reject the null hypothesis that states that heterostedasticity is not present.

The next set of test will be the normality test.

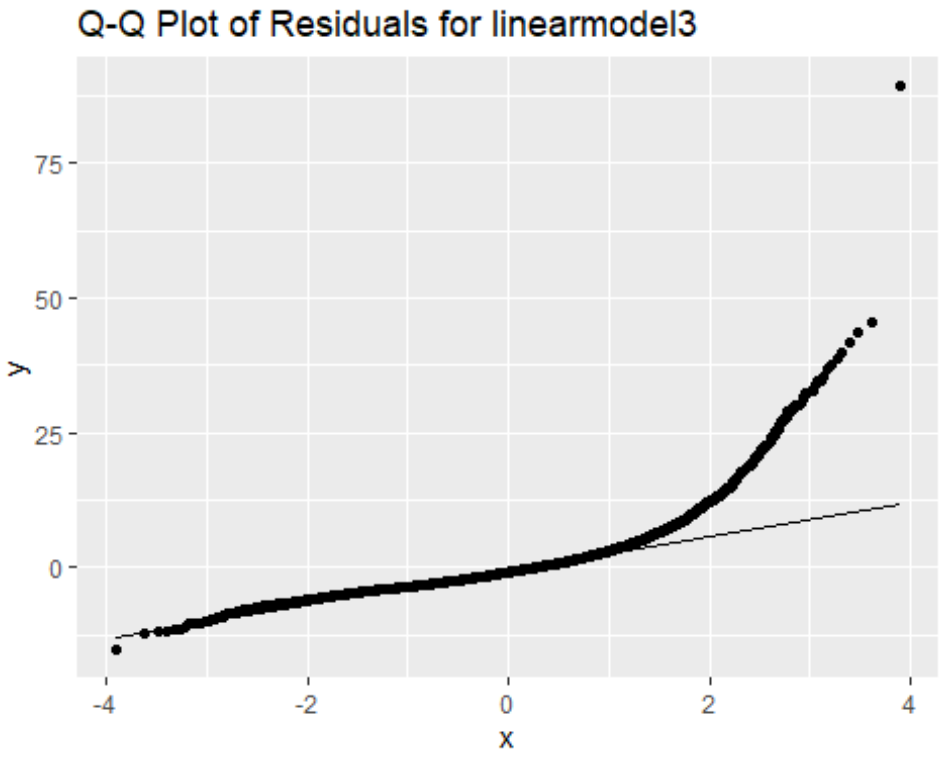
$$H_0: \textit{The sample data are significantly normally distributed}$$
$$H_a: \textit{The sample data are not significantly normally distributed}$$

```
qplot(residuals(linearModel3),geom ="histogram", binwidth = 1, main =
" Histogram of residuals", xlab = "residuals", color = "red", fill = I
("blue"))
```



From the plot above, the residuals seem to be normally distributed with a slight skew to the right. We will then proceed to do the Q-Q plot.

```
ggplot(dataClean, aes(sample = linearModel3$residuals)) + stat_qq() +
stat_qq_line() + ggtitle("Q-Q Plot of Residuals for linearmodel3")
```



The Q-Q plot above shows that the residuals seem to have normal distribution as they overlay the normal line plot except towards the right end of the plot where they separate from the line.

The next will be to carry out the Shapiro test for normality.

```
shapiro.test(residuals(linearModel3)[10:5000]) # Testing for normality
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(linearModel3)[10:5000]
## W = 0.7804, p-value < 2.2e-16
```

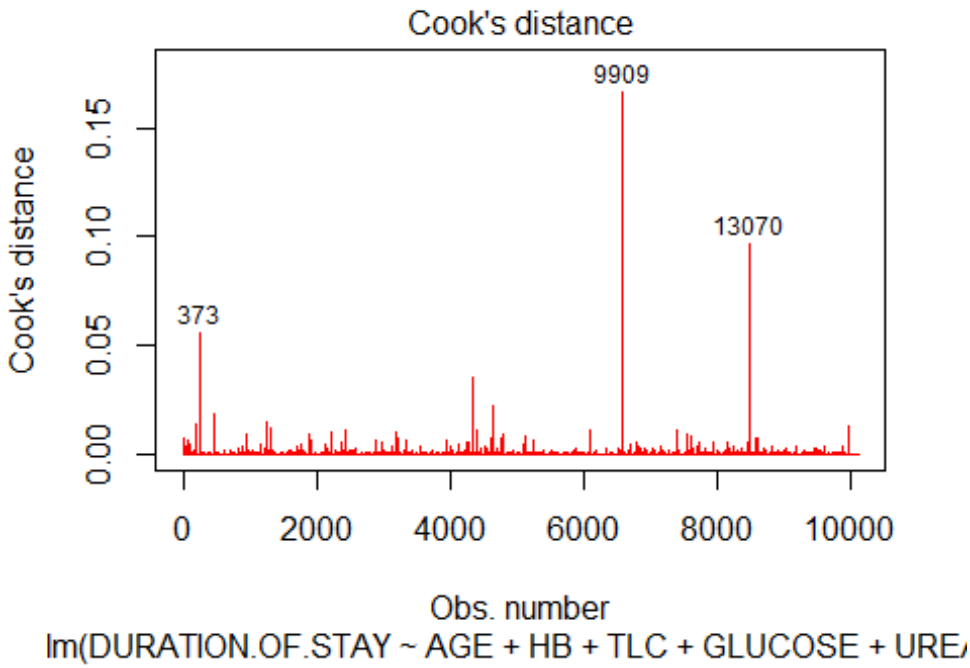
With such low p-value<2.2e-16<0.05, it is evident that the null hypothesis that there is normal distribution in the residuals of the model should be rejected in this case.

```
dataClean[cooks.distance(linearModel3)>1,]
```

```
##  [1] MRD.No.                AGE                    GENDER
##  [4] RURAL                  DURATION.OF.STAY       OUTCOME
##  [7] SMOKING                ALCOHOL                DM
## [10] HTN                    CAD                    PRIOR.CMP
## [13] CKD                    HB                     TLC
## [16] PLATELETS              GLUCOSE                UREA
## [19] CREATININE             RAISED.CARDIAC.ENZYMES EF
## [22] SEVERE.ANAEMIA         ANAEMIA                STABLE.ANGINA
```

```
## [25] ACS                          STEMI                          HEART.FAILURE
## [28] AKI
## <0 rows> (or 0-length row.names)
```

```
plot(linearModel3, pch=18, col="red", which = c(4))
```



There does not seem to be any significant outliers as all cook's distances computed are less than 0.5

