Report on Diabetes Prediction Modelling

1. Abstract:

Disease diagnosis is a major problem area for researchers for a long time. To accurately diagnosis a disease is of prime concern for a doctor. There are various conventional methods of disease diagnosis, but application of soft computing technique with information technology has given a new dimension to this area. In this particular work one approach has been proposed for the classification of subjects into two classes namely: Diabetic & Non-diabetic.

2. Introduction:

The use of classifier systems in medical diagnosis is increasing gradually. In this paper one classifier technique of decision tree analysis is implemented for the forecasting of Diabetes and concluded with best forecasting techniques which has a maximum accuracy.

3. Objective:

To predict diabetes based on the information given cost of misclassification.

	Predicted Cases		
Actual Cases		Diabetes	No Diabetes
	Diabetes	0	50
	No Diabetes	150	0

4. Data Summary:

The dataset which we use in our work is Diabetic database. All patients are at least 21 years old. The binary response variable takes the values “True‟ or “False” where “True‟ means a positive test for diabetes and “False‟ is a negative test for diabetes. There are 268 (34.9%) cases in class “True” and 500 (65.1%) cases in class “False”.

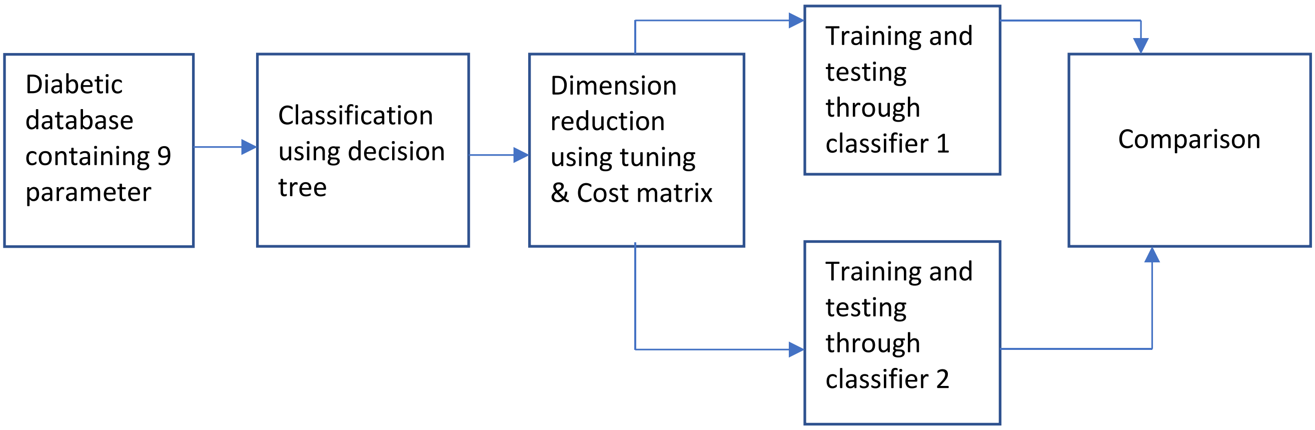
- No. of observations:768
- Total No. of attributes:9

Attributes	**Type**
Insulin	Numeric
Blood Pressure	Numeric
BMI	Numeric

Preganancies	Numeric
Plasma Glucose	Numeric
Triceps Skin Thickness	Numeric
Diabetes Pedigree	Numeric
Age	Numeric
Diabetes	Categorical

4.1 Block diagram:

The overall system block diagram is shown in figure below. In the first part the original diabetic database consisting of the entire 8 feature is shown. This dataset will be utilized for all the classification tasks throughout the study. The entire experimental work can be divided into two major sections:


Diabetic database containing 9 parameter
Classification using decision tree
Dimension reduction using tuning & Cost matrix
Training and testing through classifier 1
Training and testing through classifier 2
Comparison

4.2 Data Preparation:

- Dataset was checked for any missing values or Not Applicable.
- Dataset was divided into two sets containing
  a) Training Set : 700 observations.
  b) Test Set : 67 observations.
- Training data set and test set were fed into 10 different models. K-Fold Cross Validation technique was used as the control parameter for each of them.

5. Results:

- Results of cost of misclassifications were calculated based on the predictions for Type I and Type II errors with various model as mentioned below.

