PREDICTIVE MODELING OF DIABETES RISK FACTORS

Prof . MAGESH TARALA

BUAN 6356 - Business Analytics With R

PROJECT SUBMITTED BY

Jan, Muneeb Ullah
Khan, Raja Mamoon
Tanmayee Ashok Dharam
Vijay Refkin puvvala
Yajjala, Jesse Jackson
Nalli, Gladin

TABLE OF CONTENTS

TOPIC **PAGE NO**

Predictive Modeling of Diabetes Risk Factors

Exploratory Data Analysis

Dataset Summary:
Diabetes dataset has a total of 390 records. It contains 16 Predictors in total, 3 are categorical and 13 are numerical variables. Out of the 13 numerical predictors, 3 are derived attributes. In our analysis, we assessed the performance of classifiers by comparing models that incorporate all predictors with those involving single attributes and derived attributes. Following is the breakdown of all Predictors:

Categorical:
Patient number, Gender and Diabetes class.
Numerical – Single attribute:
Cholesterol, Glucose, HDL Chol, Age, Height, Weight, Systolic BP, Diastolic BP, Waist and Hip
Numerical – Derived attributes:
Chol to HDL ratio, BMI and Waist to Hip ratio.

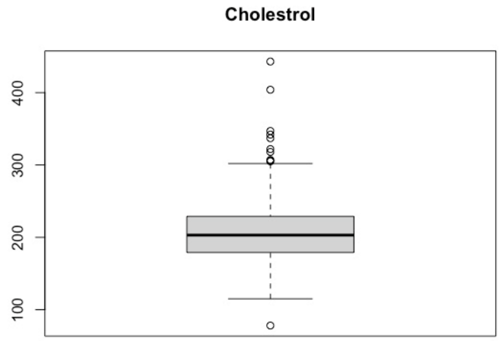
It is worth noting that the dataset has been obtained from a US hospital's directory which makes it an observational study. In other words, it does not portray a true representation of whole population. This observational study restricts us from making inferences to the whole population, but we can make causal inferences.

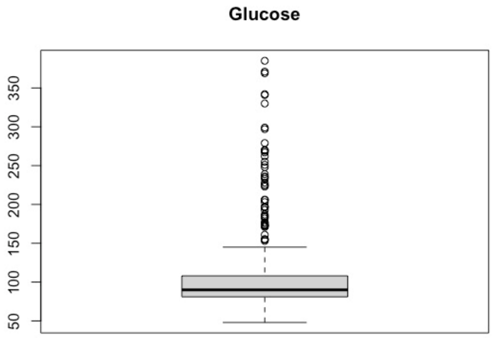
Early Treatment of Dataset:

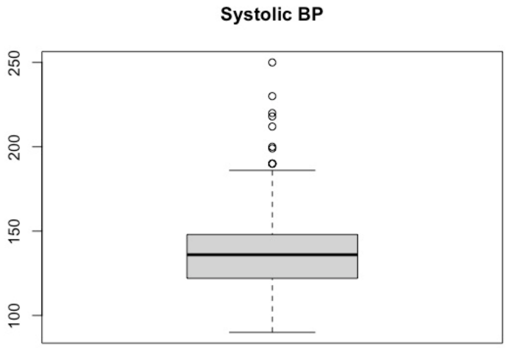
1. Removing Patient column as it does not provide any helpful information.
2. Making data types consistent, converting commas to periods.
3. No missing values – No data imputation required.

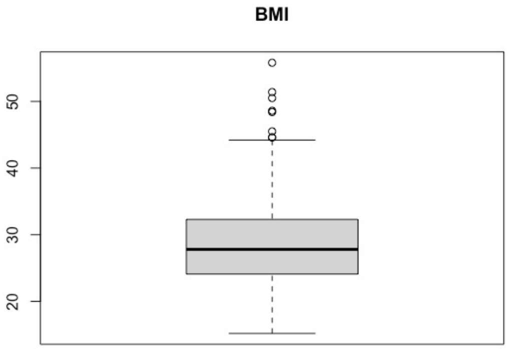
Univariate Analysis:

1. Checking for Outliers

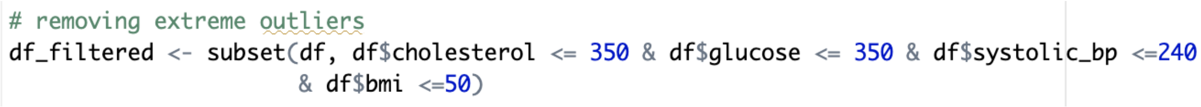








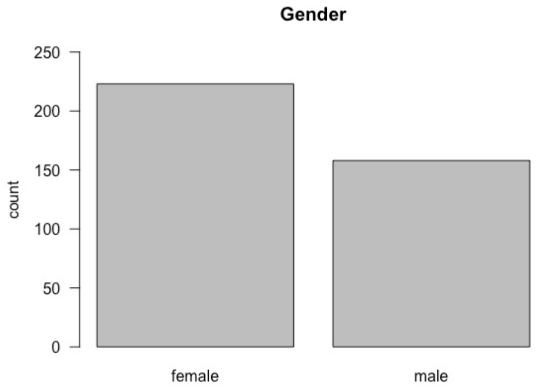
Removed extreme outliers by data filtering.



```
# removing extreme outliers
df_filtered <- subset(df, df$cholesterol <= 350 & df$glucose <= 350 & df$systolic_bp <=240
                      & df$bmi <=50)
```

Gender Classification

Gender	Count
Male	158
Female	223

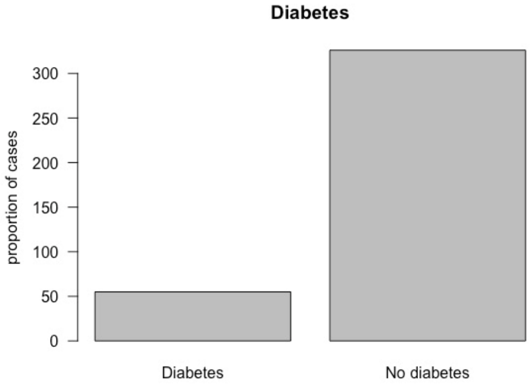


Diabetes Class Proportion:

Diabetes	Proportion
Yes	14.1%
No	83.5%

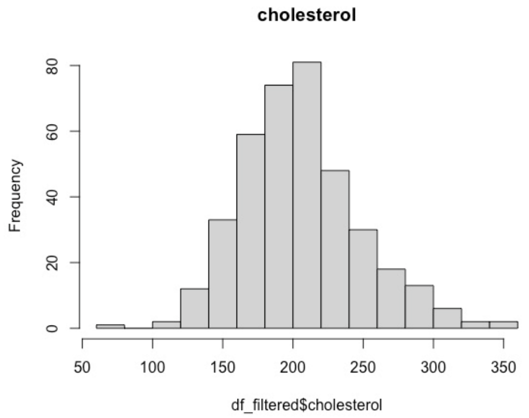
Diabetes Class Numbers:

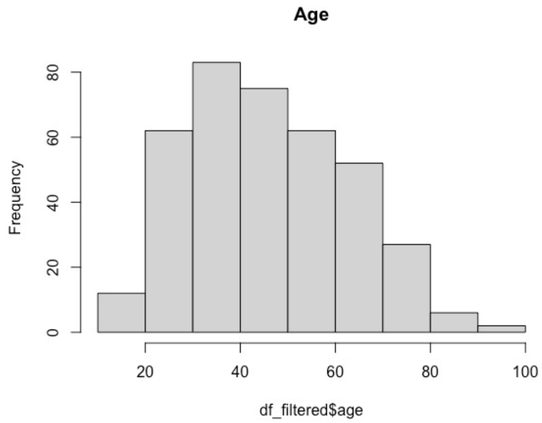
Diabetes	Count
Yes	55
No	326

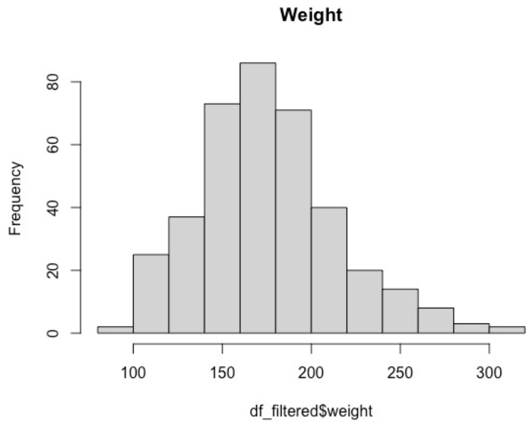


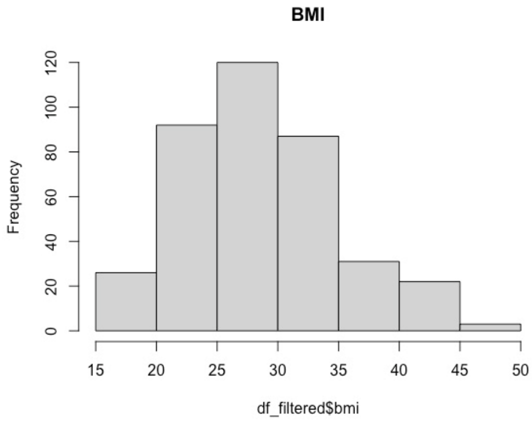
Checking Normality of Predictors

All of the predictors are near normal but not exactly normally distributed. We have used sampling distribution of the sampling mean which is normally distributed to perform statistical analysis. Below are few of the Original Distributions.


cholesterol
Frequency
0
20
40
60
80
50
100
150
200
250
300
350
df_filtered$cholesterol


Age
Frequency
0
20
40
60
80
20
40
60
80
100
df_filtered$age

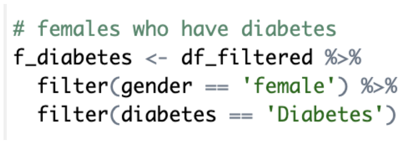

Weight
Frequency
0
20
40
60
80
100
150
200
250
300
df_filtered$weight


BMI
Frequency
0
20
40
60
80
100
120
15
20
25
30
35
40
45
50
df_filtered$bmi

Bivariate Analysis

Diabetes Class based on Gender.

For Females:

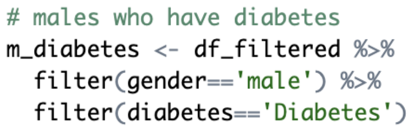


```
# females who have diabetes
f_diabetes <- df_filtered %>%
  filter(gender == 'female') %>%
  filter(diabetes == 'Diabetes')
```

Count	32
Proportion of total women	14.35%

Out of 158 women, 32 have diabetes which is 14.35% of the total women count.