Using stepwise regression to confirm model selection

```
library(olsrr)
```

```
forwardModel = ols_step_forward_p(linearModel1, penter = 0.05, details
= TRUE) # stepwise regression model using "forward" option.
```

```
## Forward Selection Method
## ---------------------------
##
## Candidate Terms:
##
## 1. factor(GENDER)
## 2. AGE
## 3. HB
## 4. TLC
## 5. PLATELETS
```

```
## 6. GLUCOSE
## 7. UREA
## 8. CREATININE
##
## We are selecting variables based on p value...
##
##
## Forward Selection: Step 1
##
## - UREA
##
##                          Model Summary
## ----------------------------------------------------------------
## R                        0.203       RMSE                 4.734
## R-Squared                0.041       Coef. Var           71.801
## Adj. R-Squared           0.041       MSE                 22.412
## Pred R-Squared           0.041       MAE                  3.161
## ----------------------------------------------------------------
##  RMSE: Root Mean Square Error
##  MSE: Mean Square Error
##  MAE: Mean Absolute Error
##
##                                  ANOVA
## -------------------------------------------------------------------
-------
##                   Sum of
##                  Squares           DF    Mean Square       F
Sig.
## -------------------------------------------------------------------
-------
## Regression      9789.486            1       9789.486    436.788
0.0000
## Residual      226881.283        10123         22.412
## Total         236670.770        10124
## -------------------------------------------------------------------
-------
##
##                                 Parameter Estimates
## -------------------------------------------------------------------
------------------
##       model     Beta    Std. Error    Std. Beta      t        Sig
lower    upper
## -------------------------------------------------------------------
------------------
## (Intercept)    5.365         0.075                 71.254    0.000
5.217    5.513
```

```
##        UREA    0.026         0.001        0.203    20.899    0.000
0.023    0.028
## ----------------------------------------------------------------
------------------
##
##
##
## Forward Selection: Step 2
##
## - HB
##
##                          Model Summary
## ----------------------------------------------------------------
## R                        0.231       RMSE                 4.705
## R-Squared                0.053       Coef. Var           71.353
## Adj. R-Squared           0.053       MSE                 22.134
## Pred R-Squared           0.053       MAE                  3.151
## ----------------------------------------------------------------
##  RMSE: Root Mean Square Error
##  MSE: Mean Square Error
##  MAE: Mean Absolute Error
##
##                                   ANOVA
## ----------------------------------------------------------------
-------
##                   Sum of
##                  Squares           DF    Mean Square       F
Sig.
## ----------------------------------------------------------------
-------
## Regression     12634.432            2       6317.216    285.413
0.0000
## Residual      224036.338        10122         22.134
## Total         236670.770        10124
## ----------------------------------------------------------------
-------
##
##                                     Parameter Estimates
## ----------------------------------------------------------------
----------------------
##       model      Beta    Std. Error    Std. Beta       t        Sig
lower     upper
## ----------------------------------------------------------------
----------------------
## (Intercept)     8.586         0.294                  29.224    0.00
0     8.010     9.162
```

```
##        UREA     0.021          0.001         0.165     16.162    0.00
0     0.018     0.024
##          HB    -0.243          0.021        -0.116    -11.337    0.00
0    -0.284    -0.201
## ----------------------------------------------------------------------
----------------------
##
##
##
## Forward Selection: Step 3
##
## - TLC
##
##                           Model Summary
## ---------------------------------------------------------------
## R                        0.257       RMSE                 4.673
## R-Squared                0.066       Coef. Var           70.877
## Adj. R-Squared           0.066       MSE                 21.840
## Pred R-Squared           0.065       MAE                  3.112
## ---------------------------------------------------------------
##  RMSE: Root Mean Square Error
##  MSE: Mean Square Error
##  MAE: Mean Absolute Error
##
##                                    ANOVA
## ----------------------------------------------------------------------
-------
##                   Sum of
##                  Squares            DF    Mean Square       F
Sig.
## ----------------------------------------------------------------------
-------
## Regression     15631.813             3       5210.604    238.585
0.0000
## Residual      221038.957         10121         21.840
## Total         236670.770         10124
## ----------------------------------------------------------------------
-------
##
##                                     Parameter Estimates
## ----------------------------------------------------------------------
----------------------
##       model      Beta    Std. Error    Std. Beta       t        Sig
lower     upper
## ----------------------------------------------------------------------
----------------------
```

```
## (Intercept)     7.890         0.298                  26.490    0.00
0     7.306     8.473
##        UREA     0.019         0.001        0.148     14.412    0.00
0     0.016     0.021
##          HB    -0.250         0.021       -0.120    -11.781    0.00
0    -0.292    -0.209
##         TLC     0.077         0.007        0.114     11.715    0.00
0     0.064     0.090
## ----------------------------------------------------------------------
----------------------
##
##
##
## Forward Selection: Step 4
##
## - GLUCOSE
##
##                          Model Summary
## ------------------------------------------------------------------
## R                        0.265       RMSE                 4.663
## R-Squared                0.070       Coef. Var           70.715
## Adj. R-Squared           0.070       MSE                 21.739
## Pred R-Squared           0.069       MAE                  3.098
## ------------------------------------------------------------------
##  RMSE: Root Mean Square Error
##  MSE: Mean Square Error
##  MAE: Mean Absolute Error
##
##                                    ANOVA
## ----------------------------------------------------------------------
-------
##                   Sum of
##                  Squares           DF    Mean Square       F
Sig.
## ----------------------------------------------------------------------
-------
## Regression     16667.391            4       4166.848    191.672
0.0000
## Residual      220003.379        10120         21.739
## Total         236670.770        10124
## ----------------------------------------------------------------------
-------
##
##                                      Parameter Estimates
## ----------------------------------------------------------------------
----------------------
```

```
##       model      Beta    Std. Error    Std. Beta       t        Sig
lower     upper
## ------------------------------------------------------------------
----------------------
## (Intercept)     7.285         0.310                 23.518    0.00
0     6.678     7.893
##        UREA     0.018         0.001        0.141     13.730    0.00
0     0.015     0.021
##          HB    -0.243         0.021       -0.116    -11.448    0.00
0    -0.285    -0.202
##         TLC     0.071         0.007        0.105     10.721    0.00
0     0.058     0.084
##     GLUCOSE     0.004         0.001        0.067      6.902    0.00
0     0.003     0.005
## ------------------------------------------------------------------
----------------------
##
##
##
## Forward Selection: Step 5
##
## - AGE
##
##                          Model Summary
## ----------------------------------------------------------------
## R                        0.269       RMSE                 4.658
## R-Squared                0.072       Coef. Var           70.646
## Adj. R-Squared           0.072       MSE                 21.697
## Pred R-Squared           0.071       MAE                  3.092
## ----------------------------------------------------------------
##  RMSE: Root Mean Square Error
##  MSE: Mean Square Error
##  MAE: Mean Absolute Error
##
##                                   ANOVA
## ------------------------------------------------------------------
-------
##                   Sum of
##                  Squares           DF    Mean Square       F
Sig.
## ------------------------------------------------------------------
-------
## Regression     17115.452            5       3423.090    157.765
0.0000
## Residual      219555.317        10119         21.697
## Total         236670.770        10124
```

```
## ----------------------------------------------------------------
-------
##
##                                  Parameter Estimates
## ----------------------------------------------------------------
----------------------
##       model      Beta    Std. Error    Std. Beta       t        Sig
lower     upper
## ----------------------------------------------------------------
----------------------
## (Intercept)     6.187         0.393                  15.759    0.00
0     5.418     6.957
##        UREA     0.017         0.001        0.134     12.863    0.00
0     0.014     0.020
##          HB    -0.230         0.021       -0.110    -10.765    0.00
0    -0.272    -0.188
##         TLC     0.072         0.007        0.105     10.779    0.00
0     0.059     0.085
##     GLUCOSE     0.004         0.001        0.065      6.696    0.00
0     0.003     0.005
##         AGE     0.016         0.004        0.045      4.544    0.00
0     0.009     0.023
## ----------------------------------------------------------------
----------------------
##
##
##
## Forward Selection: Step 6
##
## - factor(GENDER)
##
##                          Model Summary
## ---------------------------------------------------------------
## R                        0.270       RMSE                 4.657
## R-Squared                0.073       Coef. Var           70.631
## Adj. R-Squared           0.072       MSE                 21.688
## Pred R-Squared           0.071       MAE                  3.092
## ---------------------------------------------------------------
##  RMSE: Root Mean Square Error
##  MSE: Mean Square Error
##  MAE: Mean Absolute Error
##
##                                   ANOVA
## ----------------------------------------------------------------
-------
##                    Sum of
```

```
##                   Squares            DF    Mean Square       F
Sig.
## ----------------------------------------------------------------------
-------
## Regression     17232.807             6       2872.134     132.43
0.0000
## Residual      219437.963         10118         21.688
## Total         236670.770         10124
## ----------------------------------------------------------------------
-------
##
##                                       Parameter Estimates
## ----------------------------------------------------------------------
-------------------------
##           model      Beta    Std. Error    Std. Beta       t
Sig      lower     upper
## ----------------------------------------------------------------------
-------------------------
##     (Intercept)     6.249          0.393                  15.883
0.000     5.478     7.020
##            UREA     0.017          0.001         0.132     12.570
0.000     0.014     0.019
##              HB    -0.246          0.022        -0.118    -10.961
0.000    -0.291    -0.202
##             TLC     0.072          0.007         0.106     10.826
0.000     0.059     0.085
##         GLUCOSE     0.004          0.001         0.066      6.761
0.000     0.003     0.005
##             AGE     0.016          0.004         0.045      4.504
0.000     0.009     0.023
## factor(GENDER)M     0.236          0.101         0.023      2.326
0.020     0.037     0.434
## ----------------------------------------------------------------------
-------------------------
##
##
##
## No more variables to be added.
##
## Variables Entered:
##
## + UREA
## + HB
## + TLC
## + GLUCOSE
## + AGE
```

```
## + factor(GENDER)
##
##
## Final Model Output
## ------------------
##
##                          Model Summary
## ----------------------------------------------------------------
## R                        0.270       RMSE                 4.657
## R-Squared                0.073       Coef. Var           70.631
## Adj. R-Squared           0.072       MSE                 21.688
## Pred R-Squared           0.071       MAE                  3.092
## ----------------------------------------------------------------
##  RMSE: Root Mean Square Error
##  MSE: Mean Square Error
##  MAE: Mean Absolute Error
##
##                                    ANOVA
## ---------------------------------------------------------------------
-------
##                   Sum of
##                  Squares            DF    Mean Square       F
Sig.
## ---------------------------------------------------------------------
-------
## Regression     17232.807             6       2872.134     132.43
0.0000
## Residual      219437.963         10118         21.688
## Total         236670.770         10124
## ---------------------------------------------------------------------
-------
##
##                                       Parameter Estimates
## ---------------------------------------------------------------------
--------------------------
##           model      Beta    Std. Error    Std. Beta       t
Sig      lower     upper
## ---------------------------------------------------------------------
--------------------------
##     (Intercept)     6.249         0.393                  15.883
0.000     5.478     7.020
##            UREA     0.017         0.001         0.132     12.570
0.000     0.014     0.019
##              HB    -0.246         0.022        -0.118    -10.961
0.000    -0.291    -0.202
##             TLC     0.072         0.007         0.106     10.826
```

```
0.000     0.059     0.085
##          GLUCOSE     0.004         0.001        0.066       6.761
0.000     0.003     0.005
##              AGE     0.016         0.004        0.045       4.504
0.000     0.009     0.023
## factor(GENDER)M     0.236         0.101        0.023       2.326
0.020     0.037     0.434
## ----------------------------------------------------------------
-------------------------
```

