FINAL PROJECT REPORT
ON
HOSPITAL READMISSION

COURSE: ITCS – 6162 – 092

UNC CHARLOTTE

TEAM MEMBERS:

BALA VENKAT RAM BALANTRAPU	800966649
KISHORE RAJ ETHIRAJ	800987054

TABLE OF CONTENTS

Abstract

On a national average, about 20% of the patients throughout the United States are being re-admitted to the hospital within 30 days of discharge. In 2015, 2610 hospitals were heavily penalized for excessive re-admission rates. 25 % of the total annual cost is being spent on preventable re-admissions which can be completely avoided by proper patient care management.

It is increasingly identified that the management of hyperglycemia in the hospitalized patient has a significant bearing on outcome, in terms of both morbidity and mortality. This recognition has led to the development of formalized protocols in the intensive care unit (ICU) setting with rigorous glucose targets in many institutions.

The objective of the project is to develop a model of readmission risk that doctors can consider when determining when to discharge a patient. The primary focus is on readmissions among their diabetic patients, and a dataset (10kDiabetes.csv) describing this patient population has been provided. The present analysis of a large clinical database was undertaken to examine historical patterns of diabetes care in patients with diabetes admitted in a US hospital and to inform future directions to improve patient safety. We hypothesize that the measurement of HbAlc is associated with reduction in re-admission rates in individuals admitted in hospital.

We used the given dataset and applied correlation techniques and models based on machine learning algorithms to predict the most influential factors affecting patient re-admission and additional insights that can strengthen the readmission risk model.

1. DATA CLEANING

The columns without missing values in the dataset are: - Gender, Age, Time in hospital,Numberoflabprocedures,Numberofprocedures,Numberofmedications,Number of outpatient, Number of emergency, Number of inpatient, Number of diagnosis, Max glucose serum, Alc result, Metamorphin, Readmitted, Acetohexamide,Troglitazone,Examide,Citoglipton,Glimepiride.PioglitazoneMetformin.Rosiglitazone, Metformin.Pioglitazone.

The below table provides an overview of the data cleaning methods used on the dataset for the variables with missing values. The column variable consists of all the variables in the dataset. The column cleaned specifies whether the variable was cleaned or not. The value "**yes**" means the missing values of the variable were cleaned and imputed. The value "**no**" means the missing values were not cleaned. The column "imputation method describes the method used for imputation. We used two methods of imputation.

1. KNN: - Imputed the missing values using the value of the K nearest neighbors.
2. Deleted: - The columns which are removed because they were not considered significant (>95% missing value or not effecting the output variable).

Variable	Cleaned	Imputation Method
Race	Yes	KNN
Weight	No	Deleted
Discharge_disposition_id	Yes	KNN
Admission_type_id	Yes	KNN
Payer_code	No	Deleted
Medical_specialist	Yes	KNN
Dia_1	Yes	KNN
Dia_2	Yes	KNN
Dia_3	Yes	KNN
Dia1_dsc	Yes	KNN
Dia2_dsc	Yes	KNN
Dia3_dsc	Yes	KNN

Table 1 Cleaning methods for variables with missing values

2. MAIN TASK

Problem Statement: To predict the reason for re-admission of patients.
Operations performed : Correlation Analysis, PCA, Regression Analysis, Model building
Language used: R Language

a) Correlation Analysis:

We have two different types of variables in the given data - numeric variables and categorical variables. For the numeric variables, we performed correlation analysis whereas for the categorical variables we performed the chi-sq test.

Numeric Variables:

time_in_hospital,num_lab_procedures,num_procedures,num_medications,number_outpatients,number_emergency,number_inpatients,number_diagnoses

Corr Plot:

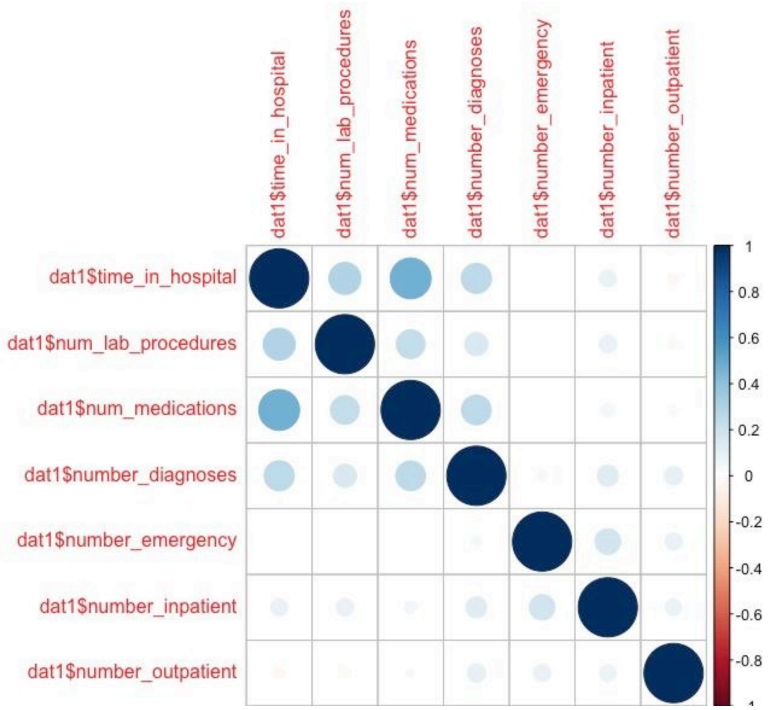


Figure 1: Correlation plot

Interpretation:
The above corr plot is a representation of the correlation between each of the numeric variables in the dataset. The denser the shade and bigger the size of the circle, the higher the relationship between the variables. Based on the correlation analysis, it is observed that all the numeric variables are positively correlated i.e there is a positive relationship between every pair of numeric variables. It is also observed that the variables Time_in_hospital and num_procedures have a very high correlation with readmitted data.

b) **Chi-Sq Test**
We performed the chi-sq test on the categorical variables and below are the chi-sq values and p values for each of the variables.

Categorical Variables	ChiSq Values	P Values
Race	31.407	0.000002527
Age	86.968	6.598E-15
Admission_type_id	18.850	0.002
Discharge_disposition_id	226.325	6.5E-37
Admission_source_id	111.910	6E-20
Medical_speciality	118.670	1.12E-17
Repaglinide	14.504	0.002
pioglitazone	30.993	0.000000852
Insulin	31.670	6.136E-07
Change	25.859	0.000000367
diabeticsMed	28.031	0.000000119
Diag_1_desc	665.336	5.012E-07
Diag_2_desc	557.938	0.00002252
Diag_3_desc	632.034	1.371E-07

Table 2: Chi-sq test results

Interpretation:

From the above table, we notice that all the variables have significant p value and are correlated with readmitted data. Diag_1_desc, Diag_2_desc, Diag_3_desc and discharge_disposition_id have very high chiSq values which indicate that they are highly correlated to readmission data when compared to other variables.

c) REGRESSION ANALYSIS:

We use regression analysis for estimating the relationships among the variables to analyze which variables influence the readmission rate the most and can be further used in building the readmission risk model.