For Males:



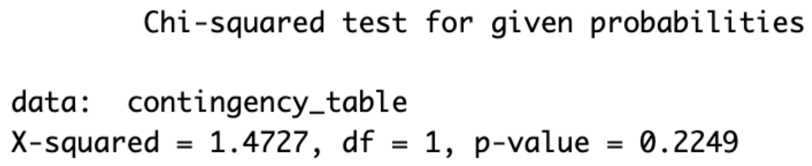
```
# males who have diabetes
m_diabetes <- df_filtered %>%
  filter(gender=='male') %>%
  filter(diabetes=='Diabetes')
```

Count	23
Proportion of total men	14.56%

Out of 158 men, 23 have diabetes which is 14.56% of the total men count.

Performing a Chi Square test to check if the difference in proportions is statistically significant.

H0: Proportion of individuals with diabetes is same for males and females.
H1: Proportion of individuals with diabetes is different for males and females.

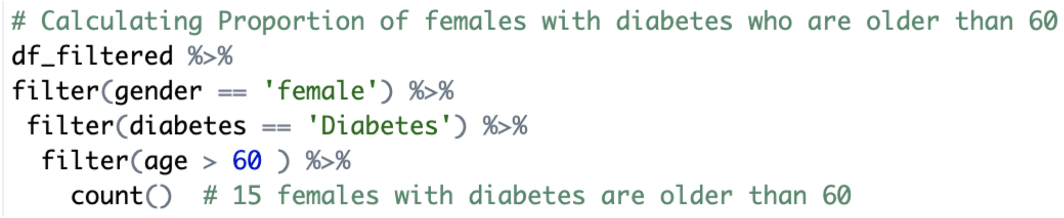


```
                Chi-squared test for given probabilities

        data:  contingency_table
        X-squared = 1.4727, df = 1, p-value = 0.2249
```

As the p-value is greater than the threshold significance level, with 95% confidence we can conclude that Proportion of individuals who have diabetes is same irrespective of gender. In simple terms, Gender plays a very small or no role at all in determining if an individual will have diabetes or not.

Testing the Hypothesis that claims older females are more likely to have diabetes. We are considering individuals who are above the age of 60 as old as 60 falls above the 3rd quartile of the Age predictor.

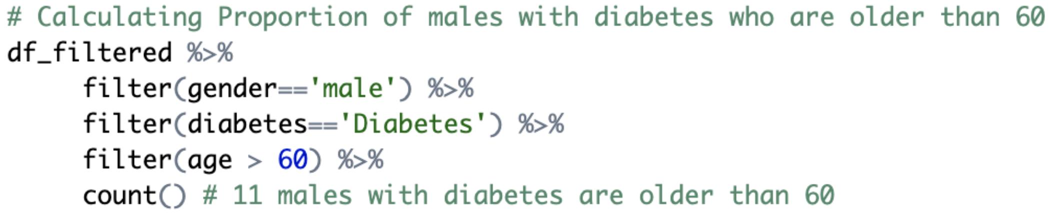
For females:



```
# Calculating Proportion of females with diabetes who are older than 60
df_filtered %>%
filter(gender == 'female') %>%
 filter(diabetes == 'Diabetes') %>%
  filter(age > 60 ) %>%
    count()  # 15 females with diabetes are older than 60
```

15 females with diabetes are older than 60 which is 0.08% of total female with diabetes.

For males:



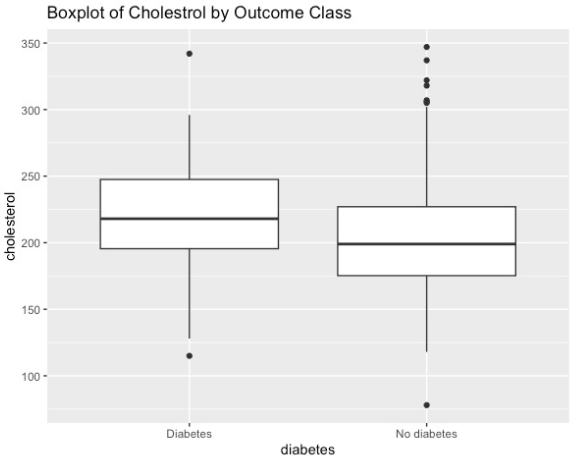
```
# Calculating Proportion of males with diabetes who are older than 60
df_filtered %>%
     filter(gender=='male') %>%
     filter(diabetes=='Diabetes') %>%
     filter(age > 60) %>%
     count() # 11 males with diabetes are older than 60
```

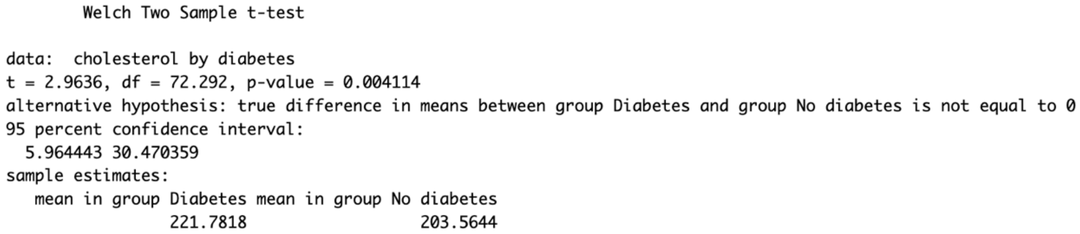
11 males with diabetes are older than 60 which is 0.07% of the total males with diabetes.

From the above test, we can infer that the probability of having diabetes for both males and females is similar. The 0.001% difference in proportions is statistically insignificant.

Correlation of Predictors with the Outcome variable

Boxplot of Cholesterol by outcome class along with the T-test to check its significance:

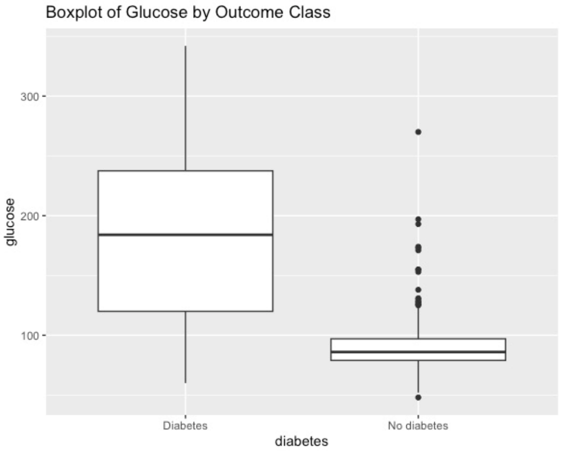


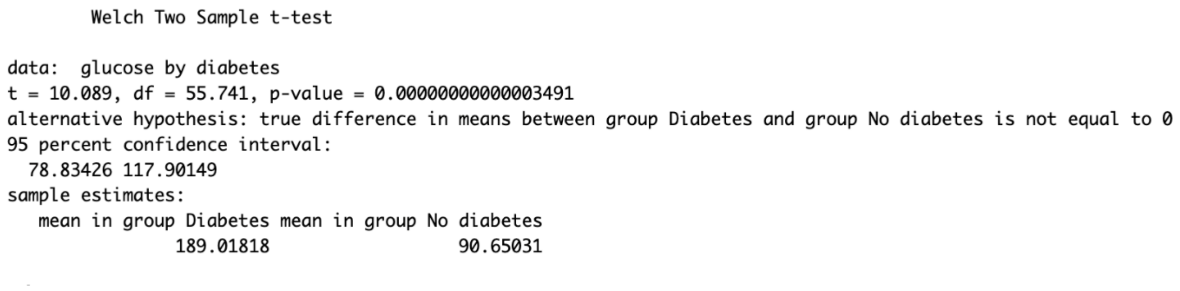


```
        Welch Two Sample t-test

data:  cholesterol by diabetes
t = 2.9636, df = 72.292, p-value = 0.004114
alternative hypothesis: true difference in means between group Diabetes and group No diabetes is not equal to 0
95 percent confidence interval:
  5.964443 30.470359
sample estimates:
   mean in group Diabetes mean in group No diabetes
                 221.7818                  203.5644
```

As the p-value is less than the significance level, we reject the null Hypothesis and conclude that the Cholesterol mean is statistically different in individuals who have diabetes against who don't have diabetes. In simple terms, Cholesterol is an important factor in determining if an individual has diabetes or not.

Boxplot of Glucose by outcome class along with the T-test to check its significance:

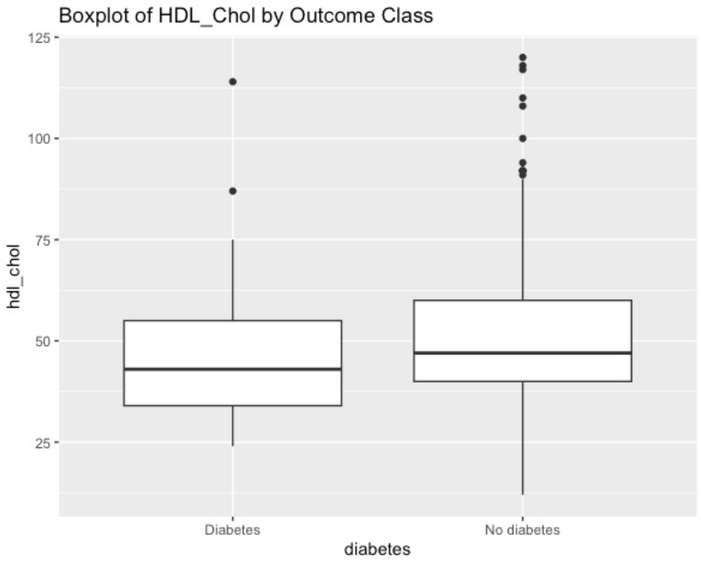


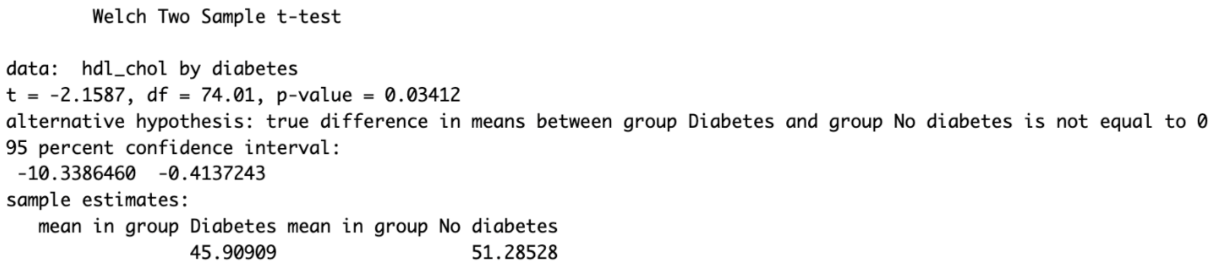


```
        Welch Two Sample t-test

data:  glucose by diabetes
t = 10.089, df = 55.741, p-value = 0.00000000000003491
alternative hypothesis: true difference in means between group Diabetes and group No diabetes is not equal to 0
95 percent confidence interval:
  78.83426 117.90149
sample estimates:
   mean in group Diabetes mean in group No diabetes
               189.01818                 90.65031
```

As the p-value is less than the significance level, we reject the Null Hypothesis and conclude that mean glucose level is statistically different in individuals with diabetes against those who don't have diabetes. In simple terms, Glucose is a very important factor in determining if an individual has diabetes.