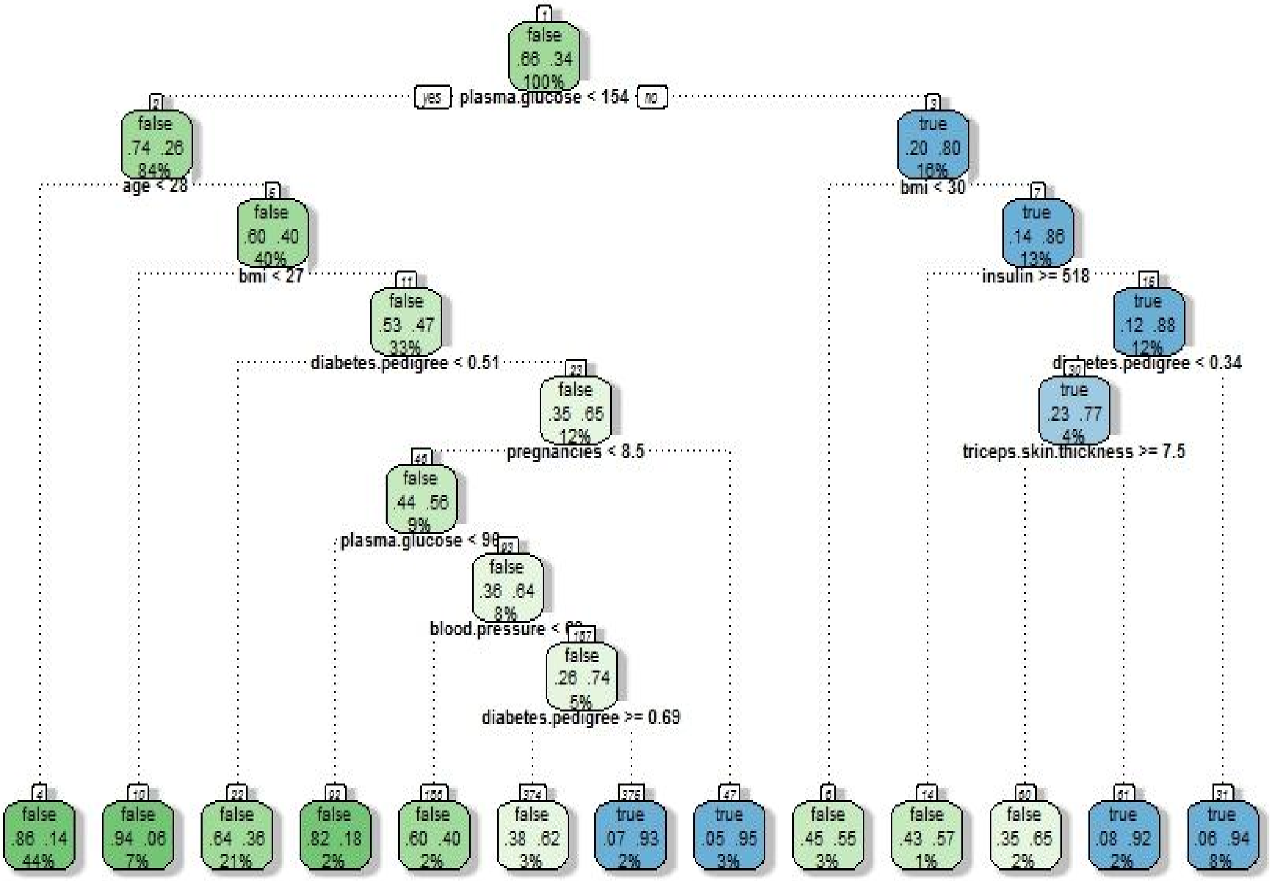
SL NO.	Model	Kappa	Accuracy	CV	Cost
1	rpart	0.4232	0.75	10	950.00
2	C5.0	0.5243	0.7794	10	1250.00
3	J48	0.4892	0.7647	10	1300.00

- Caret (rpart) was used to build the final mode with the following results

5.1 Confusion Matrix:

Caret(rpart)		
Prediction	Reference	
	False	True
True	40	16
False	1	11
Cost of misclassification		950

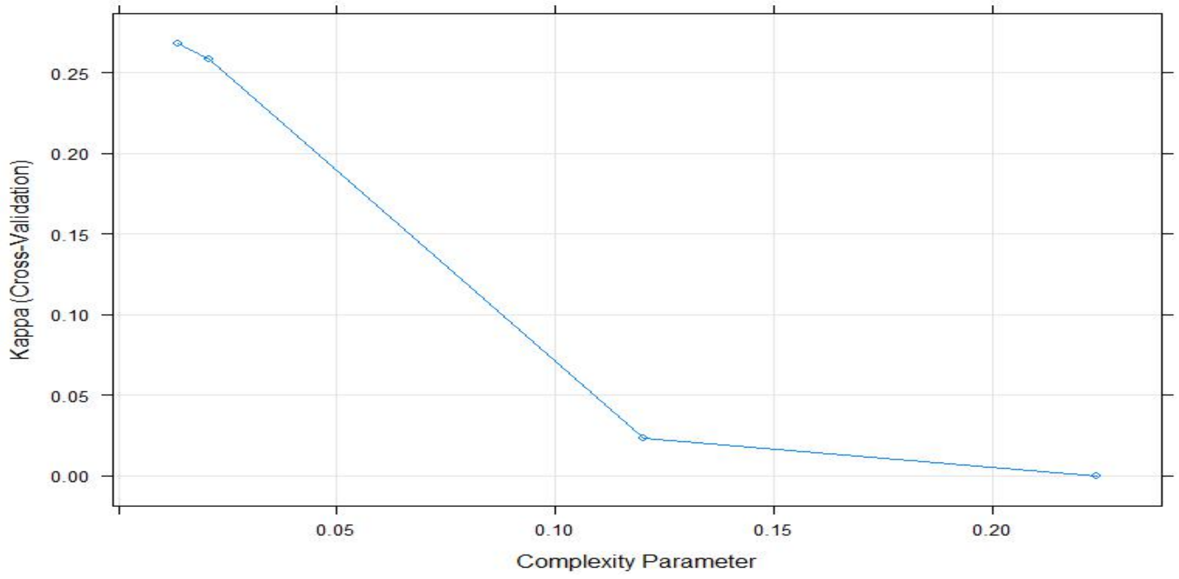
5.2 Decision Tree:



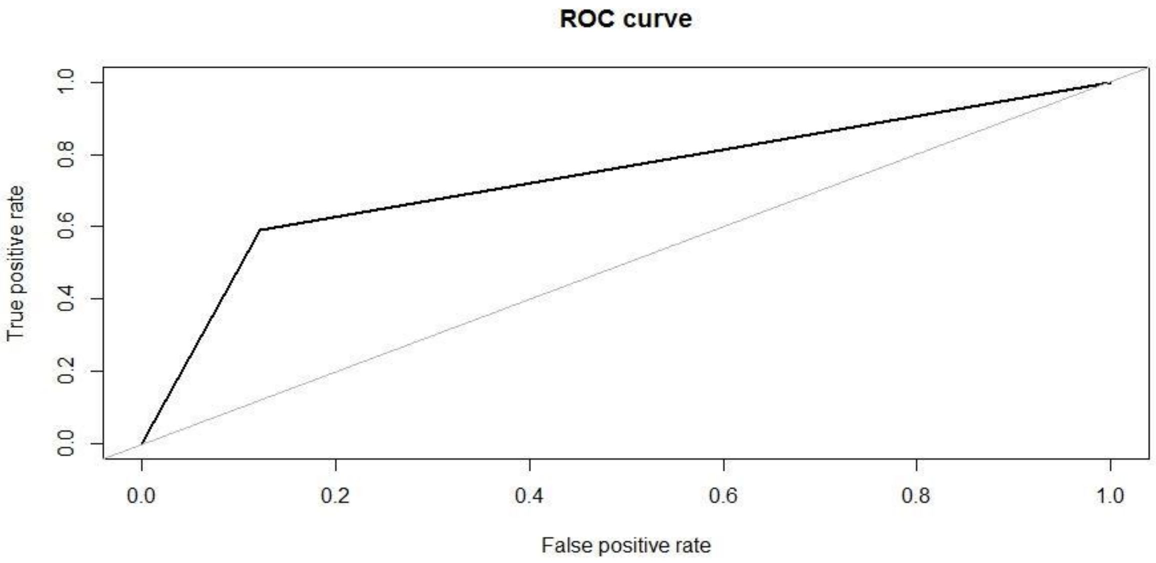
5.3 Tuning Parameters:

The optional parameter for the classification splitting function i.e., the "parms" parameter was passed in the rpart training model. The loss matrix must have zeros on the diagonal and positive off-diagonal elements. The splitting index can be gini or information. The default priors are proportional to the data counts, the losses default to 1, and the split defaults to gini.

Plot of Kappa v/s Complexity parameter



5.4 ROC Curve



With two classes the Receiver Operating Characteristic (ROC) curve can be used to estimate performance using a combination of sensitivity and specificity. The area under the ROC curve is a common metric of performance.

6. Conclusion:

- Model built with Caret(rpart) package gives the minimum cost of misclassification of INR 950.00 with a test data of 67 observations
- No. of Misclassification of Type 1 error in model built with Caret(rpart) is less as compared to other two models.
- No. of Misclassification of Type 2 error in model built with Caret(rpart) is less as compared to other two models

Note:

- Type 2 error: Prediction false (having No Diabetes) and reference true (having Diabetes)
- Type 1 error: Prediction True (having Diabetes) and reference false (having No Diabetes)

7. R Codes:

```
setwd("E:\\Home\\R\\DM1 Assignment")
```

# Library

```
library("rattle")

library("caret")

library("rpart")
```

# Reading the file

```
fulldata <- read.xlsx("CustomDiabetesDataset.xlsx",1)

diabetes_orginial<- diabetes_orginial[,-9]

fulldata<-diabetes_orginial
```

# Splitting the data info

```
rows <- 1:700

train <- fulldata[rows,]

test <- fulldata[-rows,]
```

# Summary Statistic

```
summary(train[,-9])

boxplot(train[,-9])
```

# Checking Correlational table to analyse which variable are highly correlated

```
cor(train[,-9])
```

```
# Define training control

train_control<- trainControl(method="cv",number = 10)


# Cost Matrix

cost_matrix <- matrix(c(0,50,150,0), byrow = T)

# Model

diab_model <- train(diabetes~.,data=train,trControl=train_control,

method="J48",

metric="Kappa",

tuneLength = 4 ),

parms= list(loss = cost_matrix))

varimp(diab_model)

# Tree

rattle::fancyRpartPlot(diab_model$finalModel)

plot(diab_model)


# Pred Test data

pred.diabetes <- predict(diab_model,newdata = test)

confusionMatrix(pred.diabetes,test$diabetes)

table(pred.diabetes,test$diabetes)

# ROC

roc.curve(test$diabetes,pred.diabetes)
```