Linear Regression analysis:

We performed the linear regression analysis for numeric variables. Below are the results indicating the parameters of linear regression.

Coefficients	Estimate	Standard Error	t value	Pr(>|t|)
(Intercept)	0.267	0.013	19.875	<2e-16***
time_in_hospital	0.002	0.001	1.113	0.257
Num_lab_procedures	0.001	0.000	5.510	3.86e-08***
Num_procedures	-0.010	0.003	-3.232	0.001***
Num_medications	0.001	0.000	2.612	0.009**
Num_emergency	0.031	0.007	4.227	2.39e-05***
Num_inpatient	0.101	0.005	17.670	<2e-16***

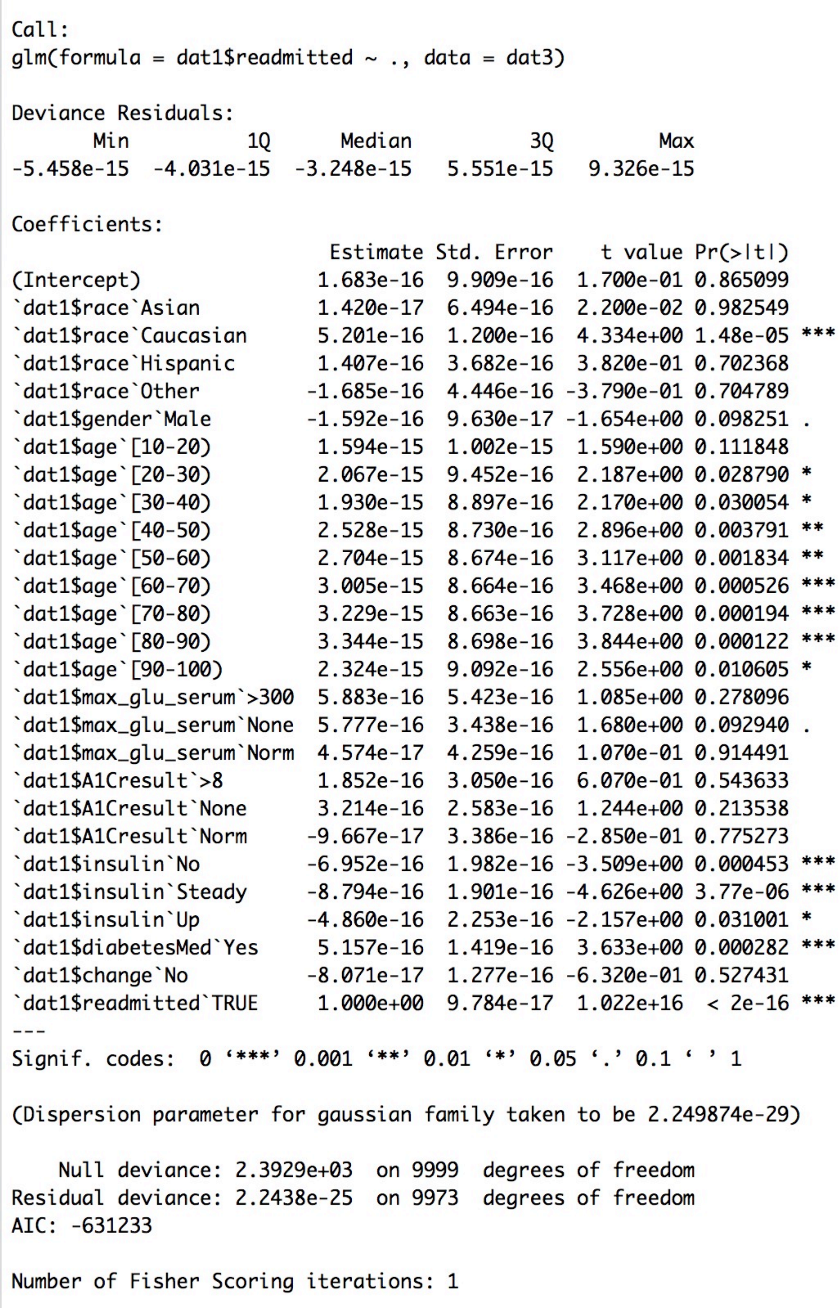
Table 3: Linear Regression Analysis results

Interpretation:

The Pr values for all the variables are less than 0.05. This indicates that all the variables are significant enough to be considered for further model building.

Logistic Regression analysis:

We performed the Logistic Regression analysis for categorical variables. The below figure shows the results of the analysis. It is inferred from below Pr values that all the variables are significant enough to be considered for further model building (Pr values < 0.05).



```
Call:
glm(formula = dat1$readmitted ~ ., data = dat3)

Deviance Residuals:
       Min          1Q      Median          3Q         Max
-5.458e-15  -4.031e-15  -3.248e-15   5.551e-15   9.326e-15

Coefficients:
                           Estimate Std. Error    t value Pr(>|t|)
(Intercept)               1.683e-16  9.909e-16  1.700e-01 0.865099
`dat1$race`Asian          1.420e-17  6.494e-16  2.200e-02 0.982549
`dat1$race`Caucasian      5.201e-16  1.200e-16  4.334e+00 1.48e-05 ***
`dat1$race`Hispanic       1.407e-16  3.682e-16  3.820e-01 0.702368
`dat1$race`Other         -1.685e-16  4.446e-16 -3.790e-01 0.704789
`dat1$gender`Male        -1.592e-16  9.630e-17 -1.654e+00 0.098251 .
`dat1$age`[10-20)         1.594e-15  1.002e-15  1.590e+00 0.111848
`dat1$age`[20-30)         2.067e-15  9.452e-16  2.187e+00 0.028790 *
`dat1$age`[30-40)         1.930e-15  8.897e-16  2.170e+00 0.030054 *
`dat1$age`[40-50)         2.528e-15  8.730e-16  2.896e+00 0.003791 **
`dat1$age`[50-60)         2.704e-15  8.674e-16  3.117e+00 0.001834 **
`dat1$age`[60-70)         3.005e-15  8.664e-16  3.468e+00 0.000526 ***
`dat1$age`[70-80)         3.229e-15  8.663e-16  3.728e+00 0.000194 ***
`dat1$age`[80-90)         3.344e-15  8.698e-16  3.844e+00 0.000122 ***
`dat1$age`[90-100)        2.324e-15  9.092e-16  2.556e+00 0.010605 *
`dat1$max_glu_serum`>300  5.883e-16  5.423e-16  1.085e+00 0.278096
`dat1$max_glu_serum`None  5.777e-16  3.438e-16  1.680e+00 0.092940 .
`dat1$max_glu_serum`Norm  4.574e-17  4.259e-16  1.070e-01 0.914491
`dat1$A1Cresult`>8        1.852e-16  3.050e-16  6.070e-01 0.543633
`dat1$A1Cresult`None      3.214e-16  2.583e-16  1.244e+00 0.213538
`dat1$A1Cresult`Norm     -9.667e-17  3.386e-16 -2.850e-01 0.775273
`dat1$insulin`No         -6.952e-16  1.982e-16 -3.509e+00 0.000453 ***
`dat1$insulin`Steady     -8.794e-16  1.901e-16 -4.626e+00 3.77e-06 ***
`dat1$insulin`Up         -4.860e-16  2.253e-16 -2.157e+00 0.031001 *
`dat1$diabetesMed`Yes     5.157e-16  1.419e-16  3.633e+00 0.000282 ***
`dat1$change`No          -8.071e-17  1.277e-16 -6.320e-01 0.527431
`dat1$readmitted`TRUE     1.000e+00  9.784e-17  1.022e+16  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for gaussian family taken to be 2.249874e-29)

    Null deviance: 2.3929e+03  on 9999  degrees of freedom
Residual deviance: 2.2438e-25  on 9973  degrees of freedom
AIC: -631233

Number of Fisher Scoring iterations: 1
```