Boxplot of HDL_CHOL by outcome class along with the T-test to check its significance:

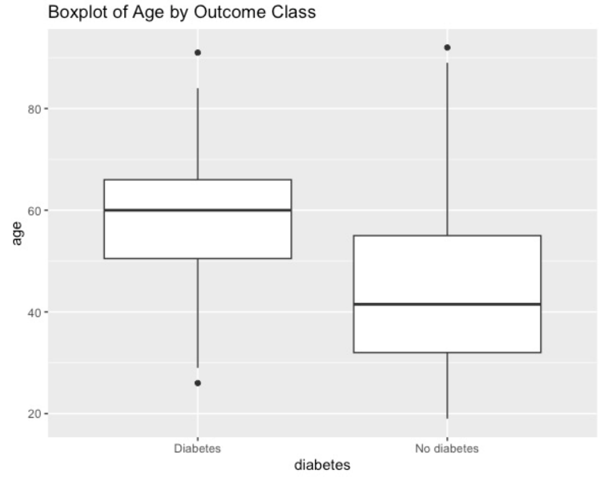


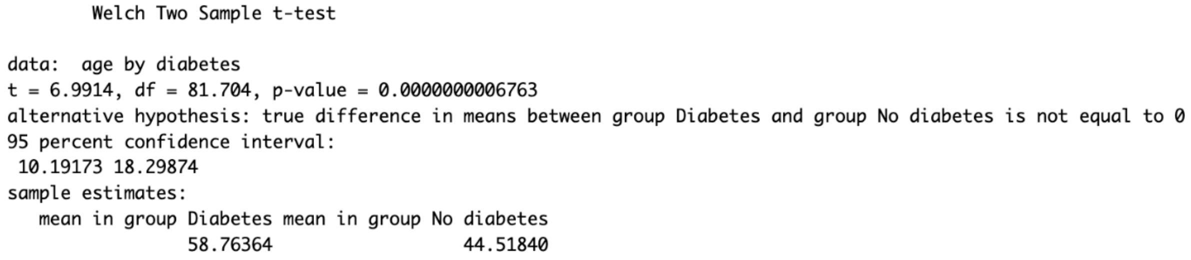


```
       Welch Two Sample t-test

data:  hdl_chol by diabetes
t = -2.1587, df = 74.01, p-value = 0.03412
alternative hypothesis: true difference in means between group Diabetes and group No diabetes is not equal to 0
95 percent confidence interval:
 -10.3386460  -0.4137243
sample estimates:
   mean in group Diabetes mean in group No diabetes
                45.90909                  51.28528
```

As the p-value is less than the significance level, we reject the Null Hypothesis and conclude that mean HDL_chol level is statistically different in individuals with diabetes against those who don't have diabetes. In simple terms, HDL_chol is a very important factor in determining if an individual has diabetes.

Boxplot of AGE by outcome class along with the T-test to check its significance:

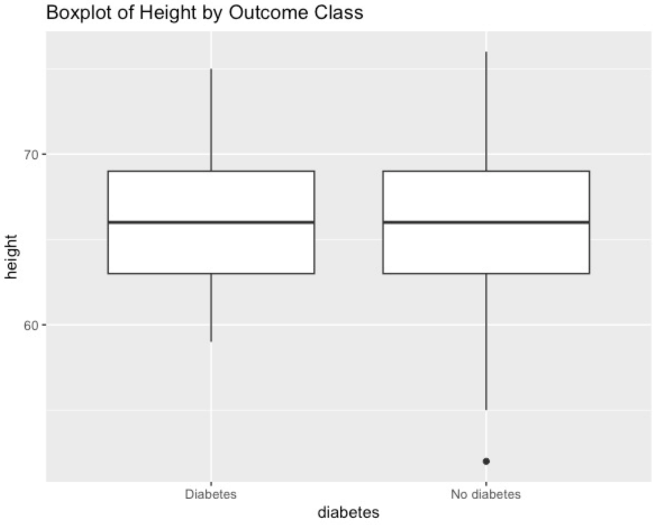


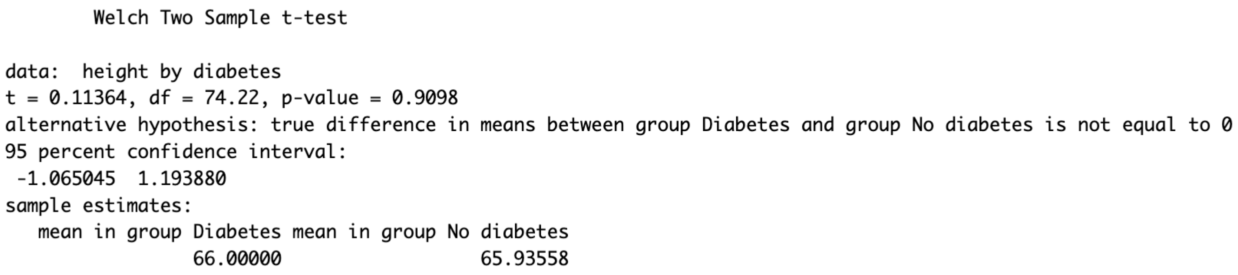


```
        Welch Two Sample t-test

data:  age by diabetes
t = 6.9914, df = 81.704, p-value = 0.0000000006763
alternative hypothesis: true difference in means between group Diabetes and group No diabetes is not equal to 0
95 percent confidence interval:
 10.19173 18.29874
sample estimates:
   mean in group Diabetes mean in group No diabetes
                 58.76364                  44.51840
```

As the p-value is less than the significance level, we reject the Null Hypothesis and conclude that mean Age is statistically different in individuals with diabetes against those who don't have diabetes. In simple terms, AGE is a very important factor in determining if an individual has diabetes.

Boxplot of HDL_CHOL by outcome class along with the T-test to check its significance:

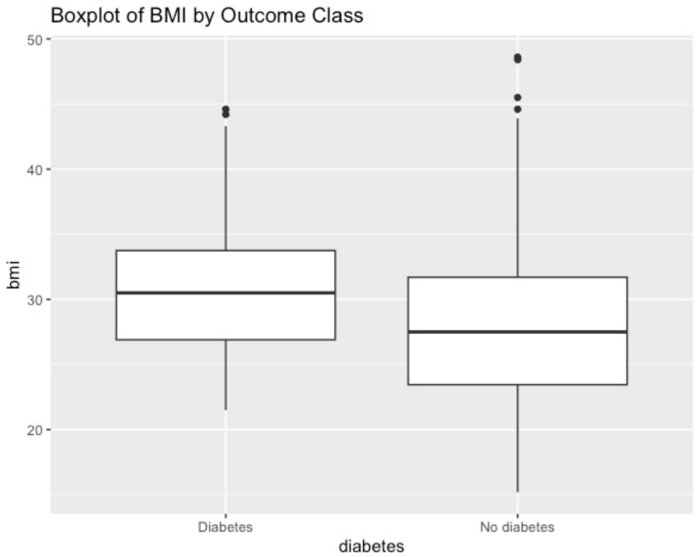


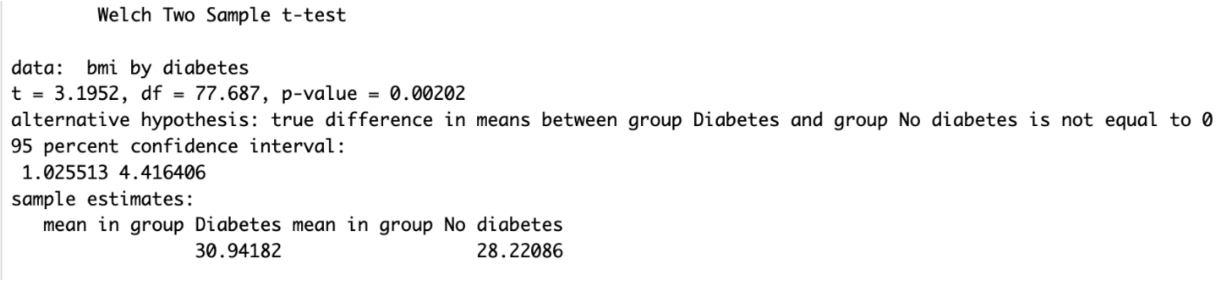


```
        Welch Two Sample t-test

data:  height by diabetes
t = 0.11364, df = 74.22, p-value = 0.9098
alternative hypothesis: true difference in means between group Diabetes and group No diabetes is not equal to 0
95 percent confidence interval:
 -1.065045  1.193880
sample estimates:
   mean in group Diabetes mean in group No diabetes
                 66.00000                  65.93558
```

As the p-value is greater than the significance level, we do not reject the Null Hypothesis and conclude that mean height is statistically different in individuals with diabetes against those who don’t have diabetes. In simple terms, Height is a not a strong factor in determining if an individual has diabetes.

Boxplot of BMI by outcome class along with the T-test to check its significance:

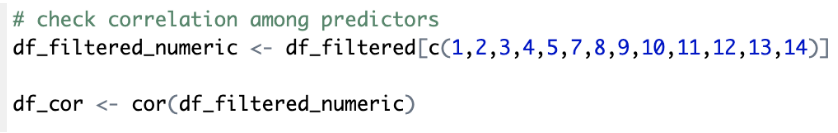




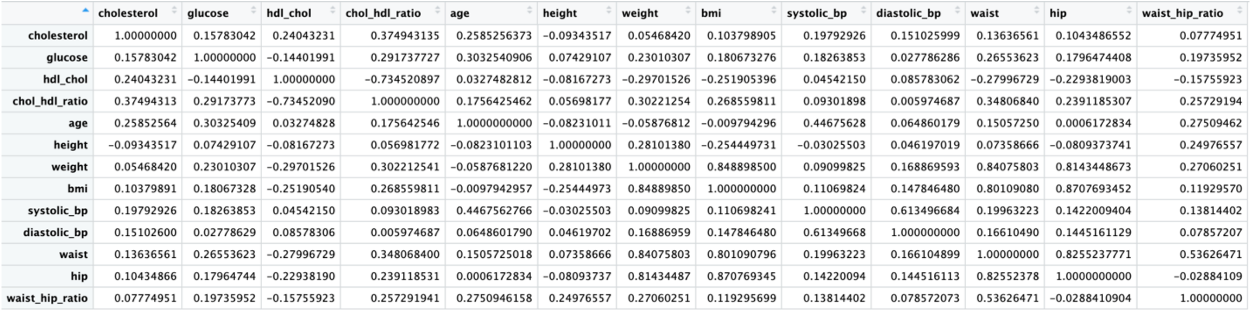
```
        Welch Two Sample t-test

data:  bmi by diabetes
t = 3.1952, df = 77.687, p-value = 0.00202
alternative hypothesis: true difference in means between group Diabetes and group No diabetes is not equal to 0
95 percent confidence interval:
 1.025513 4.416406
sample estimates:
   mean in group Diabetes mean in group No diabetes
                 30.94182                  28.22086
```

As the p-value is less than the significance level, we reject the Null Hypothesis and conclude that mean BMI level is statistically different in individuals with diabetes against those who don’t have diabetes. In simple terms, BMI is a very important factor in determining if an individual has diabetes.

