# Final Project DSC520

Mohit Matta

February 27th, 2020

## Introduction

Diabetes, often described as a "Disease of Civilization", is a major public health problem that is approaching epidemic proportions globally. Undiagnosed diabetes can be predisposing factor to a fatal cardiac stroke. Its exponentially increasing cases has become a matter of concern world wide. Usually onset of type 2 diabetes happens in middle age and sometimes in old age. But nowadays incidences of this disease are reported in children as well. Risk factors leading to diabetes range from genetic susceptibility and body weight to food habits and lifestyle. Adult with diabetes have a two-to-three fold increased risk of heart attacks, neuropathy, foot ulcers, limb amputation and kidney failure. Early diagnosis is crucial and can be accomplished through relatively inexpensive testing of blood sugar. Diabetes can be controlled by promoting healthy diet and regular exercise, thereby reducing the growing global problem of overweight people and obesity.

Classification is one of the most important decision making techniques in many real world problem. In this work, the main objective is to classify the data as diabetic or non-diabetic and improve the classification accuracy. The main objective of our model is to achieve high accuracy.Classification accuracy can be increased if we use much of the data set for training and few data sets for testing.The aim of this project is to develop a system which can predict the diabetic risk level of a patient with a higher accuracy.

### Data

The dataset can be downloaded from the link. ( https://www.kaggle.com/uciml/pima-indians-diabetes-database#diabetes.csv)

This dataset is originally from the National Institute of Diabetes and Digestive and Kidney Diseases. The objective of the dataset is to diagnostically predict whether or not a patient has diabetes, based on certain diagnostic measurements included in the dataset. Several constraints were placed on the selection of these instances from a larger database. In particular, all patients here are females at least 21 years old of Pima Indian heritage.The dataset contains 9 columns and 2000 observations.The format is csv.Below are the column names and their description:

Pregnancies: Number of times pregnant

Glucose: Plasma glucose concentration over 2 hours in an oral glucose tolerance test

BloodPressure: Diastolic blood pressure (mm Hg)

SkinThickness: Triceps skin fold thickness (mm)

Insulin: 2-Hour serum insulin (mu U/ml)

BMI: Body mass index (weight in kg/(height in m)2)

DiabetesPedigreeFunction: Diabetes pedigree function (a function which scores likelihood of diabetes based on family history)

Age: Age (years)

Outcome: Class variable (0 if non-diabetic, 1 if diabetic)

## Import and Clean Dataset

Raw data files may lack headers, contain wrong data types (e.g. numbers stored as strings), wrong category labels, unknown or unexpected character encoding and so on. In short, reading such files into an R data.frame directly is either difficult or impossible without some sort of preprocessing. We can say data is technically correct only after preprocessing is completed and data can be read with correct labels/datatypes.Below is the logical process I am planning to follow :

1) Obtain the dataset
2) Clean our dataset
3) Explore/Visualize dataset to allow to find patterns and trends
4) Model the data for predictive power
5) Interpret the data

Below is the structure of raw dataset :

```
## 'data.frame':    2000 obs. of  9 variables:
##  $ Pregnancies             : int  2 0 0 0 1 0 4 8 2 2 ...
##  $ Glucose                 : int  138 84 145 135 139 173 99 194 83 89 ...
##  $ BloodPressure           : int  62 82 0 68 62 78 72 80 65 90 ...
##  $ SkinThickness           : int  35 31 0 42 41 32 17 0 28 30 ...
##  $ Insulin                 : int  0 125 0 250 480 265 0 0 66 0 ...
##  $ BMI                     : num  33.6 38.2 44.2 42.3 40.7 46.5 25.6 26.1 36.8 33.5 ...
##  $ DiabetesPedigreeFunction: num  0.127 0.233 0.63 0.365 0.536 ...
##  $ Age                     : int  47 23 31 24 21 58 28 67 24 42 ...
##  $ Outcome                 : int  1 0 1 1 0 0 0 0 0 0 ...
```