The forward stepwise regression also picked UREA+HB+TLC+GLUCOSE+AGE as the independent variables

Applying the backward stepwise regression to pick the right equation

```
backwardModel = ols_step_backward_p(linearModel1, prem = 0.05, details
= T) # stepwise regression model using "backward" option.
```

```
## Backward Elimination Method
## ---------------------------
##
## Candidate Terms:
##
## 1 . factor(GENDER)
## 2 . AGE
## 3 . HB
## 4 . TLC
## 5 . PLATELETS
## 6 . GLUCOSE
## 7 . UREA
## 8 . CREATININE
##
## We are eliminating variables based on p value...
##
## - PLATELETS
##
## Backward Elimination: Step 1
##
##  Variable PLATELETS Removed
##
##                          Model Summary
## ---------------------------------------------------------------
## R                        0.270       RMSE                 4.657
## R-Squared                0.073       Coef. Var           70.632
## Adj. R-Squared           0.072       MSE                 21.689
## Pred R-Squared           0.071       MAE                  3.092
## ---------------------------------------------------------------
```

```
##  RMSE: Root Mean Square Error
##  MSE: Mean Square Error
##  MAE: Mean Absolute Error
##
##                                    ANOVA
## --------------------------------------------------------------------
-------
##                   Sum of
##                  Squares           DF    Mean Square       F
Sig.
## --------------------------------------------------------------------
-------
## Regression     17248.192            7       2464.027     113.61
0.0000
## Residual      219422.578        10117         21.689
## Total         236670.770        10124
## --------------------------------------------------------------------
-------
##
##                                       Parameter Estimates
## --------------------------------------------------------------------
--------------------------
##           model      Beta    Std. Error    Std. Beta       t
Sig      lower     upper
## --------------------------------------------------------------------
--------------------------
##     (Intercept)     6.204         0.397                 15.621
0.000     5.425     6.982
## factor(GENDER)M     0.223         0.102        0.022      2.179
0.029     0.022     0.424
##             AGE     0.016         0.004        0.045      4.540
0.000     0.009     0.023
##              HB    -0.244         0.023       -0.117    -10.765
0.000    -0.288    -0.200
##             TLC     0.072         0.007        0.106     10.804
0.000     0.059     0.085
##         GLUCOSE     0.004         0.001        0.066      6.780
0.000     0.003     0.005
##            UREA     0.016         0.002        0.123      8.360
0.000     0.012     0.019
##      CREATININE     0.051         0.060        0.012      0.842
0.400    -0.068     0.169
## --------------------------------------------------------------------
--------------------------
##
##
```

```
## - CREATININE
##
## Backward Elimination: Step 2
##
##  Variable CREATININE Removed
##
##                          Model Summary
## ----------------------------------------------------------------
## R                        0.270       RMSE                 4.657
## R-Squared                0.073       Coef. Var           70.631
## Adj. R-Squared           0.072       MSE                 21.688
## Pred R-Squared           0.071       MAE                  3.092
## ----------------------------------------------------------------
##  RMSE: Root Mean Square Error
##  MSE: Mean Square Error
##  MAE: Mean Absolute Error
##
##                                    ANOVA
## ------------------------------------------------------------------------
-------
##                   Sum of
##                  Squares           DF    Mean Square       F
Sig.
## ------------------------------------------------------------------------
-------
## Regression     17232.807            6       2872.134     132.43
0.0000
## Residual      219437.963        10118         21.688
## Total         236670.770        10124
## ------------------------------------------------------------------------
-------
##
##                                       Parameter Estimates
## ------------------------------------------------------------------------
--------------------------
##           model      Beta    Std. Error    Std. Beta       t
Sig      lower     upper
## ------------------------------------------------------------------------
--------------------------
##     (Intercept)     6.249         0.393                 15.883
0.000     5.478     7.020
## factor(GENDER)M     0.236         0.101        0.023      2.326
0.020     0.037     0.434
##             AGE     0.016         0.004        0.045      4.504
0.000     0.009     0.023
##              HB    -0.246         0.022       -0.118    -10.961
```

```
0.000    -0.291    -0.202
##              TLC     0.072         0.007        0.106     10.826
0.000     0.059     0.085
##          GLUCOSE     0.004         0.001        0.066      6.761
0.000     0.003     0.005
##             UREA     0.017         0.001        0.132     12.570
0.000     0.014     0.019
## ---------------------------------------------------------------------
-------------------------
##
##
##
## No more variables satisfy the condition of p value = 0.05
##
##
## Variables Removed:
##
## - PLATELETS
## - CREATININE
##
##
## Final Model Output
## ------------------
##
##                          Model Summary
## ----------------------------------------------------------------
## R                        0.270       RMSE                 4.657
## R-Squared                0.073       Coef. Var           70.631
## Adj. R-Squared           0.072       MSE                 21.688
## Pred R-Squared           0.071       MAE                  3.092
## ----------------------------------------------------------------
##  RMSE: Root Mean Square Error
##  MSE: Mean Square Error
##  MAE: Mean Absolute Error
##
##                                    ANOVA
## ---------------------------------------------------------------------
-------
##                   Sum of
##                  Squares           DF    Mean Square       F
Sig.
## ---------------------------------------------------------------------
-------
## Regression     17232.807            6       2872.134     132.43
0.0000
## Residual      219437.963        10118         21.688
```

```
## Total         236670.770        10124
## ------------------------------------------------------------------
-------
##
##                                        Parameter Estimates
## ------------------------------------------------------------------
-------------------------
##           model      Beta    Std. Error    Std. Beta       t
Sig      lower     upper
## ------------------------------------------------------------------
-------------------------
##     (Intercept)     6.249         0.393                 15.883
0.000     5.478     7.020
## factor(GENDER)M     0.236         0.101        0.023      2.326
0.020     0.037     0.434
##             AGE     0.016         0.004        0.045      4.504
0.000     0.009     0.023
##              HB    -0.246         0.022       -0.118    -10.961
0.000    -0.291    -0.202
##             TLC     0.072         0.007        0.106     10.826
0.000     0.059     0.085
##         GLUCOSE     0.004         0.001        0.066      6.761
0.000     0.003     0.005
##            UREA     0.017         0.001        0.132     12.570
0.000     0.014     0.019
## ------------------------------------------------------------------
-------------------------
```