Checking Correlation among Predictors to dampen multicollinearity:

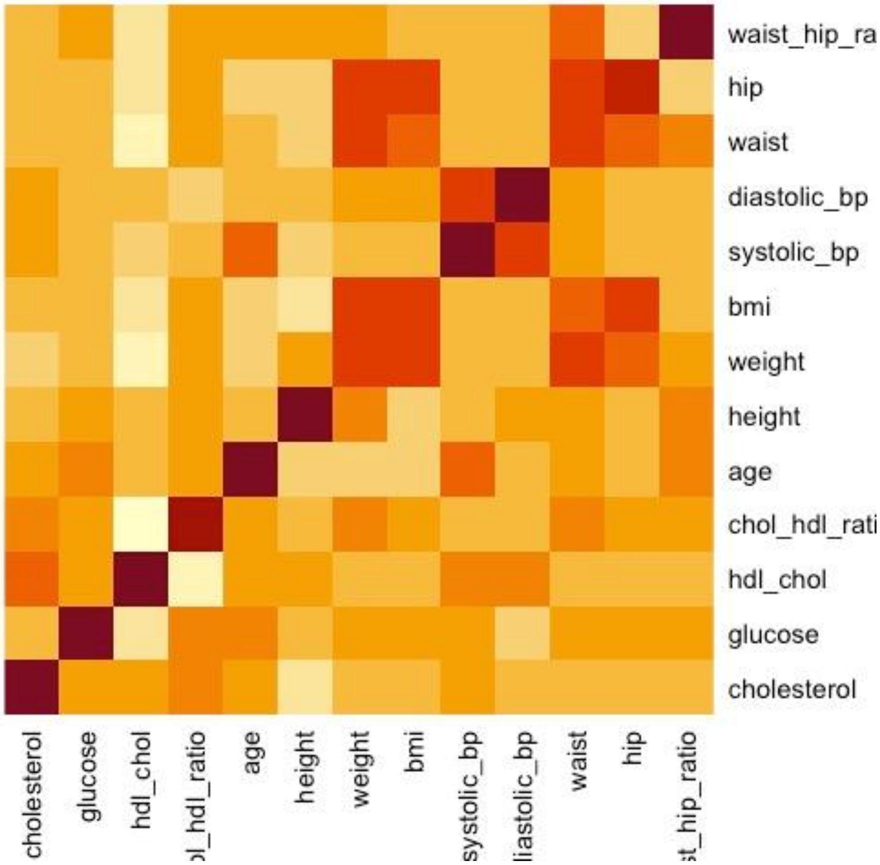


```
# check correlation among predictors
df_filtered_numeric <- df_filtered[c(1,2,3,4,5,7,8,9,10,11,12,13,14)]

df_cor <- cor(df_filtered_numeric)
```



	cholesterol	glucose	hdl_chol	chol_hdl_ratio	age	height	weight	bmi	systolic_bp	diastolic_bp	waist	hip	waist_hip_ratio
cholesterol	1.00000000	0.15783042	0.24043231	0.374943135	0.2585256373	-0.09343517	0.05468420	0.103798905	0.19792926	0.151025999	0.13636561	0.1043486552	0.07774951
glucose	0.15783042	1.00000000	-0.14401991	0.291737727	0.3032540906	0.07429107	0.23010307	0.180673276	0.18263853	0.027786286	0.26553623	0.1796474408	0.19735952
hdl_chol	0.24043231	-0.14401991	1.00000000	-0.734520897	0.0327482812	-0.08167273	-0.29701526	-0.251905396	0.04542150	0.085783062	-0.27996729	-0.2293819003	-0.15755923
chol_hdl_ratio	0.37494313	0.29173773	-0.73452090	1.000000000	0.1756425462	0.05698177	0.30221254	0.268559811	0.09301898	0.005974687	0.34806840	0.2391185307	0.25729194
age	0.25852564	0.30325409	0.03274828	0.175642546	1.0000000000	-0.08231011	-0.05876812	-0.009794296	0.44675628	0.064860179	0.15057250	0.0006172834	0.27509462
height	-0.09343517	0.07429107	-0.08167273	0.056981772	-0.0823101103	1.00000000	0.28101380	-0.254449731	-0.03025503	0.046197019	0.07358666	-0.0809373741	0.24976557
weight	0.05468420	0.23010307	-0.29701526	0.302212541	-0.0587681220	0.28101380	1.00000000	0.848898500	0.09099825	0.168869593	0.84075803	0.8143448673	0.27060251
bmi	0.10379891	0.18067328	-0.25190540	0.268559811	-0.0097942957	-0.25444973	0.84889850	1.000000000	0.11069824	0.147846480	0.80109080	0.8707693452	0.11929570
systolic_bp	0.19792926	0.18263853	0.04542150	0.093018983	0.4467562766	-0.03025503	0.09099825	0.110698241	1.00000000	0.613496684	0.19963223	0.1422009404	0.13814402
diastolic_bp	0.15102600	0.02778629	0.08578306	0.005974687	0.0648601790	0.04619702	0.16886959	0.147846480	0.61349668	1.000000000	0.16610490	0.1445161129	0.07857207
waist	0.13636561	0.26553623	-0.27996729	0.348068400	0.1505725018	0.07358666	0.84075803	0.801090796	0.19963223	0.166104899	1.00000000	0.8255237771	0.53626471
hip	0.10434866	0.17964744	-0.22938190	0.239118531	0.0006172834	-0.08093737	0.81434487	0.870769345	0.14220094	0.144516113	0.82552378	1.0000000000	-0.02884109
waist_hip_ratio	0.07774951	0.19735952	-0.15755923	0.257291941	0.2750946158	0.24976557	0.27060251	0.119295699	0.13814402	0.078572073	0.53626471	-0.0288410904	1.00000000

As the interpreting correlation matrix is confusing, below is a heatmap of all the predictors:



Variable Selection based on Exploratory data analysis:

Dropped Variables:

1. Patient Number
   It doesn't determine the provide any useful information.

2. Gender
   Based on the Chi Square test, the probability for males and females having diabetes is statistically the same.

3. Cholesterol & HDL_Chol
   A derived attribute of these two measurements is used – CHOL to HDL ratio.

4. Height & Weight
   A derived attribute of these two measurements is used – BMI.

5. Waist & Hip
   A derived attribute of these two measurements is used – Waist to Hip ratio.

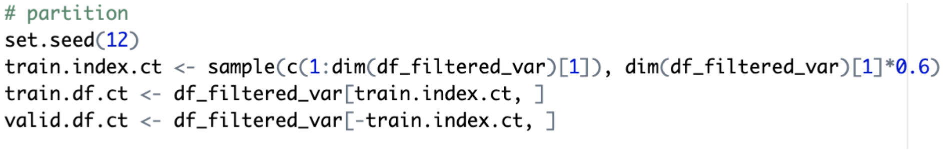
List of Shortlisted Variables:

1. Glucose
2. Chol to HDL ratio
3. Age
4. BMI
5. Systolic BP
6. Diastolic BP
7. Waist to Hip ratio
8. Diabetes Class

Machine Learning Models

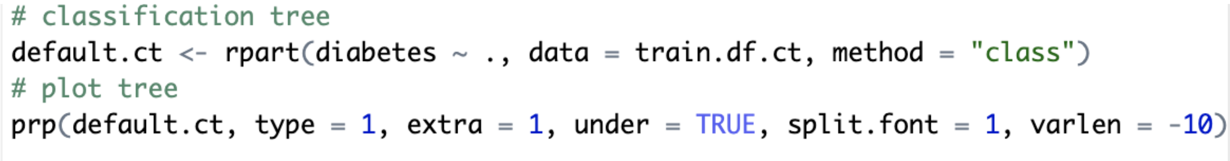
Classification Tree:

Step 1: Partitioning the data.

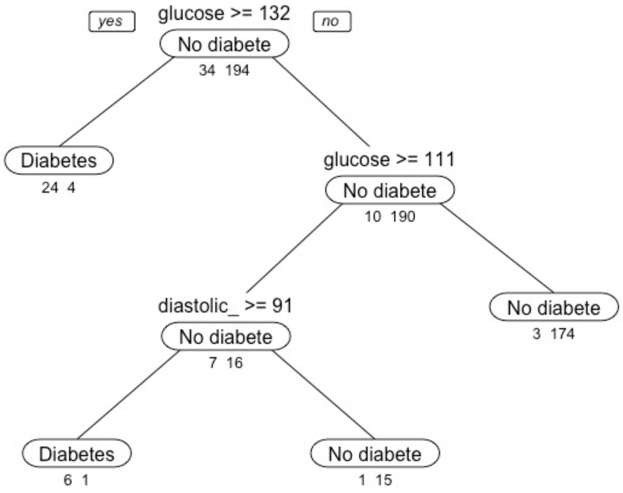


```
# partition
set.seed(12)
train.index.ct <- sample(c(1:dim(df_filtered_var)[1]), dim(df_filtered_var)[1]*0.6)
train.df.ct <- df_filtered_var[train.index.ct, ]
valid.df.ct <- df_filtered_var[-train.index.ct, ]
```