Figure 3: Results of Logistic regression

We considered two cases for Logistic regression analysis. The case where the readmission is False and the case where readmission is True. We derived the variables and their respective values that affect the condition of readmission being true and readmission being false. The below figures depict the results.

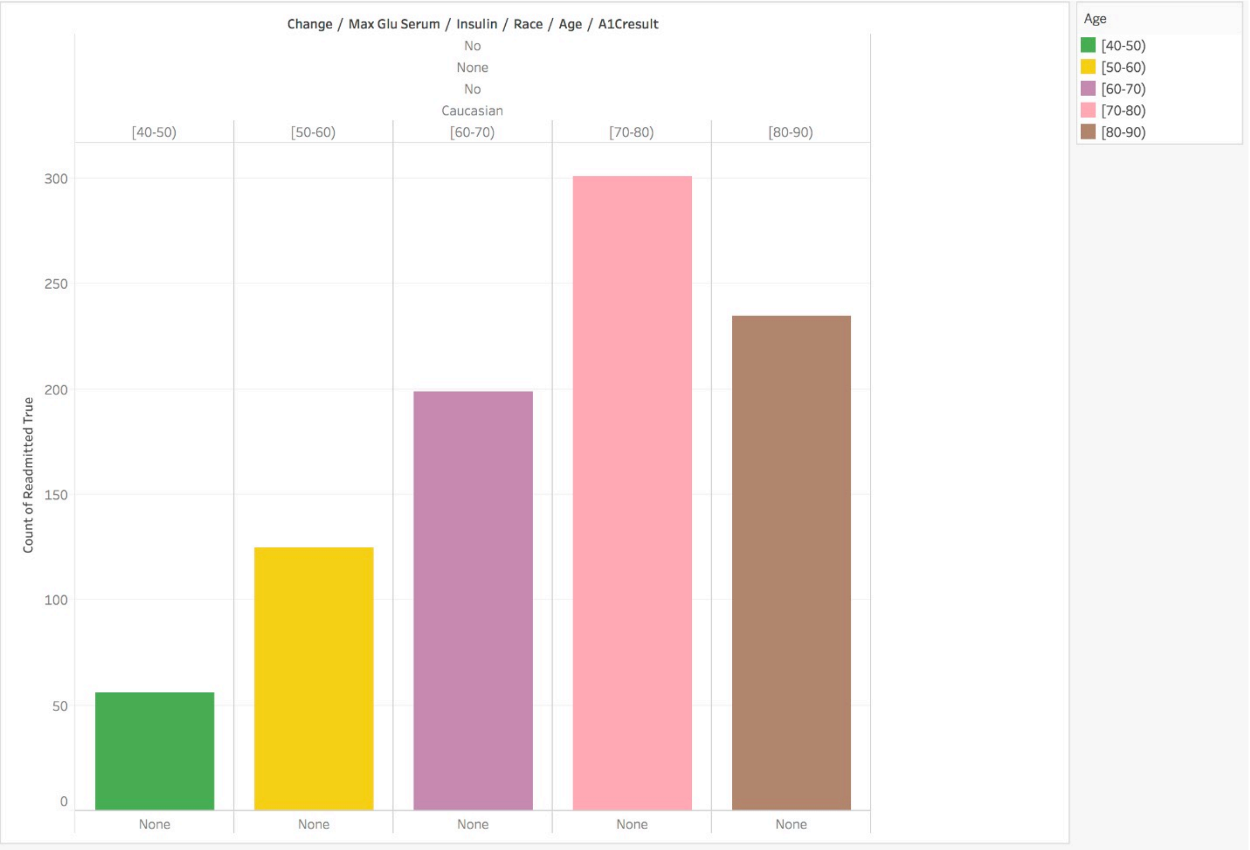


Figure 5: Results for count of Readmitted true

Interpretation:

Based on the above graph, for the conditions of No Change, No Max Glucose Serum, No insulin and race Caucasian, the highest count of A1c Result for Readmitted condition holding true is between the age group of 70 – 80.