The backward stepwise regression also picked UREA+HB+TLC+GLUCOSE+AGE as the independent variables

IV.2.2 - REGRESSION TREE IN PREDICTING DURATION OF STAY IN HOSPITAL

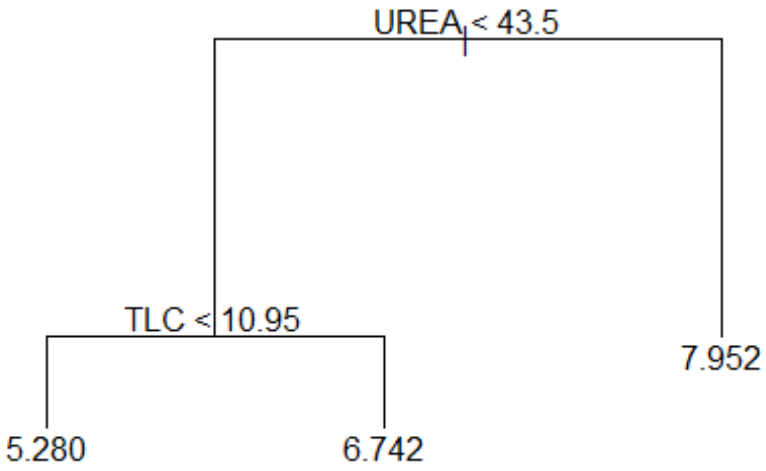
Applying the model to Regression Tree on a 25:75 percent split

```
library(tree)
set.seed (10)
idtree=sample(1:nrow(dataClean),3/4*nrow(dataClean))
trainTree=dataClean[idtree,]
testTree=dataClean[-idtree,]
tree.clean<-tree(DURATION.OF.STAY~AGE+HB+TLC+GLUCOSE+UREA, trainTree)
summary(tree.clean)
```

```
##
## Regression tree:
## tree(formula = DURATION.OF.STAY ~ AGE + HB + TLC + GLUCOSE +
##     UREA, data = trainTree)
## Variables actually used in tree construction:
## [1] "UREA" "TLC"
## Number of terminal nodes:  3
## Residual mean deviance:  20.97 = 159100 / 7590
## Distribution of residuals:
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -6.9520 -2.9520 -0.9519  0.0000  1.7200 40.0500
```

Plotting the tree

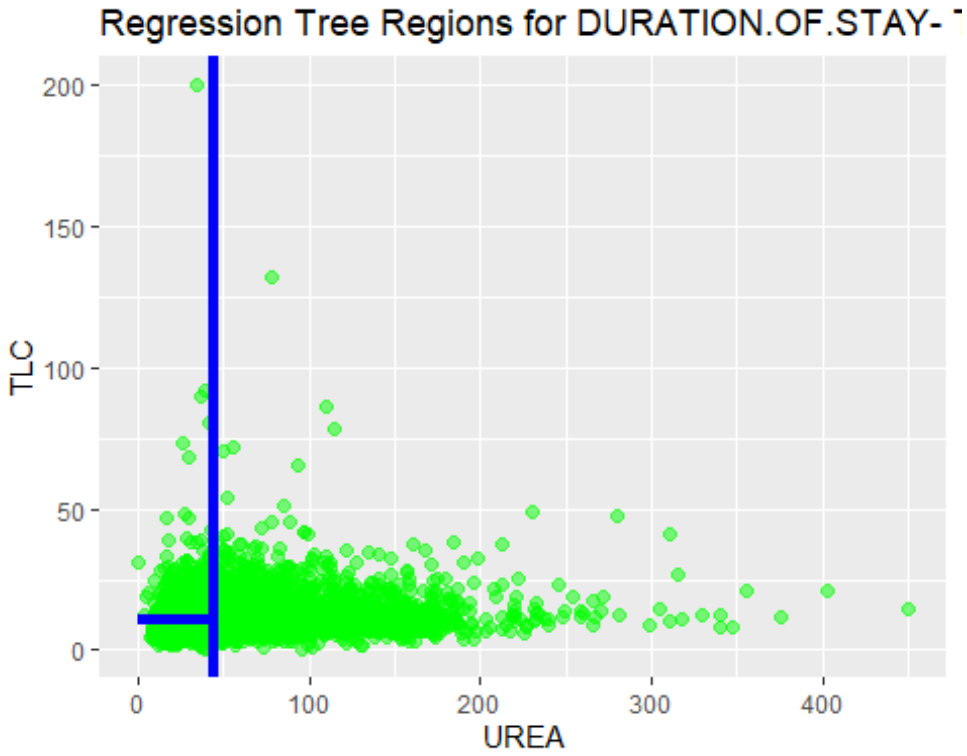
```
plot(tree.clean)
text(tree.clean ,pretty =0)
```



With the above minimalistic number of nodes, there is no need to prune the tree further. Therefore, the Regression Tree only used UREA and TLC to predict “DURATION.OF.STAY” in the model.