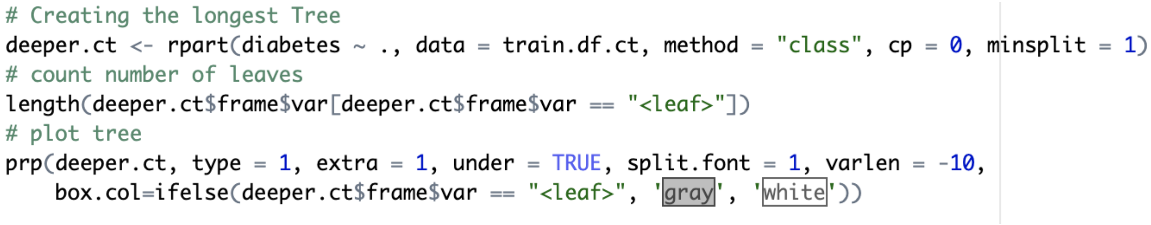
Step 2: Creating the Default Tree



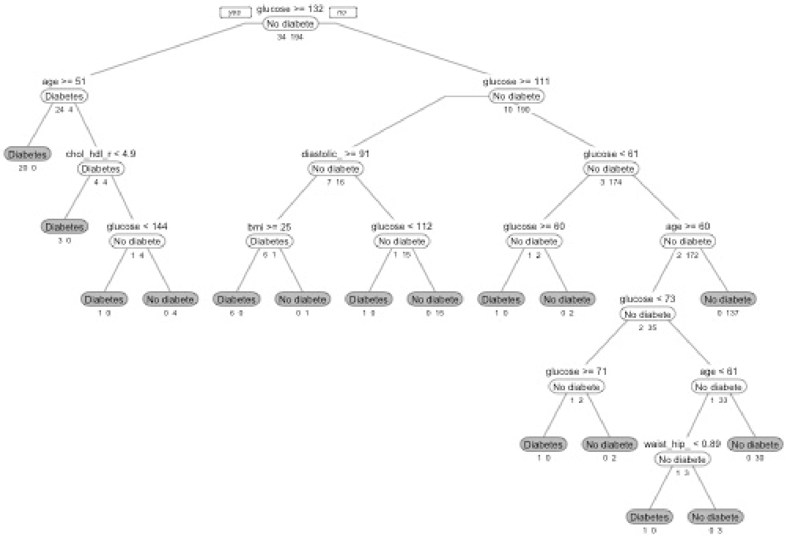
```
# classification tree
default.ct <- rpart(diabetes ~ ., data = train.df.ct, method = "class")
# plot tree
prp(default.ct, type = 1, extra = 1, under = TRUE, split.font = 1, varlen = -10)
```



Step 3: Creating the Longest Tree



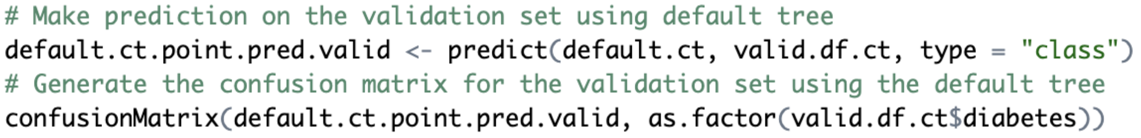
```
# Creating the longest Tree
deeper.ct <- rpart(diabetes ~ ., data = train.df.ct, method = "class", cp = 0, minsplit = 1)
# count number of leaves
length(deeper.ct$frame$var[deeper.ct$frame$var == "<leaf>"])
# plot tree
prp(deeper.ct, type = 1, extra = 1, under = TRUE, split.font = 1, varlen = -10,
    box.col=ifelse(deeper.ct$frame$var == "<leaf>", 'gray', 'white'))
```



The longest tree perfectly fits the data even modeling the noise which is what we want to avoid. To reduce the issue of overfitting we have to trim this deeper cut.

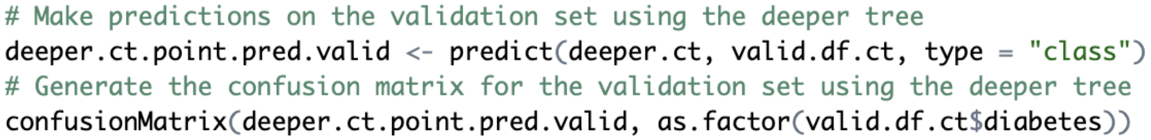
Step 4: Making predictions based on validation set for the default and deeper cur trees:

Default Tree:



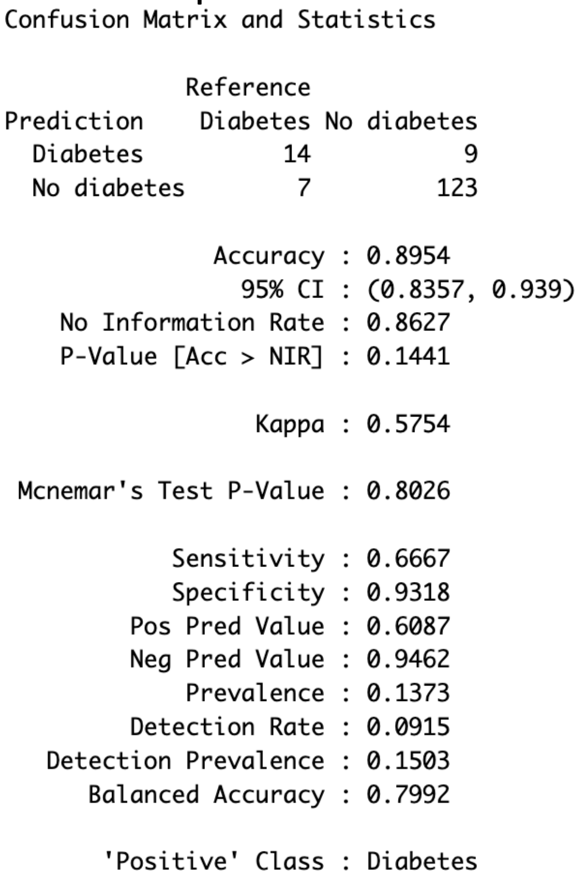
```
# Make prediction on the validation set using default tree
default.ct.point.pred.valid <- predict(default.ct, valid.df.ct, type = "class")
# Generate the confusion matrix for the validation set using the default tree
confusionMatrix(default.ct.point.pred.valid, as.factor(valid.df.ct$diabetes))
```

Longest Tree:



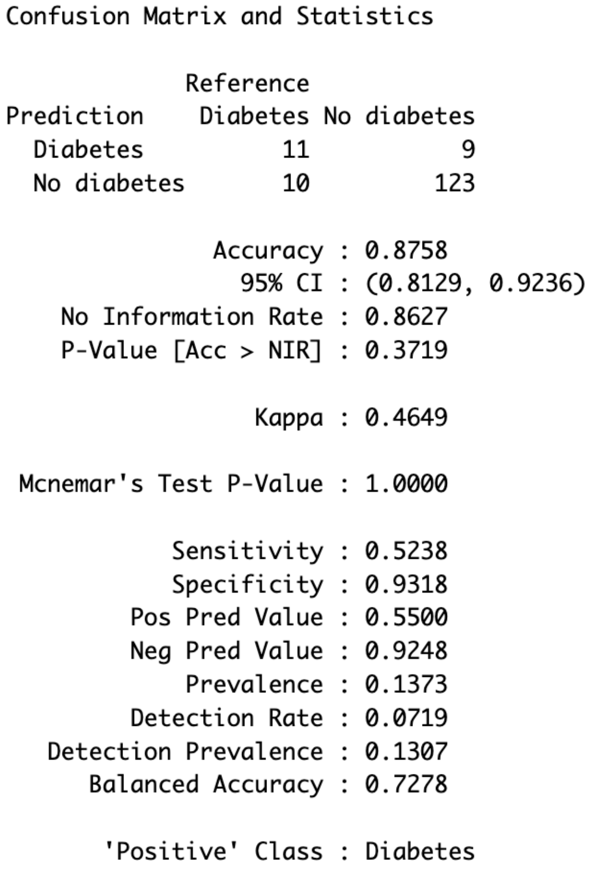
```
# Make predictions on the validation set using the deeper tree
deeper.ct.point.pred.valid <- predict(deeper.ct, valid.df.ct, type = "class")
# Generate the confusion matrix for the validation set using the deeper tree
confusionMatrix(deeper.ct.point.pred.valid, as.factor(valid.df.ct$diabetes))
```

Deeper Cut



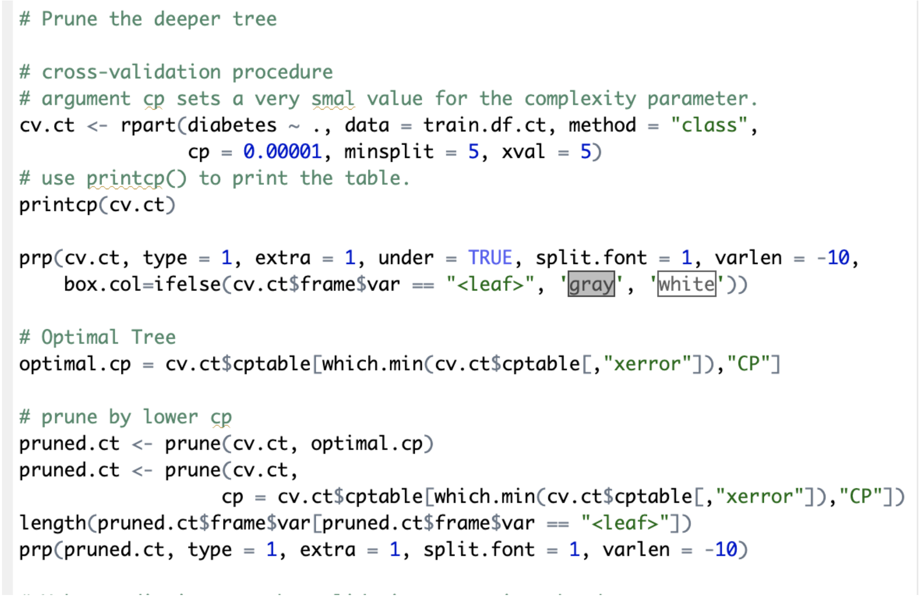
```
Confusion Matrix and Statistics

             Reference
Prediction    Diabetes No diabetes
  Diabetes          14           9
  No diabetes        7         123

               Accuracy : 0.8954
                 95% CI : (0.8357, 0.939)
    No Information Rate : 0.8627
    P-Value [Acc > NIR] : 0.1441

                  Kappa : 0.5754

 Mcnemar's Test P-Value : 0.8026

            Sensitivity : 0.6667
            Specificity : 0.9318
         Pos Pred Value : 0.6087
         Neg Pred Value : 0.9462
             Prevalence : 0.1373
         Detection Rate : 0.0915
   Detection Prevalence : 0.1503
      Balanced Accuracy : 0.7992

       'Positive' Class : Diabetes
```