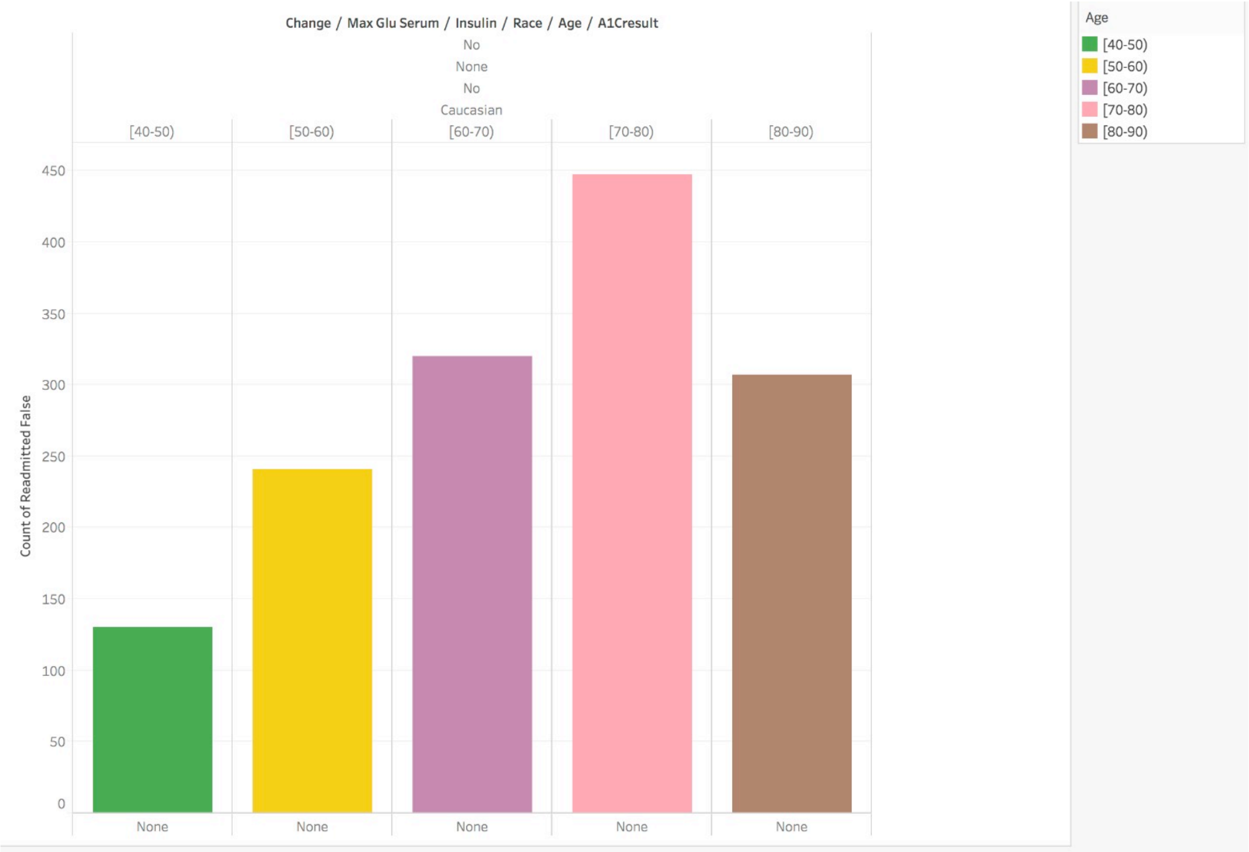


Figure 5: Results for count of Readmitted false

Interpretation:

Based on the above graph, for the conditions of No Change, No Max Glucose Serum, No insulin and race Caucasian, the highest count of A1c Result for Readmitted condition holding true is between the age group of 70 – 80

d) PRINCIPAL COMPONENT ANALYSIS

Principal component analysis (PCA) is a statistical procedure that uses an orthogonal transformation to convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables called principal components (or sometimes, principal modes of variation).

We performed PCA on our dataset in order to map similar variables together and reduce the dimensionality of the data. We chose the number of factors as 3 for the analysis. Below are the results of the PCA analysis:

Columns	PC1	PC2	PC3	h2	u2	Com
Num_lab_procedures	0.52	0.15	-0.54	0.58	0.42	2.1
Time_in_hospital	0.76	0.04	-0.17	0.61	0.39	1.1
Num_procedures	0.59	-0.26	0.49	0.66	0.34	2.4
Num_medications	0.83	-0.04	0.19	0.73	0.27	1.1
Number_outpatient	-0.02	0.42	0.6	0.54	0.46	1.8
Number_emergency	0.01	0.68	0.17	0.49	0.51	1.1
Number_inpatient	0.12	0.73	-0.19	0.58	0.42	1.2

Table 4: Results of PCA

	PC1	**PC2**	**PC3**
SS Loadings	1.89	1.26	1.02
Proportion var	0.27	0.18	0.15
Cumulative var	0.27	0.45	0.60
Proportion explained	0.45	0.30	0.25
Cumulative proportion	0.45	0.75	1.00

Table 5: Results of PCA

Interpretation:

From the above results, we noticed that factor 1 has the highest variance.time_in_hospital, number_of_medications come under PC1 and can be reduced to high_inpatient_number as both variables are directly proportional. From PC2 we see that number_of_inpatient, number_of_emergency fall under this factor and can be reduced to a column high_inpatient_risky. From PC3 we see that number_of_procedures, number_of_outpatient fall under this factor and can be reduced to a column freq_tests.

PCA Plot:

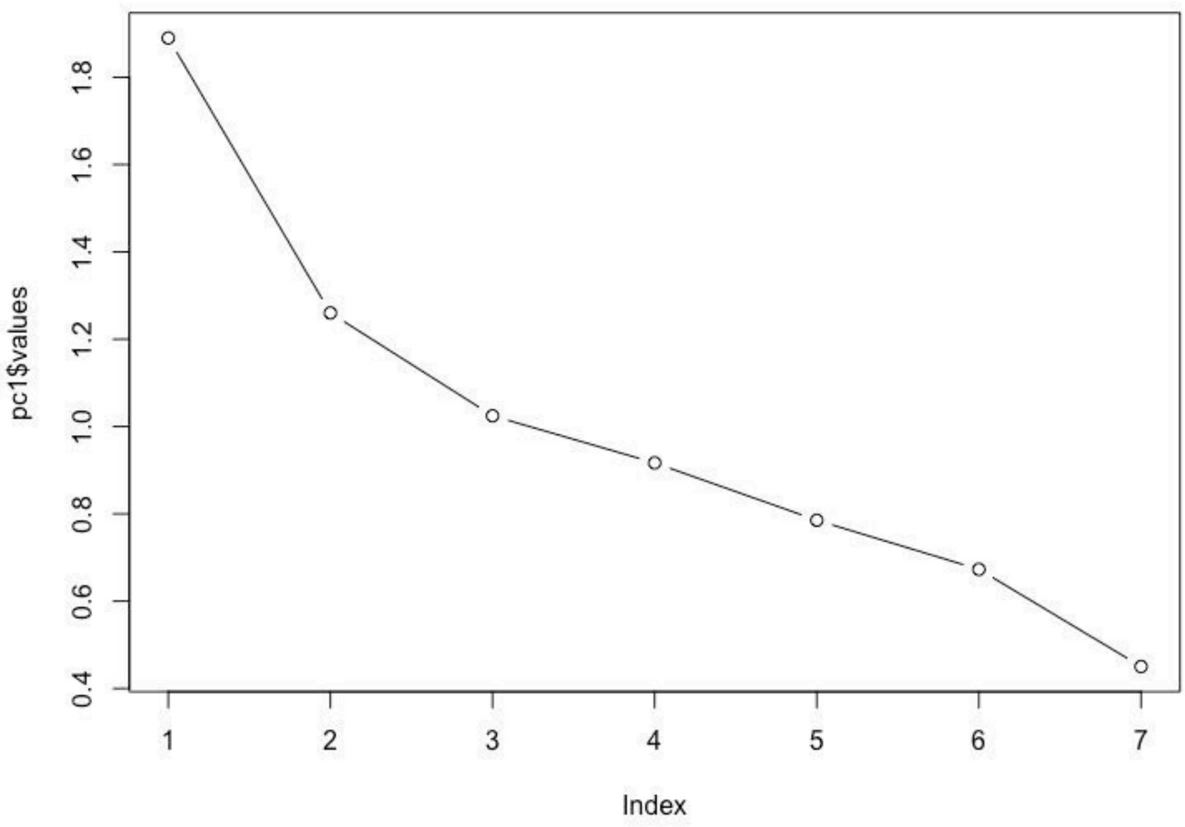


Figure 6: PCA Plot

e) **Model Building**: In the Model building phase we worked on building various models before we concluded on our best fit model. The top 4 models which best fit the data were:

1) Decision Tree Model
2) Random Forest Model
3) Naive Bayesian Model
4) Logistic Regression Model

Data Specifications:

- Training data: 80% of the data set
- Test data : 20% of the data set
- Predictors : race, gender, age, admission_type_iddischarge_disposition_id, admission_source_id,time_in_hospital,medical_speciality,num_lab_procedures,num_procedures,num_medications,num_outpatient,num_emergency,num_inpatient, diag_1,diag_2, diag_3,number_diagnosis,max_glu_serum, a1cresult,insulin,change, diabetes_medication, readmitted.

Model Name: Decision Tree Model

a) **Outcome :** Re-Admitted
Predictors:
race,gender,age,admission_type_id,time_in_hospital, num_lab_procedures,num_procedures,num_medications,num_outpatient,num_emergency,num_inpatient,number_diagnosis,max_glu_serum,A1Cresult, change,diabetesMed,readmitted.
Results :
Confusion Matrix:

Predicted	False	True
False	1085	578
True	102	195

Error : 34.69%
Accuracy : 65.31%

Decision Tree:

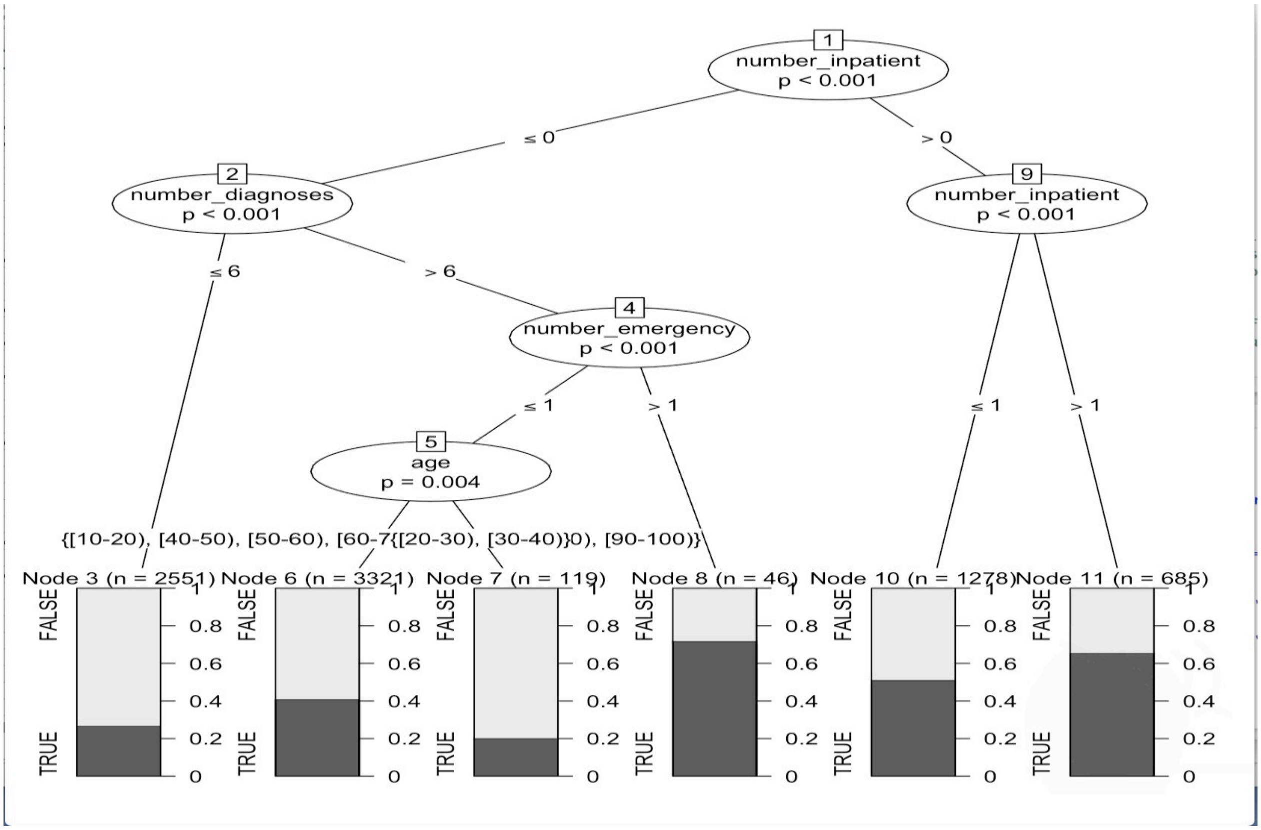


Figure 7: Decision Tree

Interpretation :

From the above model we can conclude that using a decision tree would help us predict readmission with an accuracy of 65.31% and the variables which play a crucial role in this model are number of inpatients, number of diagnosis and age.

Model Name : Random Forest Model
Specifications : Number of Trees:500

Predictors:
num_lab_procedures,medical_speciality,num_medications,age,time_in_hospital,number_diagnoses,discharge_disposition_id,number_inpatient,insulin,admission_source_id,admission_type_id,race,A1Cresult,number_outpatient,gender,metformin,glyburide,glipizide,number_emergency,change,max_glu_serum,diabetesMed,rosiglitazone,pioglitazone,glimepiride,repaglinide,nateglinide,glyburide.metformin, acarbose

Results :

Confusion Matrix:

Predicted	False	True	Class.Error
False	4685	5	0.001066
True	3136	16	0.994923

Error : 38.05%

Accuracy : 61.95%

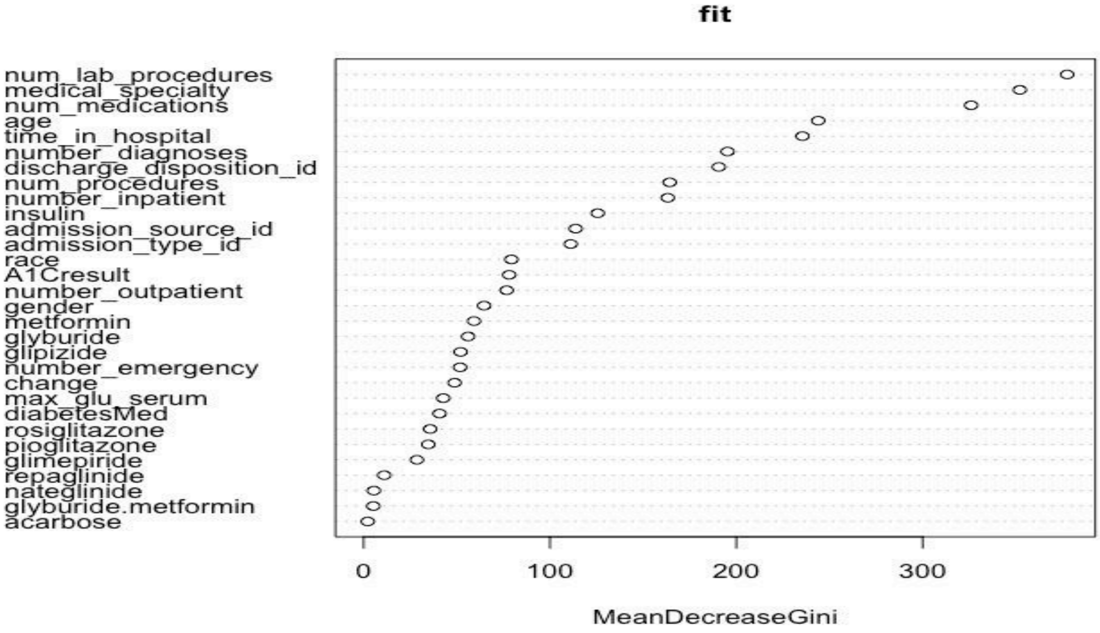


Figure 8: Random Forest

Interpretation :

From the above model we can conclude that using the random forest would help us predict readmission with an accuracy of 61.95% and the variables which play a crucial role in this model are number of lab procedures, medical specialty, num of medications and age.

Model Name : Naïve Bayesian Predictors: num_lab_procedures,medical_speciality,num_medications,age,time_in_hospital,number_diagnoses,discharge_disposition_id,number_inpatient,insulin,admission_source_id,admission_type_id,race,A1Cresult,number_outpatient,gender,metformin,glyburide,glipizide,number_emergency,change,max_glu_serum,diabetesMed,

Results:

Confusion Matrix:

Predicted	**False**	**True**
False	1094	601
True	126	179

Error : 36.35%

Accuracy : 63.65%

Interpretation :

The Naïve Bayesian model helped us predict the readmission rate with an accuracy level of 63.65%

Model Name : Logistic Regression Predictors: num_lab_procedures,medical_speciality,num_medications,age,time_in_hospital,number_diagnoses,discharge_disposition_id,number_inpatient,insulin,admission_source_id,admission_type_id,race,A1Cresult,number_outpatient,gender,metformin,glyburide,glipizide,number_emergency,change,max_glu_serum,diabetesMed

Results :

Confusion Matrix:

Predicted	**False**	**True**
False	1032	521
True	188	259

Error	33.45%
Accuracy	66.55%
P-Value[Acc>NIR]	0.0005781
Kappa	0.1928
Mcnemer's Test P-Value	<2.2e-16
Sensitivity	0.8459
Specificity	0.3321
Pos Pred Value	0.6645
Neg Pred Value	0.5794

ROC Plot:

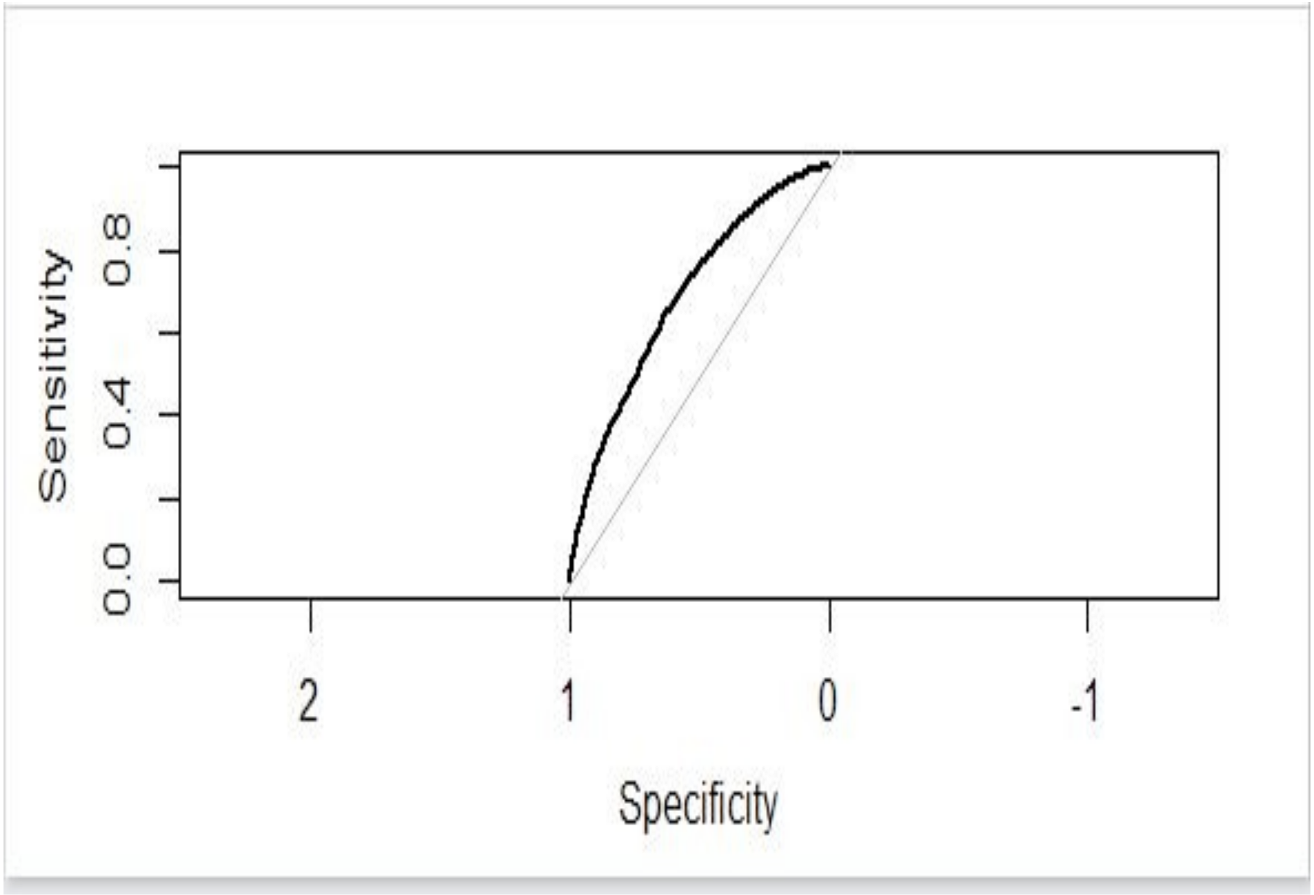


Figure 9: ROC Plot

Interpretation :

The best model which fit the data in all 4 models we tried till now is the logistic regression model with an accuracy of 66.55%. The ROC curve for the logistic regression shows that it is the best fit model when compared to others.