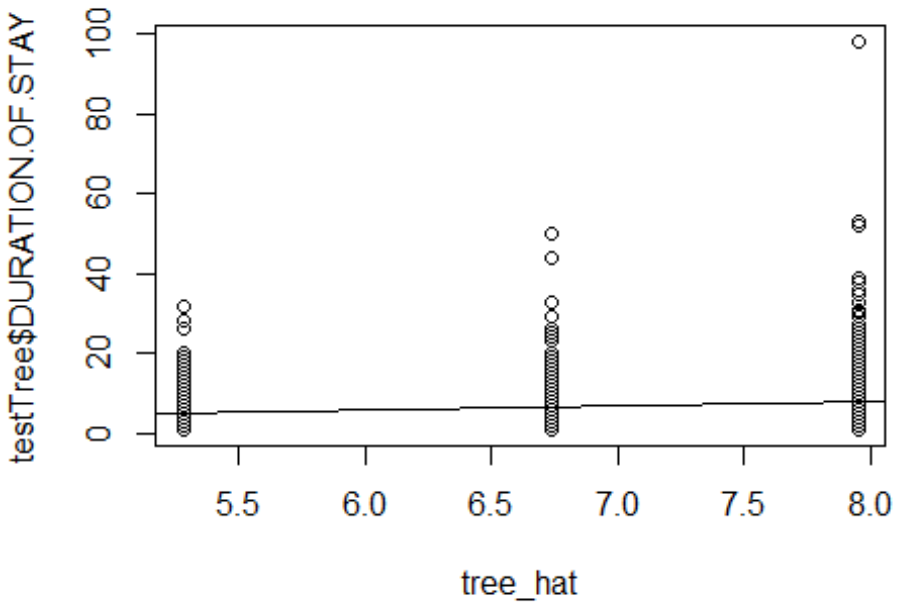
```
#Generating plots with regions
ggplot(data=trainTree, aes(x=UREA, y=TLC))+
  geom_point(size=2, alpha=0.5, color = "green")+
  geom_vline(xintercept=43.5, color="blue", size=2)+
```

```
  geom_segment(aes(x=0,y=10.95, xend=43.5,yend=10.95), color="blue", s
ize=2)+
  ggtitle("Regression Tree Regions for DURATION.OF.STAY- TLC x UREA")
```



Applying the tree to the test set, we get

```
tree_hat<-predict(tree.clean,testTree)
plot(tree_hat,testTree$DURATION.OF.STAY)
abline(0,1)
```




```
sqrt(mean((tree_hat-testTree$DURATION.OF.STAY)^2))

## [1] 5.007815
```

The root square of the mean square error of the tree is 5.008

IV.2.3 - LINEAR REGRESSION VS REGRESSION TREE COMPARISON IN PREDICTING DURATION OF STAY IN HOSPITAL

```
library(caret)
library(AppliedPredictiveModeling)

head(dataClean)

##   MRD.No. AGE GENDER RURAL DURATION.OF.STAY   OUTCOME SMOKING ALCOH
OL DM HTN
## 1  234735  81      M     R                3 DISCHARGE       0
0  1   0
## 2  234696  65      M     R                5 DISCHARGE       0
1  0   1
## 4  234635  67      F     U                8 DISCHARGE       0
0  0   1
## 5  234486  60      F     U               23 DISCHARGE       0
0  0   1
## 6  234675  44      M     U               10 DISCHARGE       0
```

```
0  1   1
## 7  234563  56      F     U                6 DISCHARGE       0
0  1   1
##   CAD PRIOR.CMP CKD   HB  TLC PLATELETS GLUCOSE UREA CREATININE
## 1   0         0   0  9.5 16.1       337      80   34       0.90
## 2   1         0   0 13.7  9.0       149     112   18       0.90
## 4   1         0   0 12.8  9.9       286     130   27       0.60
## 5   0         1   0 13.6  9.1        26     144   55       1.25
## 6   1         1   0 13.5 22.3       322     217   51       0.90
## 7   1         1   0 13.3 12.6       166     277   28       0.60
##   RAISED.CARDIAC.ENZYMES EF SEVERE.ANAEMIA ANAEMIA STABLE.ANGINA AC
S STEMI
## 1                      1 35              0       1             0
1     0
## 2                      0 42              0       0             0
0     0
## 4                      0 42              0       0             0
0     0
## 5                      0 16              0       0             0
0     0
## 6                      0 25              0       0             0
1     0
## 7                      0 30              0       0             0
1     1
##   HEART.FAILURE AKI
## 1             1   0
## 2             0   0
## 4             0   0
## 5             0   0
## 6             1   0
## 7             1   0
```

```
data(dataClean)
```

```
set.seed(10)
foldsz<-createFolds(dataClean$DURATION.OF.STAY, k=10)
summary(foldsz)
```

```
##        Length Class  Mode
## Fold01 1012   -none- numeric
## Fold02 1012   -none- numeric
## Fold03 1012   -none- numeric
## Fold04 1013   -none- numeric
## Fold05 1013   -none- numeric
## Fold06 1012   -none- numeric
## Fold07 1014   -none- numeric
```

```
## Fold08 1012   -none- numeric
## Fold09 1013   -none- numeric
## Fold10 1012   -none- numeric
```

```
RMSElinear<-function(id2){
  TrainzLM<-dataClean[-id2,]
  TestzLM<-dataClean[id2,]
  fitzLM<-lm(DURATION.OF.STAY~AGE+HB+TLC+GLUCOSE+UREA, data=TrainzLM)
  predzLM<-predict(fitzLM,TestzLM)
  return(sqrt(mean((TestzLM$DURATION.OF.STAY - predzLM)^2)))
}
```

```
rmsezLM=lapply(foldsz,RMSElinear)
rmsezLM
```

```
## $Fold01
## [1] 4.316222
##
## $Fold02
## [1] 4.566372
##
## $Fold03
## [1] 5.334692
##
## $Fold04
## [1] 4.296595
##
## $Fold05
## [1] 5.307696
##
## $Fold06
## [1] 4.500408
##
## $Fold07
## [1] 4.150382
##
## $Fold08
## [1] 4.622685
##
## $Fold09
## [1] 4.361638
##
## $Fold10
## [1] 4.977151
```

Now computing the average RMSE for all the folds from the linear model

```
mean(as.numeric(rmsezLM))
```

```
## [1] 4.643384
```

The computed mean RMSE from the linear model is 4.6434

Applying the same analysis to Regression Tree and getting the RMSE missclassification analysis

```
RMSEtree<-function(id2z){
  TrainRT<-dataClean[-id2z,]
  TestRT<-dataClean[id2z,]
  fitRT<-tree(DURATION.OF.STAY~AGE+HB+TLC+GLUCOSE+UREA, TrainRT)
  predRT<-predict(fitRT,TestRT)
  return(sqrt(mean((TestRT$DURATION.OF.STAY - predRT)^2)))
}

rmseRT=lapply(foldsz,RMSEtree)
rmseRT
```

```
## $Fold01
## [1] 4.35368
##
## $Fold02
## [1] 4.590497
##
## $Fold03
## [1] 5.388264
##
## $Fold04
## [1] 4.35427
##
## $Fold05
## [1] 5.370227
##
## $Fold06
## [1] 4.541843
##
## $Fold07
## [1] 4.159639
##
## $Fold08
## [1] 4.683187
##
## $Fold09
## [1] 4.379479
##
```

```
## $Fold10
## [1] 5.03549
```

Now computing the average RMSE for all the folds using the Regression Tree

```
mean(as.numeric(rmseRT))
```

```
## [1] 4.685657
```

The computed mean RMSE from the Regression Tree is 4.6857

Therefore the RMSE from both the Linear Model (4.6434) and the Regression Tree (4.6857) are about the same. The Linear Model has a slightly better RMSE than the Regression Tree, but there is not much of a difference between them.