Default Tree



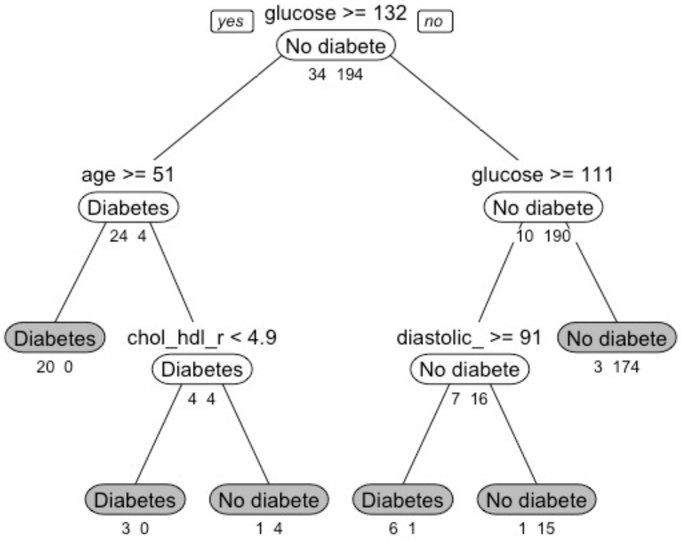
```
Confusion Matrix and Statistics

             Reference
Prediction    Diabetes No diabetes
  Diabetes          11           9
  No diabetes       10         123

               Accuracy : 0.8758
                 95% CI : (0.8129, 0.9236)
    No Information Rate : 0.8627
    P-Value [Acc > NIR] : 0.3719

                  Kappa : 0.4649

 Mcnemar's Test P-Value : 1.0000

            Sensitivity : 0.5238
            Specificity : 0.9318
         Pos Pred Value : 0.5500
         Neg Pred Value : 0.9248
             Prevalence : 0.1373
         Detection Rate : 0.0719
   Detection Prevalence : 0.1307
      Balanced Accuracy : 0.7278

       'Positive' Class : Diabetes
```

Step 5: Pruning the longest Tree:



```r
# Prune the deeper tree

# cross-validation procedure
# argument cp sets a very smal value for the complexity parameter.
cv.ct <- rpart(diabetes ~ ., data = train.df.ct, method = "class",
               cp = 0.00001, minsplit = 5, xval = 5)
# use printcp() to print the table.
printcp(cv.ct)

prp(cv.ct, type = 1, extra = 1, under = TRUE, split.font = 1, varlen = -10,
    box.col=ifelse(cv.ct$frame$var == "<leaf>", 'gray', 'white'))

# Optimal Tree
optimal.cp = cv.ct$cptable[which.min(cv.ct$cptable[,"xerror"]),"CP"]

# prune by lower cp
pruned.ct <- prune(cv.ct, optimal.cp)
pruned.ct <- prune(cv.ct,
                  cp = cv.ct$cptable[which.min(cv.ct$cptable[,"xerror"]),"CP"])
length(pruned.ct$frame$var[pruned.ct$frame$var == "<leaf>"])
prp(pruned.ct, type = 1, extra = 1, split.font = 1, varlen = -10)
```

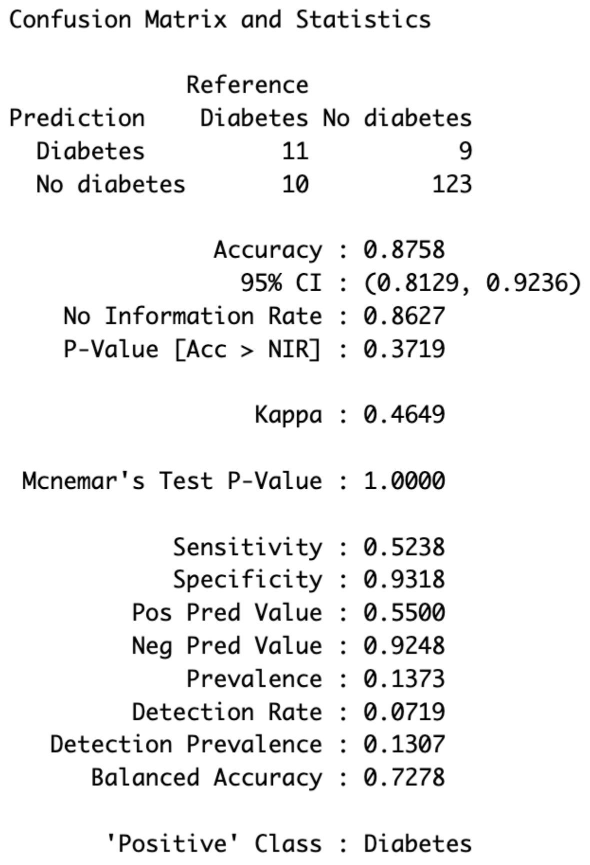
Classification Tree After Pruning



Rules:

1. If the glucose level is greater or equal to 132 & Age is greater or equal to 51 then the individual has diabetes
2. If the glucose level is greater or equal to 132 & Age is less than 51 & Chol_HDL ratio is less than 4.9 then the individual has diabetes
3. If the glucose level is greater or equal to 132 & Age is less than 51 & Chol_HDL ratio is greater or equal to 4.9 then the individual does not have diabetes
4. If the glucose is level is less than 132 & less than 111 then the individual does not have diabetes.
5. If the glucose is level is less than 132 & greater or equal to 111 & diastolic BP is greater or equal to 91 then the individual has diabetes.
6. If the glucose is level is less than 132 & greater or equal to 111 & diastolic BP is less than 91 then the individual does not have diabetes.

Step 6: Evaluating performance of the Pruned Tree

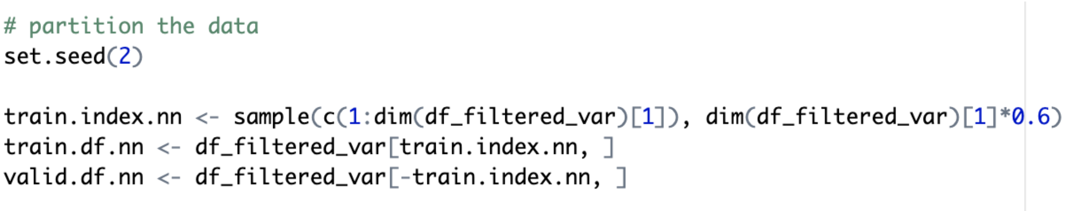


```
Confusion Matrix and Statistics

             Reference
Prediction    Diabetes No diabetes
  Diabetes          11           9
  No diabetes       10         123

               Accuracy : 0.8758
                 95% CI : (0.8129, 0.9236)
    No Information Rate : 0.8627
    P-Value [Acc > NIR] : 0.3719

                  Kappa : 0.4649

 Mcnemar's Test P-Value : 1.0000

            Sensitivity : 0.5238
            Specificity : 0.9318
         Pos Pred Value : 0.5500
         Neg Pred Value : 0.9248
             Prevalence : 0.1373
         Detection Rate : 0.0719
   Detection Prevalence : 0.1307
      Balanced Accuracy : 0.7278

       'Positive' Class : Diabetes
```

The pruned classification tree can predict the class of patient with an accuracy of 87.58%.

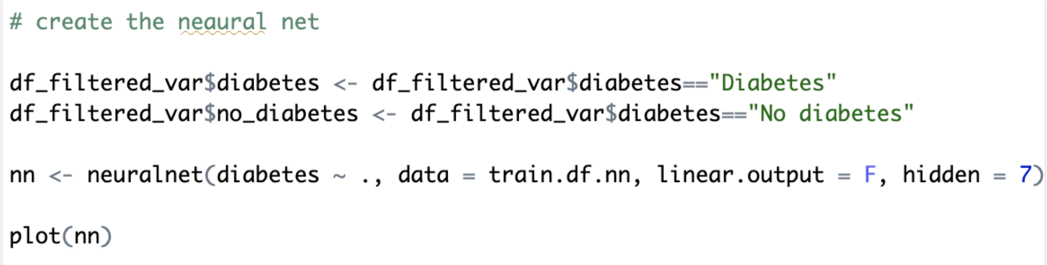
Neural Nets:

Step 1: Partitioning the Data

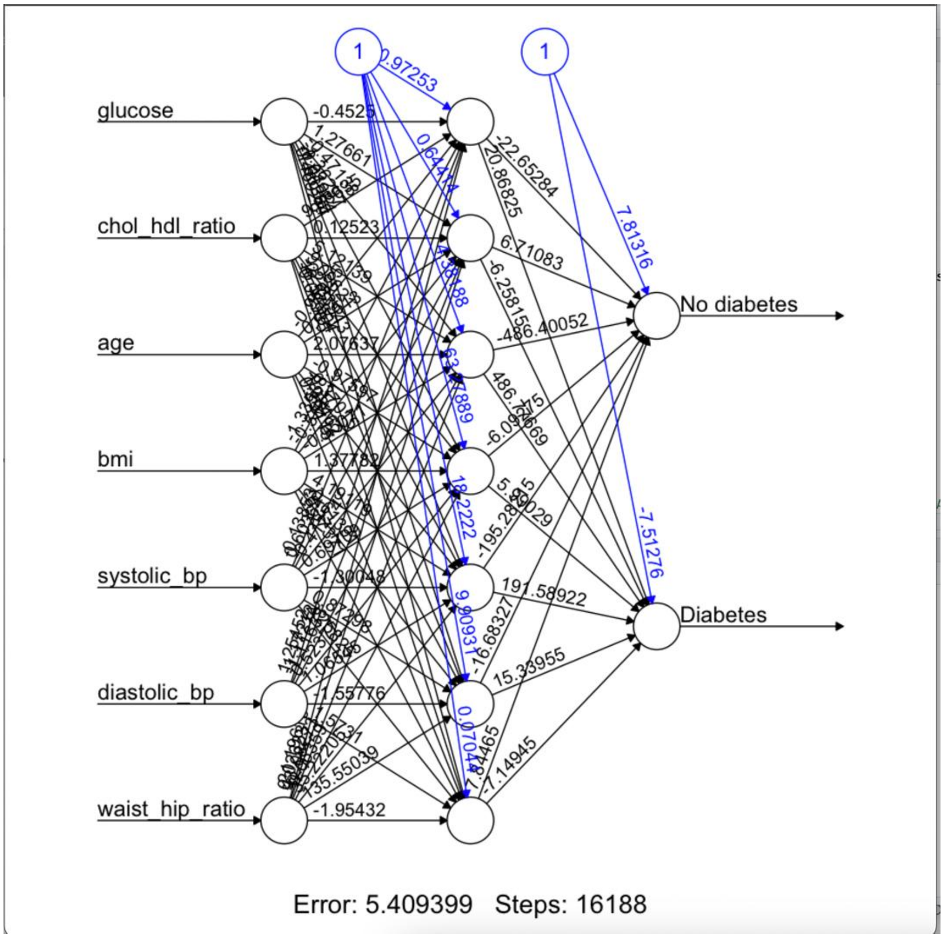


```
# partition the data
set.seed(2)

train.index.nn <- sample(c(1:dim(df_filtered_var)[1]), dim(df_filtered_var)[1]*0.6)
train.df.nn <- df_filtered_var[train.index.nn, ]
valid.df.nn <- df_filtered_var[-train.index.nn, ]
```