3. CONCLUSION FOR MAIN TASK

From the above results and interpretations, we can conclude that the main reasons for the patient being readmission are A1Cresult, Insulin, Max glu-serum tests not being taken during the time readmission assuming that previous results which the patient have are valid. The model that best fits our data is the logistic regression model with an accuracy of approximately 66%. Therefore we conclude by saying readmission would be prevented at a high rate if we can cautiously take the test of the patients and note the new reading before admission without depending on previous data.

4. SUB TASK 1

Problem Statement: Predicting the time in hospital
Operations performed: Outlier detection
Language used: R programming.

We have used Outlier detection techniques to detect the outliers for each of the variables to analyze the influence/variation on the readmission rate due to the outliers. The variables used are Num_procedures, num_diagnoses, num_emergency, num_inpatient, num_outpatient, num_medications. The output variable is time_in_hospital.

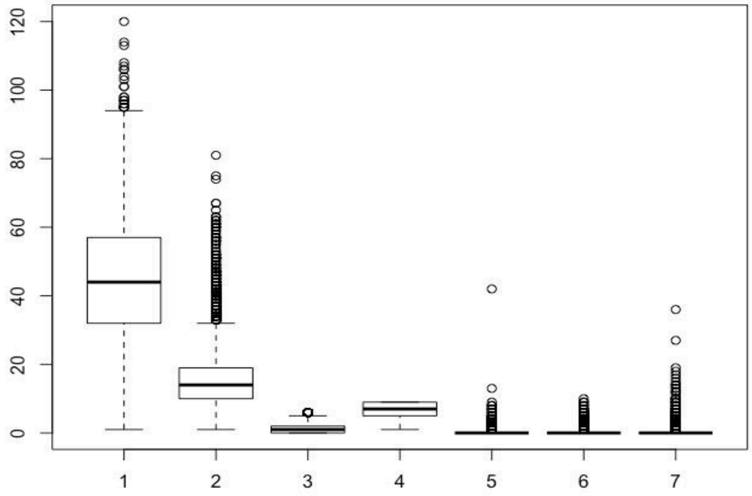


Figure 10: Boxplot for Outliers

Interpretation: As we can conclude that all the numerical variables have significant number of outliers which would affect the output variable i.e, Time_in_hospital. Further we choose to perform a Linear regression model on the data to predict the time in hospital from future.

Linear Modeling:

Coefficients	**Estimate**	**Std Error**	**T value**	**Pr(>|t|)**
(Intercept)	**-0.3630**	**0.1041**	**-3.487**	**0.000491*****
d$num_lab_procedures	**0.0268**	**0.0013**	**19.588**	**< 2e-16 *****
d$num_medications	**0.1425**	**0.0035**	**40.137**	**<2e-16*****
d$num_procedures	**0.0207**	**0.0166**	**1.241**	**0.2147**
d$number_diagnoses	**0.1975**	**0.0133**	**14.845**	**<2e-16*****
d$number_emergency	**-0.0791**	**0.0403**	**-1.963**	**0.04965***
d$number_inpatient	**0.1403**	**0.0309**	**4.528**	**6.01e-06*****
d$number_outpatient	**-0.1399**	**0.0231**	**-6.043**	**1.57e-09*****

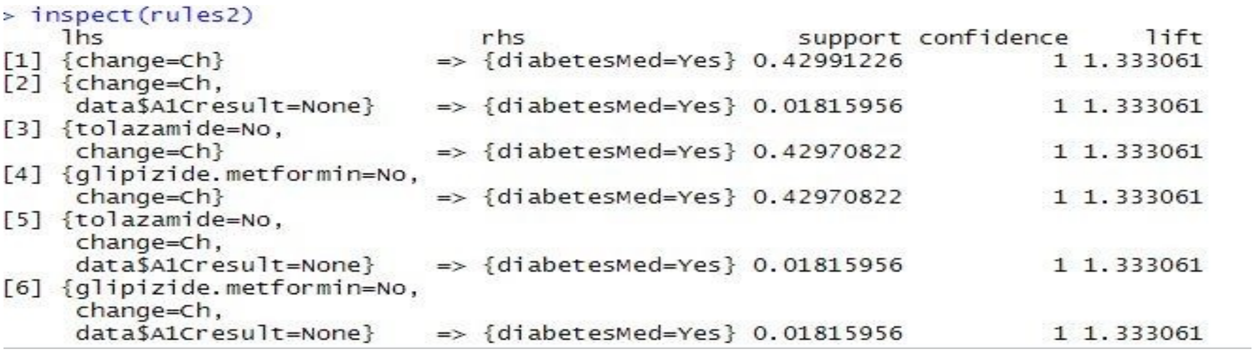
Interpretation: - From the above results of Linear modelling we can conclude that all the predictors have played a significant role in the model apart from number_emergency.

Error: - 29.25% **Accuracy: -** 69.75%

5. SUBTASK 2

Problem Statement: Prediction of the changes in diabetics medication that should be prescribed to the patient basing on his Insulin, A1Cresult and few significant medicines.
Operations performed: Apriori Algorithm
Language used: R programming

Results: -



```
> inspect(rules2)
    lhs                            rhs                 support confidence     lift
[1] {change=Ch}                 => {diabetesMed=Yes} 0.42991226          1 1.333061
[2] {change=Ch,
     data$A1Cresult=None}       => {diabetesMed=Yes} 0.01815956          1 1.333061
[3] {tolazamide=No,
     change=Ch}                 => {diabetesMed=Yes} 0.42970822          1 1.333061
[4] {glipizide.metformin=No,
     change=Ch}                 => {diabetesMed=Yes} 0.42970822          1 1.333061
[5] {tolazamide=No,
     change=Ch,
     data$A1Cresult=None}       => {diabetesMed=Yes} 0.01815956          1 1.333061
[6] {glipizide.metformin=No,
     change=Ch,
     data$A1Cresult=None}       => {diabetesMed=Yes} 0.01815956          1 1.333061
```

Interpretation: -

We have used Pattern matching techniques to detect the patterns for each of the variables to analyze the influence/variation on the readmission rate due to the group of clusters detected by pattern maching. The Apriori algorithm generated 6 rules which were sorted by lift values. The variables used are Num_procedures, num_diagnoses, num_emergency, num_inpatient, A1Cresult, tolazamide, glipizide,metaformin, num_outpatient, num_medications, time_in_hospital. The output variable is Diabetesmed.

Therefore, from next time when a new patient arrives we can use these following rules and conclude if he should be prescribed a change in medication or no.