PART IV -CONCLUSIONS AND RECOMMENDATIONS

Linear regression model is not able to predict the duration of stay effectively as it has an adjusted R-squared of around 7%. In all test cases tried, the null hypothesis assumptions were rejected.

Regression Tree computed with k-folds split did not show any improvement over the the same split carried out on the Linear Model.

In order to be able to effectively predict the duration of a patient's stay in the hospital, it is recommended to incorporate more advanced machine learning algorithms.

It is also recommended to probe the linear model further by incorporating interaction terms and higher order relationships between the predictor variables and the response variable.

PART V - MULTINOMIAL REGRESSION - OUTCOME AS RESPONSE VARIABLE

```
# Data Partition
set.seed(10)
ind = sample(2,nrow(dataClean),
             replace = TRUE,
             prob = c(0.75,0.25))

training = dataClean[ind==1,]
testing = dataClean[ind==2,]

library(nnet)
training$OUTCOME = relevel(training$OUTCOME , ref = "EXPIRY")
mymodel = multinom(OUTCOME~AGE+EF+TLC+HB+DURATION.OF.STAY+ALCOHOL+SMOK
ING+DM+CAD+CKD+PLATELETS+GLUCOSE+UREA
+ STABLE.ANGINA + ACS+ STEMI, data = training)
```

```
## # weights:  54 (34 variable)
## initial  value 8287.931106
## iter  10 value 5466.778223
## iter  20 value 4332.846418
## iter  30 value 2937.212536
## iter  40 value 2620.909149
## iter  50 value 2350.746483
## iter  60 value 2343.184090
## iter  60 value 2343.184080
## iter  60 value 2343.184080
## final  value 2343.184080
## converged
```

```
summary(mymodel)
```

```
## Call:
## multinom(formula = OUTCOME ~ AGE + EF + TLC + HB + DURATION.OF.STAY
+
##     ALCOHOL + SMOKING + DM + CAD + CKD + PLATELETS + GLUCOSE +
##     UREA + STABLE.ANGINA + ACS + STEMI, data = training)
##
## Coefficients:
##           (Intercept)         AGE         EF         TLC         HB
## DAMA        -3.335689 -0.01022090 0.06943216 -0.02586685 0.07778164
## DISCHARGE   -1.768795 -0.01412845 0.08637592 -0.07423450 0.14975432
##           DURATION.OF.STAY ALCOHOL1 SMOKING1       DM1      CAD1
CKD1
## DAMA           -0.01180685 2.292814 1.220637 0.6427785 0.8147678 0.
9946880
## DISCHARGE       0.09460193 1.847177 0.999394 0.6381260 1.5231775 0.
9690656
##             PLATELETS       GLUCOSE         UREA STABLE.ANGINA1
ACS1
## DAMA      0.002142865 -0.0009503827 -0.005048649       45.84682 -0.
1277824
## DISCHARGE 0.003836509 -0.0018901035 -0.013520273       46.94313 -0.
6684959
##               STEMI1
## DAMA      -0.5129005
## DISCHARGE -0.2669919
##
## Std. Errors:
##           (Intercept)         AGE          EF         TLC         H
B
## DAMA        0.7322778 0.006131834 0.006860337 0.006450345 0.0350286
9
```

```
## DISCHARGE   0.5400947 0.004507943 0.005513972 0.007208335 0.0259482
2
##           DURATION.OF.STAY  ALCOHOL1  SMOKING1       DM1      CAD1
CKD1
## DAMA            0.02018411 0.5275647 0.4765698 0.1758526 0.1620501
0.2718627
## DISCHARGE       0.01333967 0.5029664 0.4285453 0.1319422 0.1197485
0.2065251
##              PLATELETS      GLUCOSE        UREA STABLE.ANGINA1
ACS1
## DAMA      0.0007773120 0.0008259179 0.001925202      0.2133376 0.17
79117
## DISCHARGE 0.0005850808 0.0005924896 0.001473778      0.2133376 0.13
35685
##              STEMI1
## DAMA      0.2300104
## DISCHARGE 0.1623627
##
## Residual Deviance: 4686.368
## AIC: 4754.368
```

```r
#2 Tailed z test
z = summary(mymodel)$coefficients/summary(mymodel)$standard.errors
p = (1 - pnorm(abs(z), 0, 1)) * 2
p
```

```
##            (Intercept)         AGE EF          TLC           HB
## DAMA      5.232993e-06 0.095542611  0 6.068049e-05 2.638401e-02
## DISCHARGE 1.056725e-03 0.001723679  0 0.000000e+00 7.867416e-09
##           DURATION.OF.STAY     ALCOHOL1   SMOKING1          DM1
CAD1
## DAMA          5.585761e-01 1.386207e-05 0.01042824 2.569705e-04 4.9
59428e-07
## DISCHARGE     1.324052e-12 2.401272e-04 0.01969746 1.322067e-06 0.0
00000e+00
##                  CKD1    PLATELETS     GLUCOSE       UREA STABLE.A
NGINA1
## DAMA      2.534108e-04 5.837651e-03 0.249856151 0.00873129
0
## DISCHARGE 2.702286e-06 5.481682e-11 0.001422216 0.00000000
0
##                  ACS1     STEMI1
## DAMA      4.726124e-01 0.02575397
## DISCHARGE 5.589304e-07 0.10009029
```

```r
#Confusion Matrix and Misclassification Error - Training Data
j = predict(mymodel, training)
head(j)
```

```
## [1] DISCHARGE DISCHARGE DISCHARGE DISCHARGE DISCHARGE DISCHARGE
## Levels: EXPIRY DAMA DISCHARGE
```

```r
head(training$OUTCOME)
```

```
## [1] DISCHARGE DISCHARGE DISCHARGE DISCHARGE DISCHARGE DISCHARGE
## Levels: EXPIRY DAMA DISCHARGE
```

```r
#Confusion Matrix
tab = table(j,training$OUTCOME)
tab
```

```
##
## j           EXPIRY DAMA DISCHARGE
##   EXPIRY       127   18        48
##   DAMA           0    1         1
##   DISCHARGE    332  303      6714
```

```r
#Misclassification
1-sum(diag(tab))/sum(tab)
```

```
## [1] 0.09305408
```

```r
#Confusion Matrix and Misclassification Error - Test Data

#Confusion Matrix
i = predict(mymodel, testing)
tab1 = table(i,testing$OUTCOME)
tab1
```

```
##
## i           DAMA DISCHARGE EXPIRY
##   EXPIRY       5        26     44
##   DAMA         1         2      0
##   DISCHARGE  118      2295     90
```

```r
#Misclassification Error
m_class = (44+2295+1)/2581
1 - m_class
```

```
## [1] 0.09337466
```

```r
#Prediction and Model Assessment
```

```
#Model Assessment(training data)
tab/colSums(tab)
```

```
##
## j               EXPIRY       DAMA  DISCHARGE
##   EXPIRY    0.27668845 0.03921569 0.10457516
##   DAMA      0.00000000 0.00310559 0.00310559
##   DISCHARGE 0.04909064 0.04480260 0.99275469
```

```
#Model Assessment(testing data)

# For Discharge the prediction accuracy is
2295/(2295+2+26)
```

```
## [1] 0.9879466
```

```
# For EXPIRY the prediction accuracy is
44/(44+0+90)
```

```
## [1] 0.3283582
```

```
# For DAMA the prediction accuracy is
1/(5+1+118)
```

```
## [1] 0.008064516
```

```
#Overall Misclassification Rate
(5+26+2+0+118+90)/(5+26+44+1+2+0+118+2295+90)
```

```
## [1] 0.09337466
```