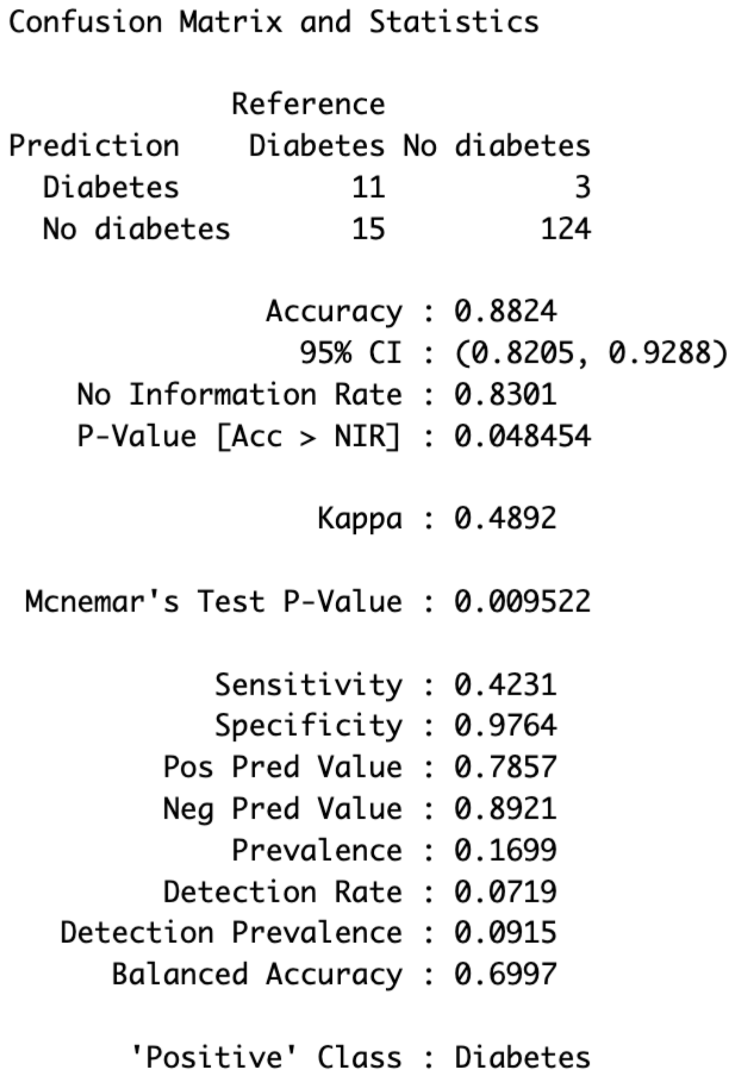
Step 2: Creating the Neural Net



```
# create the neaural net

df_filtered_var$diabetes <- df_filtered_var$diabetes=="Diabetes"
df_filtered_var$no_diabetes <- df_filtered_var$diabetes=="No diabetes"

nn <- neuralnet(diabetes ~ ., data = train.df.nn, linear.output = F, hidden = 7)

plot(nn)
```



Evaluating Performance of the Neural Network:

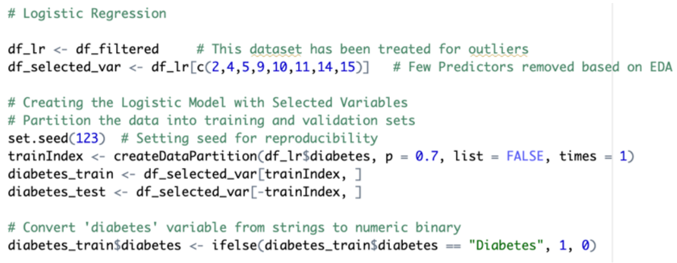


```
Confusion Matrix and Statistics

             Reference
Prediction    Diabetes No diabetes
  Diabetes          11           3
  No diabetes       15         124

               Accuracy : 0.8824
                 95% CI : (0.8205, 0.9288)
    No Information Rate : 0.8301
    P-Value [Acc > NIR] : 0.048454

                  Kappa : 0.4892

 Mcnemar's Test P-Value : 0.009522

            Sensitivity : 0.4231
            Specificity : 0.9764
         Pos Pred Value : 0.7857
         Neg Pred Value : 0.8921
             Prevalence : 0.1699
         Detection Rate : 0.0719
   Detection Prevalence : 0.0915
      Balanced Accuracy : 0.6997

       'Positive' Class : Diabetes
```

The model can Predict the outcome class with an accuracy of 88.24% .

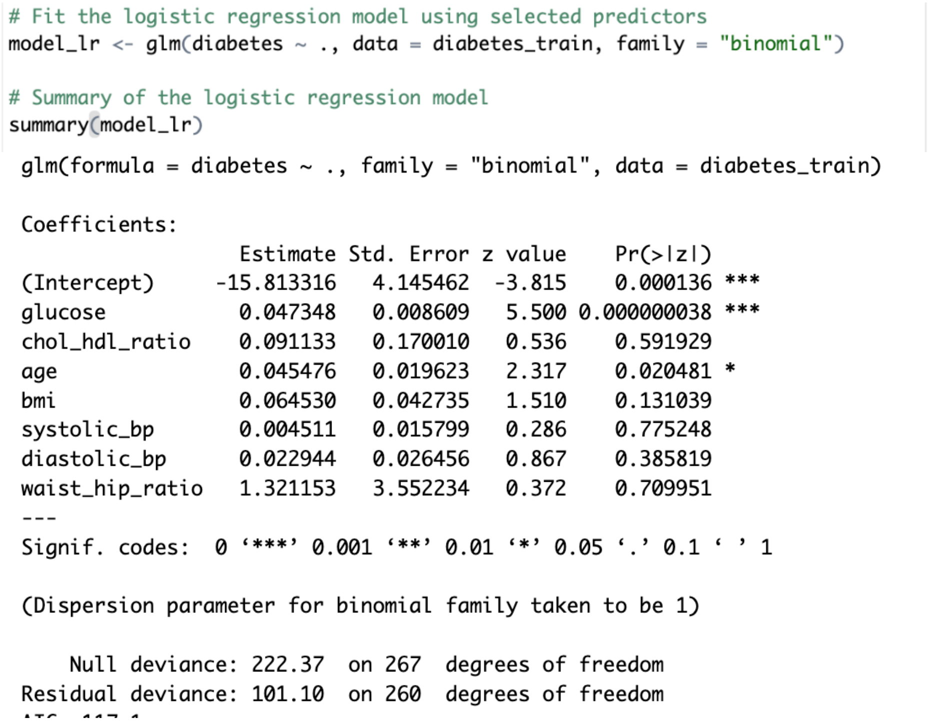
Logistic Regression:

Step 1: Defining & Partitioning the Data



```
# Logistic Regression

df_lr <- df_filtered     # This dataset has been treated for outliers
df_selected_var <- df_lr[c(2,4,5,9,10,11,14,15)]   # Few Predictors removed based on EDA

# Creating the Logistic Model with Selected Variables
# Partition the data into training and validation sets
set.seed(123)  # Setting seed for reproducibility
trainIndex <- createDataPartition(df_lr$diabetes, p = 0.7, list = FALSE, times = 1)
diabetes_train <- df_selected_var[trainIndex, ]
diabetes_test <- df_selected_var[-trainIndex, ]

# Convert 'diabetes' variable from strings to numeric binary
diabetes_train$diabetes <- ifelse(diabetes_train$diabetes == "Diabetes", 1, 0)
```

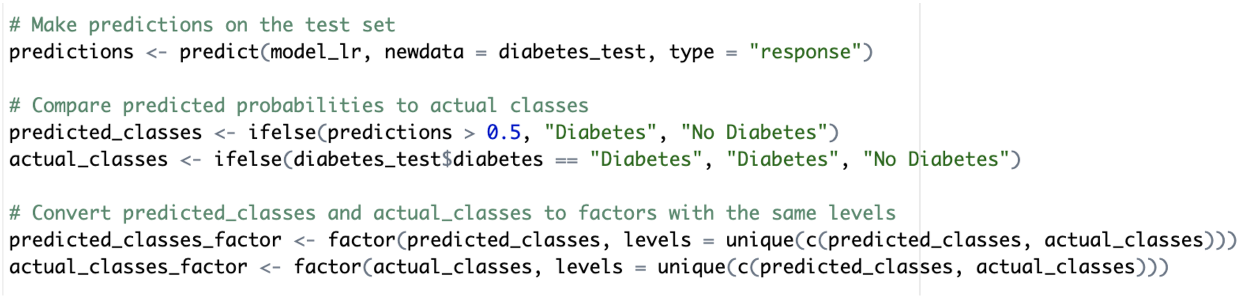
Step 2: Fitting the Model



```
# Fit the logistic regression model using selected predictors
model_lr <- glm(diabetes ~ ., data = diabetes_train, family = "binomial")

# Summary of the logistic regression model
summary(model_lr)
```