APPENDIX

R code

```
install.packages("corrplot")
library(corrplot)
install.packages('nnet')
library(nnet)
install.packages("mlogit")
library(mlogit)
```

#Setting the path from where we read the file

```
setwd("/Users/balavenkatrambalantrapu/Desktop")
```

#Reading the file after imputing the missing values and removing non significant columns

```
dat1<-read.csv("final.csv", header=TRUE, strip.white = TRUE,na.strings = c(""," ",".","NA"))
summary(dat1)
```

# View the data initially before we perform our analysis

```
View(dat1)
```

#Combining all the numerical variables to perform the correlation analysis

```
dat2<-cbind.data.frame(dat1$time_in_hospital,dat1$num_lab_procedures,dat1$num_medications, dat1$number_diagnoses, dat1$number_emergency, dat1$number_inpatient,dat1$number_outpatient,dat1$readmitted)
```

#Plotting the corrplot

```
corrplot(cor(dat2),"circle","full")
d<-pairs(dat2)
```

#Performing Chi-Sq test on the categorical variables

```
ch<-chisq.test(dat3)
summary(ch)
```

#Performing Logistic Regression on all the categorical variables

```
dat3<-cbind.data.frame(dat1$race,dat1$gender,dat1$age, dat1$max_glu_serum,dat1$A1Cresult,dat1$insulin,dat1$diabetesMed,dat1$change,dat1$readmitted)
typeof(dat3$`dat1$max_glu_serum`)
x<-na.omit(dat3)
for(i in 1:9){
  x[,i]<-as.factor(x[,i])
}
dat4<-glm(dat1$readmitted~., data=dat3)
summary(dat4)
```

#Performing Linear Modeling on all the numerical variables

```
dat5<-lm(readmitted~time_in_hospital+num_lab_procedures+num_procedures+num_medications+number_emergency+number_inpatient , data=dat1)
summary(dat5)
```

#Performing Principal component Analysis

```
install.packages("stats")
library(stats)
pc1 <-  principal(dat1, nfactors = 3, rotate="none")
summary(pc1)
```

```
plot(pc1$values,type="b")

#Model Building
#Partitioning of data into training and Testing data
testdata<-sample_frac(f, 0.2)
sid <-as.numeric(rownames(testdata))
traindata <- f[-sid,]
View(traindata)

#Decision Tree Model
library(ROCR)
library(rpart)
fit <- ctree(traindata$readmitted~., data =traindata, control=
ctree_control(mincriterion = .99, minsplit = 1000))
varimp(w)
info.gain.ctree(fit)
varImpPlot(fit)
summary(fit)

#Predict Output
predicted= predict(fit, traindata, type='prob')
roc<-performance(predicted, "tpr", "fpr")
plot(roc,)
w<-weights(fit)
tab<-table(predicted,testdata$readmitted)
```

```
tab


predicted
error = mean(predicted != testdata$readmitted)
error
dev.off()
#Plot the output
plot(fit)
plot(fit, type="simple",          # no terminal plots
    inner_panel=node_inner(fit,
                       abbreviate = F,          # short variable names
                       pval = T,                # no p-values
                       id = T),                 # no id of node
    terminal_panel=node_terminal(fit,
                            abbreviate =T,
                            digits = 1,                # few digits on numbers
                            fill = c("grey"),          # make box white not grey
                            id = FALSE)
)
```

#RandomForest

```
install.packages("randomForest")
library(randomForest)
f$dia
which( colnames(f)==c("diag_3_desc","diag_2_desc" ))
f<-subset(f[,-c(37,38,39)])
```

```
fit <- randomForest(traindata$readmitted~., data =traindata, controls =
ctree_control(mincriterion = .99, minsplit = 1500), proximity=TRUE)
cforest(traindata$readmitted~., data =traindata, controls =
cforest_control(mincriterion = .99, minsplit = 500))
varImpPlot(fit)
summary(fit)
#Predict Output
predicted= predict(fit,testdata)
tab<-table(predicted,testdata$readmitted)
tab
randomForest::getTree(fit)
randomForest::partialPlot(traindata$readmitted~., data =traindata, controls =
ctree_control(mincriterion = .95, minsplit = 500))
predicted
error = mean(predicted != testdata$readmitted)
error
#Plot the output
MDSplot(fit, traindata$readmitted)


## Naive Bayesian
library(e1071)
classifier<-naiveBayes(traindata[,1:48], traindata[,49])
classifier<-naiveBayes(change ~.,data=traindata1,usekernel=T)
plot(classifier)
table(predict(classifier, testdata[,-20]), testdata[,20])test_predictions =
predict(classifier, testdata, type = "class")
```

```
test_error = sum(test_predictions != testdata$readmitted)/nrow(testdata)

test_error
```

##Logistic

```
install.packages("mlogit")

log <- glm(readmitted~.,data=data,family = binomial(logit))

prob=predict(log,type=c("response"))

data$prob=prob

install.packages("pROC")

library(pROC)

g <- roc(readmitted ~ prob, data = data)

plot(g)
```

##Additional Task 1 & 2

```
d<-read.csv("ram.csv", header=TRUE, strip.white = TRUE,na.strings = c(""," ",".","NA"))

d1<-d$num_medications

d1<-cbind(d$num_procedures,d$number_diagnoses, d$number_emergency, d$number_inpatient, d$number_outpatient)

d1[,-3]

d2<-d1[,1] as.matrix(d1)

install.packages("mvoutlier")

library(mvoutlier)

uni.plot(d)

boxplot(d)
```

```
View(d)

install.packages("mlogit")

library(mlogit)

g<-
lm(d$time_in_hospital~d$num_lab_procedures+d$num_medications+d$num_proc
edures+d$number_diagnoses+
d$number_emergenc+d$number_inpatient+d$number_outpatient,
family="binomial")

summary(g)

dist=daisy(d, stand=TRUE,metric=c("gower"))

library(HSAUR)

library(cluster)

library(daisy)

out2=outliers.ranking(dist,method="sizeDiff",meth="ward.D")

as.table(head(out2$rank.outliers,100))

as.table(head(out2$prob.outliers,100))


patternData <- subset(d[,c(3,4,5,19,33,36,37,40,38)])

install.packages("arulesViz")


library(arulesViz)

patternPatients <- as(patternData,"transactions")

rules<-apriori(patternPatients,

          parameter=list(minlen=2,support=0.1,confidence=0.6),

          appearance=list (rhs=c("readmitted=TRUE",

                         "readmitted=FALSE"),
                    default="lhs"))


patternData <- subset(d[,c(3,4,5,19,33,36,37,40,38)])
install.packages("arulesViz")
library(arulesViz)
patternPatients <- as(patternData,"transactions")
rules<-apriori(patternPatients,
         parameter=list(minlen=2,support=0.1,confidence=0.6),
         appearance=list (rhs=c("readmitted=TRUE",
                       "readmitted=FALSE"),
                    default="lhs"))
```