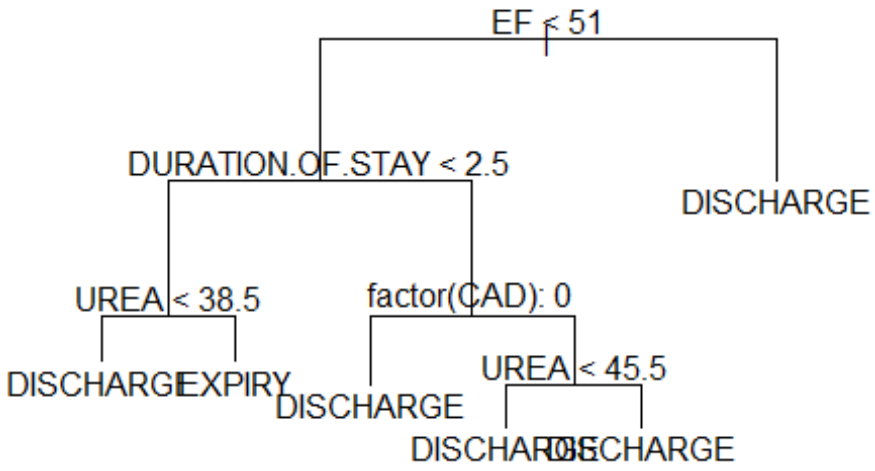
In summary, For Discharge the prediction accuracy is 0.9879466, for EXPIRY the prediction accuracy is 0.3283582 and for DAMA the prediction accuracy is 0.008064516.

```
library(tree)

# Doing the Classification Tree
OUTCOME_tree.fit<-tree(OUTCOME~AGE+EF+TLC+HB+DURATION.OF.STAY+factor(A
LCOHOL)+factor(SMOKING)+factor(DM)+factor(CAD)+factor(CKD)+PLATELETS+G
LUCOSE+UREA+ STABLE.ANGINA + ACS+ STEMI, data = training)


summary(OUTCOME_tree.fit)
```

```
##
## Classification tree:
## tree(formula = OUTCOME ~ AGE + EF + TLC + HB + DURATION.OF.STAY +
##     factor(ALCOHOL) + factor(SMOKING) + factor(DM) + factor(CAD) +
##     factor(CKD) + PLATELETS + GLUCOSE + UREA + STABLE.ANGINA +
```

```
##     ACS + STEMI, data = training)
## Variables actually used in tree construction:
## [1] "EF"               "DURATION.OF.STAY" "UREA"             "facto
r(CAD)"
## Number of terminal nodes:  6
## Residual mean deviance:  0.6386 = 4814 / 7538
## Misclassification error rate: 0.09451 = 713 / 7544
```

```
#Plotting the tree
plot(OUTCOME_tree.fit)
text(OUTCOME_tree.fit ,pretty =0)
```



Let us check the probability in each terminal node.

```
# Check the nodes of the tree
OUTCOME_tree.fit
```

```
## node), split, n, deviance, yval, (yprob)
##       * denotes terminal node
##
##  1) root 7544 6079.0 DISCHARGE ( 0.0608431 0.0426829 0.8964740 )
##    2) EF < 51 4872 4874.0 DISCHARGE ( 0.0938013 0.0484401 0.8577586
)
##      4) DURATION.OF.STAY < 2.5 520 1025.0 DISCHARGE ( 0.3307692 0.1
461538 0.5230769 )
```

```
##        8) UREA < 38.5 272  407.2 DISCHARGE ( 0.1250000 0.1323529 0.
7426471 ) *
##        9) UREA > 38.5 248  484.8 EXPIRY ( 0.5564516 0.1612903 0.282
2581 ) *
##      5) DURATION.OF.STAY > 2.5 4352 3454.0 DISCHARGE ( 0.0654871 0.
0367647 0.8977482 )
##       10) factor(CAD): 0 1055 1386.0 DISCHARGE ( 0.1611374 0.059715
6 0.7791469 ) *
##       11) factor(CAD): 1 3297 1866.0 DISCHARGE ( 0.0348802 0.029420
7 0.9356991 )
##         22) UREA < 45.5 2119  696.9 DISCHARGE ( 0.0089665 0.0240680
0.9669655 ) *
##         23) UREA > 45.5 1178 1046.0 DISCHARGE ( 0.0814941 0.0390492
0.8794567 ) *
##    3) EF > 51 2672  792.9 DISCHARGE ( 0.0007485 0.0321856 0.9670659
) *
```

The majority of terminal nodes wih more than 70% for the class indicated by the tree. In addition, the tree model will never predict DAMMA.

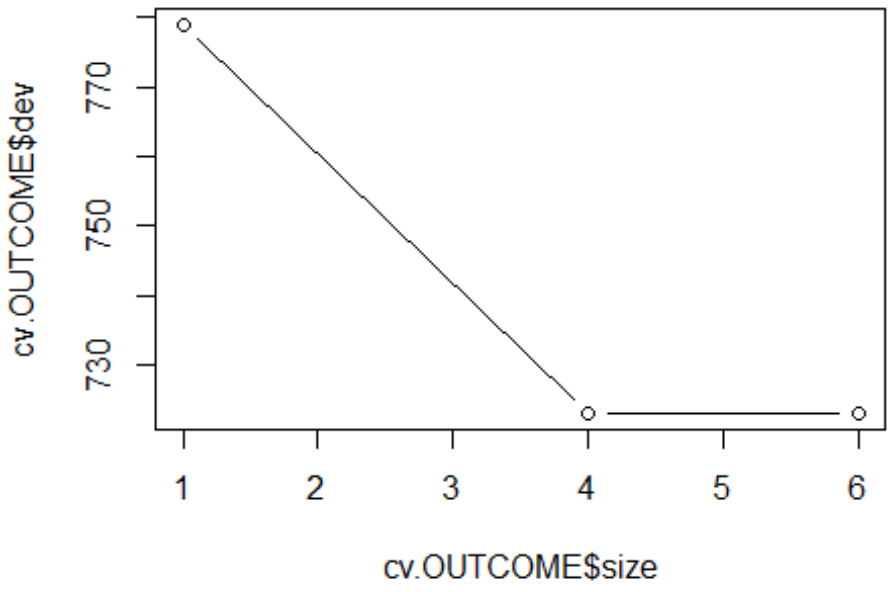
Apply the tree to the test set.

```
# Set a seed for random number generation
set.seed(10)

# Applying the unpruned tree to test part
OUTCOME_tree.pred<-predict(OUTCOME_tree.fit,testing,type = "class")
tablePred_OUTCOME=table(OUTCOME_tree.pred,testing$OUTCOME)

tablePred_OUTCOME

##
## OUTCOME_tree.pred DAMA DISCHARGE EXPIRY
##         EXPIRY      14        19     45
##         DAMA         0         0      0
##         DISCHARGE  110      2304     89

#Misclassification Rate
1-(2304+45+0)/(14+19+45+110+2304+89)

## [1] 0.08988764
```

The overall misclassification rate for OUTCOME with Classification Tree is 0.08988764 (8.99%). However, the model will never predict DAMMA.