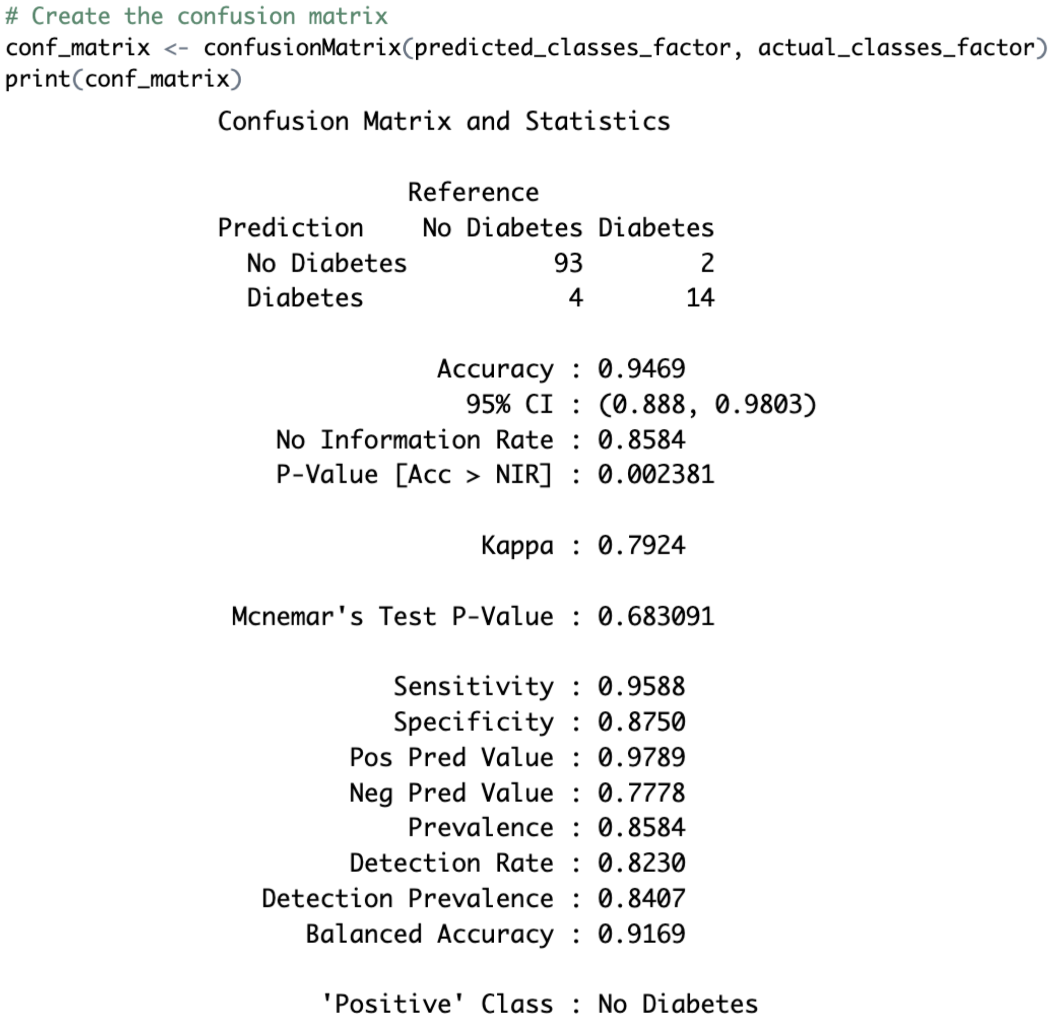
```
glm(formula = diabetes ~ ., family = "binomial", data = diabetes_train)

Coefficients:
                  Estimate Std. Error z value    Pr(>|z|)
(Intercept)     -15.813316   4.145462  -3.815    0.000136 ***
glucose           0.047348   0.008609   5.500 0.000000038 ***
chol_hdl_ratio    0.091133   0.170010   0.536    0.591929
age               0.045476   0.019623   2.317    0.020481 *
bmi               0.064530   0.042735   1.510    0.131039
systolic_bp       0.004511   0.015799   0.286    0.775248
diastolic_bp      0.022944   0.026456   0.867    0.385819
waist_hip_ratio   1.321153   3.552234   0.372    0.709951
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 222.37  on 267  degrees of freedom
Residual deviance: 101.10  on 260  degrees of freedom
```

Step 3: Making Prediction on the Validation Set



```
# Make predictions on the test set
predictions <- predict(model_lr, newdata = diabetes_test, type = "response")

# Compare predicted probabilities to actual classes
predicted_classes <- ifelse(predictions > 0.5, "Diabetes", "No Diabetes")
actual_classes <- ifelse(diabetes_test$diabetes == "Diabetes", "Diabetes", "No Diabetes")

# Convert predicted_classes and actual_classes to factors with the same levels
predicted_classes_factor <- factor(predicted_classes, levels = unique(c(predicted_classes, actual_classes)))
actual_classes_factor <- factor(actual_classes, levels = unique(c(predicted_classes, actual_classes)))
```

Step 4: Assessing Performance

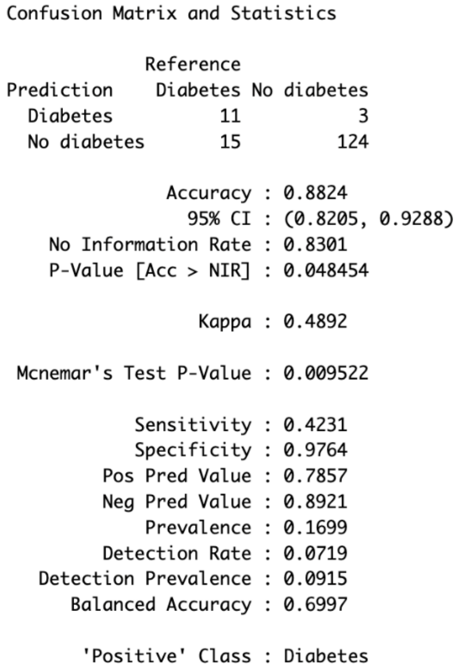


```
# Create the confusion matrix
conf_matrix <- confusionMatrix(predicted_classes_factor, actual_classes_factor)
print(conf_matrix)
```

```
Confusion Matrix and Statistics

             Reference
Prediction    No Diabetes Diabetes
  No Diabetes          93        2
  Diabetes              4       14

               Accuracy : 0.9469
                 95% CI : (0.888, 0.9803)
    No Information Rate : 0.8584
    P-Value [Acc > NIR] : 0.002381

                  Kappa : 0.7924

 Mcnemar's Test P-Value : 0.683091

            Sensitivity : 0.9588
            Specificity : 0.8750
         Pos Pred Value : 0.9789
         Neg Pred Value : 0.7778
             Prevalence : 0.8584
         Detection Rate : 0.8230
   Detection Prevalence : 0.8407
      Balanced Accuracy : 0.9169

       'Positive' Class : No Diabetes
```

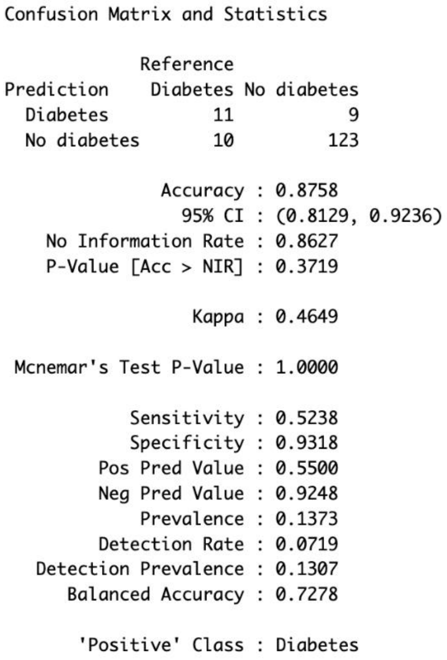
Neural Nets VS **Regression Trees** VS **Logistic Regression**

Neural Net



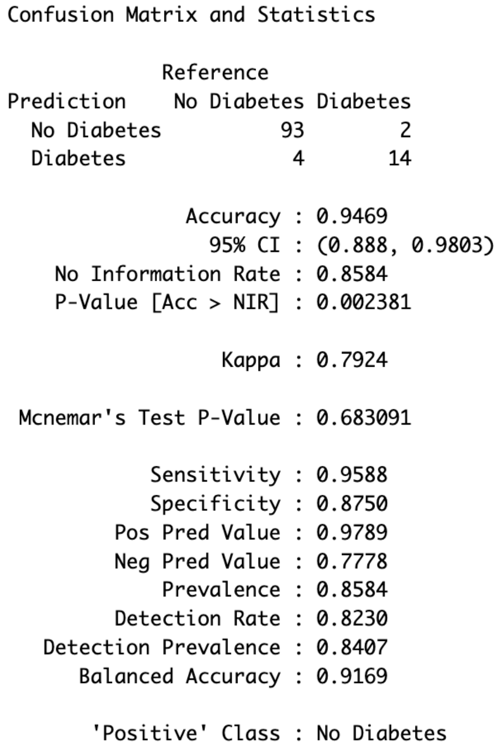
```
Confusion Matrix and Statistics

             Reference
Prediction    Diabetes No diabetes
  Diabetes          11           3
  No diabetes       15         124

               Accuracy : 0.8824
                 95% CI : (0.8205, 0.9288)
    No Information Rate : 0.8301
    P-Value [Acc > NIR] : 0.048454

                  Kappa : 0.4892

 Mcnemar's Test P-Value : 0.009522

            Sensitivity : 0.4231
            Specificity : 0.9764
         Pos Pred Value : 0.7857
         Neg Pred Value : 0.8921
             Prevalence : 0.1699
         Detection Rate : 0.0719
   Detection Prevalence : 0.0915
      Balanced Accuracy : 0.6997

       'Positive' Class : Diabetes
```

Regression Tree



```
Confusion Matrix and Statistics

             Reference
Prediction    Diabetes No diabetes
  Diabetes          11           9
  No diabetes       10         123

               Accuracy : 0.8758
                 95% CI : (0.8129, 0.9236)
    No Information Rate : 0.8627
    P-Value [Acc > NIR] : 0.3719

                  Kappa : 0.4649

 Mcnemar's Test P-Value : 1.0000

            Sensitivity : 0.5238
            Specificity : 0.9318
         Pos Pred Value : 0.5500
         Neg Pred Value : 0.9248
             Prevalence : 0.1373
         Detection Rate : 0.0719
   Detection Prevalence : 0.1307
      Balanced Accuracy : 0.7278

       'Positive' Class : Diabetes
```

Logistic Regression



```
Confusion Matrix and Statistics

             Reference
Prediction    No Diabetes Diabetes
  No Diabetes          93        2
  Diabetes              4       14

               Accuracy : 0.9469
                 95% CI : (0.888, 0.9803)
    No Information Rate : 0.8584
    P-Value [Acc > NIR] : 0.002381

                  Kappa : 0.7924

 Mcnemar's Test P-Value : 0.683091

            Sensitivity : 0.9588
            Specificity : 0.8750
         Pos Pred Value : 0.9789
         Neg Pred Value : 0.7778
             Prevalence : 0.8584
         Detection Rate : 0.8230
   Detection Prevalence : 0.8407
      Balanced Accuracy : 0.9169

       'Positive' Class : No Diabetes
```

Conclusion:

After an extensive comparison of the three models, the following conclusions have been drawn:

1. Logistic Regression exhibits the highest accuracy and precision among the models, emphasizing its robustness in predicting diabetes cases accurately.
2. Neural Networks showcase competitive accuracy and precision, but their sensitivity is comparatively lower than Logistic Regression.
3. Regression Tree displays moderate performance metrics, with balanced sensitivity and specificity, but lags Logistic Regression and Neural Networks in precision and overall accuracy.
4. Considering the trade-off between accuracy, sensitivity, specificity, and precision, Logistic Regression remains the most favorable model for diabetes prediction due to its superior performance in all key metrics.
5. Logistic Regression provides a good balance between sensitivity and specificity, making it a reliable choice for identifying diabetes cases while minimizing false positives and negatives.
6. For this dataset, Logistic Regression emerges as the preferred model for accurate diabetes prediction and risk assessment.