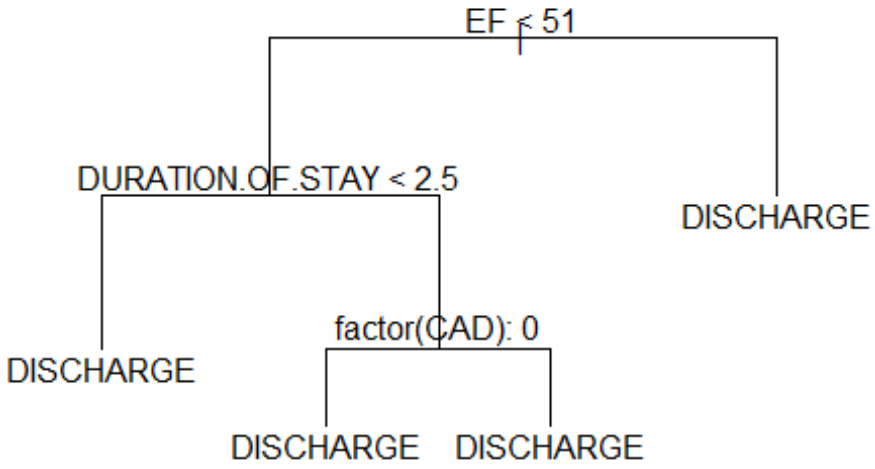
As the tree indicated several terminal nodes, let us try to prune this.

```
# Set a seed for random number generation as the cv select randomly
set.seed(10)

# Checking the cross-validation error versus the number of Terminal no
des to Prune the tree using FUN
cv.OUTCOME<-cv.tree(OUTCOME_tree.fit, FUN = prune.misclass)
plot(cv.OUTCOME$size, cv.OUTCOME$dev,type="b")
```



Based on the cv.error, it looks like the cv.error with 4 terminal nodes is similar to 6. Based on this, let prune the tree to 4 terminal nodes.

```
# Prune the tree
prune.OUTCOME=prune.tree(OUTCOME_tree.fit,best=4)
plot(prune.OUTCOME)
text(prune.OUTCOME,pretty=0)
```




```
# Check the nodes of the Prune tree
prune.OUTCOME
```

```
## node), split, n, deviance, yval, (yprob)
##       * denotes terminal node
##
##  1) root 7544 6079.0 DISCHARGE ( 0.0608431 0.0426829 0.8964740 )
##    2) EF < 51 4872 4874.0 DISCHARGE ( 0.0938013 0.0484401 0.8577586
)
##      4) DURATION.OF.STAY < 2.5 520 1025.0 DISCHARGE ( 0.3307692 0.1
461538 0.5230769 ) *
##      5) DURATION.OF.STAY > 2.5 4352 3454.0 DISCHARGE ( 0.0654871 0.
0367647 0.8977482 )
##       10) factor(CAD): 0 1055 1386.0 DISCHARGE ( 0.1611374 0.059715
6 0.7791469 ) *
##       11) factor(CAD): 1 3297 1866.0 DISCHARGE ( 0.0348802 0.029420
7 0.9356991 ) *
##    3) EF > 51 2672  792.9 DISCHARGE ( 0.0007485 0.0321856 0.9670659
) *
```

The pruned tree just indicates DISCHARGE for OUTCOME. Based on the probabilities in the terminal node, the majority of wrong predictions should happens at the terminal node of: EF < 51 and DURATION.OF.STAY < 2.5.

```r
# Set a seed for random number generation
set.seed(10)

# Applying the unpruned tree to test part
OUTCOME_Prune_tree.pred<-predict(prune.OUTCOME,testing,type = "class")
tablePred_OUTCOME=table(OUTCOME_Prune_tree.pred,testing$OUTCOME)

tablePred_OUTCOME

##
## OUTCOME_Prune_tree.pred DAMA DISCHARGE EXPIRY
##                  EXPIRY       0         0      0
##                  DAMA         0         0      0
##                  DISCHARGE  124      2323    134

#Misclassification Rate
1-(2323)/(124+2323+134)

## [1] 0.09996126
```

The overall misclassification rate for OUTCOME with the Pruned Classification Tree is 0.09996126 (10%). However, the model will only predict DISCHARGE.

PART VI -CONCLUSIONS AND RECOMMENDATIONS

Cluster sampling was not a candidate for our dataset based on the columns available in the cleaned dataset. SRS and stratified sampling were utilized, and SRS provided better accuracy when comparing the sampling means with the population mean. Moreover, SSB was very small compared to SSW which adds another point against stratified sampling being a good candidate for this dataset.

Analysis of independence using Table Contingency helped to select categorical exploratory variables that would be independent among them, but at the same time dependent on the response variable.

Based on the analysis done for HEART.FAILURE, the relevant and independent explanatory variables which influence on this indicate by our analysis were: o Quantitative: AGE, GLUCOSE, HB, TLC, CREATININE, UREA and EF o Qualitative: GENDER, RAISED.CARDIAC.ENZYMES and PRIOR.CMP

However, the statistical learning models (except LDA) are possible to predict with almost the same accuracy using only EF and UREA.

Regarding the best statistical model to predict HEART FAILURE, the LOGISTIC REGRESSION without cross validation, in other words, using stratified sampling 75% of the total population as train set, indicated the best performance in prediction the training part with misclassification rate of 23.9%, following by Classification Tree without cross validation as well (misclassification of 24.5%). However, QDA model was the one that predicted more correctly the patients with

heart failure, whereas LDA (even with only relevant and independent categorical explanatory variables due to no normal distribution of any quantitative variable) which predicted better the patients with no heart failure.

In terms of building the model with 10-fold stratified cross-validation, LDA (only the qualitative variables) had the best performance among all the statistical learning methods with the misclassification rate of 25.3%.

It is important to mention that the normality tests (both Shapiro-Will and Kolmogorov-Smirnov) indicated that none of the quantitative variables has normal distribution, and consequently they were not used in LDA modelling seeing that this statistical learning method requires normal distribution of explanatory variables.

Comparing the HEART.FAILURE model and prediction on this project with the one provided by the professor [Ref.5], the final cleaned datasets are totally different between these two projects, for example on this project will have more than 10000 units, whereas the previous work approximately 300 units. In addition, the majority of the explanatory variables used in the previous project were totally different from what was used on our project, for example in the previous project just one categorical variable was available to be used as explanatory variable.

While analysis of different model of acute kidney injury (AKI) and comparing model results, the Logistic Regression model without stratified sampling indicated the lowest or best misclassification rate of 1.03% among all the generated models to predict AKI. However, we did not observe significant differences with logistic regression with 10 k-fold stratified cross validation and classification tree with and without cross validation.

Knowing that the linear model can only explain 7% of the “DURATION.OF.STAY” response function and that the Regression Tree had similar Residual Standard Error as the Linear Regression, we can conclude that the Regression Tree and Linear Regression are not able to predict the “DURATION.OF.STAY” at the hospital properly. Also, the low R-squared value of the Linear Regression made it difficult to pursue interaction terms or even higher order relationship. Therefore, predicting the “DURATION.OF.STAY” at the hospital may best be served by other machine learning algorithms that are best suited to the dataset.

The models for OUTCOME (3 class of response possible: DISCHARGE, EXPIRY and DAMA) indicated that Multinomial Regression was the only one that could predict all possible class with overall misclassification rate of 9.3%. Classification Tree without prunning (6 terminal nodes) can only provide prediction for DISCHARGE and EXPIRY with overall misclassification rate of 8.99 %. Prunning the tree to 4 terminal nodes just indicates DISCHARGE for any input in the model, and the misclassification rate obtained for this was 10%. Based on this, werecmmend the built Multinomial Regression model to predict OUTCOME.