Below is the summary of dataset :

```
##   Pregnancies        Glucose      BloodPressure    SkinThickness
##  Min.   : 0.000   Min.   :  0.0   Min.   :  0.00   Min.   :  0.00
##  1st Qu.: 1.000   1st Qu.: 99.0   1st Qu.: 63.50   1st Qu.:  0.00
##  Median : 3.000   Median :117.0   Median : 72.00   Median : 23.00
##  Mean   : 3.704   Mean   :121.2   Mean   : 69.15   Mean   : 20.93
##  3rd Qu.: 6.000   3rd Qu.:141.0   3rd Qu.: 80.00   3rd Qu.: 32.00
##  Max.   :17.000   Max.   :199.0   Max.   :122.00   Max.   :110.00
##     Insulin           BMI        DiabetesPedigreeFunction      Age
##  Min.   :  0.00   Min.   : 0.00   Min.   :0.0780           Min.   :21.00
##  1st Qu.:  0.00   1st Qu.:27.38   1st Qu.:0.2440           1st Qu.:24.00
##  Median : 40.00   Median :32.30   Median :0.3760           Median :29.00
##  Mean   : 80.25   Mean   :32.19   Mean   :0.4709           Mean   :33.09
##  3rd Qu.:130.00   3rd Qu.:36.80   3rd Qu.:0.6240           3rd Qu.:40.00
##  Max.   :744.00   Max.   :80.60   Max.   :2.4200           Max.   :81.00
##     Outcome
##  Min.   :0.000
##  1st Qu.:0.000
##  Median :0.000
##  Mean   :0.342
##  3rd Qu.:1.000
##  Max.   :1.000
```

We can see in summary that , the columns Glucose, BloodPressure, SkinThickness, Insulin and BMI have an invalid zero value.The 0 value does not make sense and indicates missing value.It is better to replace zeros with nan since after that counting them would be easier and zeros need to be replaced with suitable values

Below are some sample records from raw dataset :

```
##   Pregnancies Glucose BloodPressure SkinThickness Insulin  BMI
## 1           2     138            62            35       0 33.6
## 2           0      84            82            31     125 38.2
## 3           0     145             0             0       0 44.2
## 4           0     135            68            42     250 42.3
## 5           1     139            62            41     480 40.7
## 6           0     173            78            32     265 46.5
##   DiabetesPedigreeFunction Age Outcome
## 1                    0.127  47       1
## 2                    0.233  23       0
## 3                    0.630  31       1
## 4                    0.365  24       1
## 5                    0.536  21       0
## 6                    1.159  58       0
```

We will now check if the dataset contains any NA values that needs to be removed , below code will show true for any NA values:

#Check if there is any value in dataset with NA

any(is.na(diabetes2))

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

This clearly shows that the dataset does not contain any NA values.

### Clean the dataset

We will first replace all 0 values for Glucose, BloodPressure,SkinThickness,Insulin and BMI columns using below code:

diabetes2[, 2:6][diabetes2[, 2:6] == 0] <- NA

We will now remove all NA values using below code :

diabetes_clean <- na.omit(diabetes2)

Lets see the structure of clean dataset after removing the values:

```
## 'data.frame':    1035 obs. of  9 variables:
##  $ Pregnancies             : int  0 0 1 0 2 4 2 7 6 2 ...
##  $ Glucose                 : int  84 135 139 173 83 125 81 195 154 117 ...
##  $ BloodPressure           : int  82 68 62 78 65 70 72 70 74 90 ...
##  $ SkinThickness           : int  31 42 41 32 28 18 15 33 32 19 ...
##  $ Insulin                 : int  125 250 480 265 66 122 76 145 193 71 ...
##  $ BMI                     : num  38.2 42.3 40.7 46.5 36.8 28.9 30.1 25.1 29.3 25.2 ...
##  $ DiabetesPedigreeFunction: num  0.233 0.365 0.536 1.159 0.629 ...
##  $ Age                     : int  23 24 21 58 24 45 25 55 39 21 ...
##  $ Outcome                 : int  0 1 0 0 0 1 0 1 0 0 ...
##  - attr(*, "na.action")= 'omit' Named int  1 3 7 8 10 11 13 14 15 20 ...
##   ..- attr(*, "names")= chr  "1" "3" "7" "8" ...
```

It now contains 1035 observations. As we can see outcome variable is and integer ,we will need to factor the outcome variable using below code :

**diabetes_clean$Outcome <- as.factor(diabetes_clean$Outcome)**

```
##    Pregnancies Glucose BloodPressure SkinThickness Insulin  BMI
## 2            0      84            82            31     125 38.2
## 4            0     135            68            42     250 42.3
## 5            1     139            62            41     480 40.7
## 6            0     173            78            32     265 46.5
## 9            2      83            65            28      66 36.8
## 12           4     125            70            18     122 28.9
##    DiabetesPedigreeFunction Age Outcome
## 2                     0.233  23       0
## 4                     0.365  24       1
## 5                     0.536  21       0
## 6                     1.159  58       0
## 9                     0.629  24       0
## 12                    1.144  45       1
```

# Count number of outcomes in cleaned dataset with 0 and 1 values

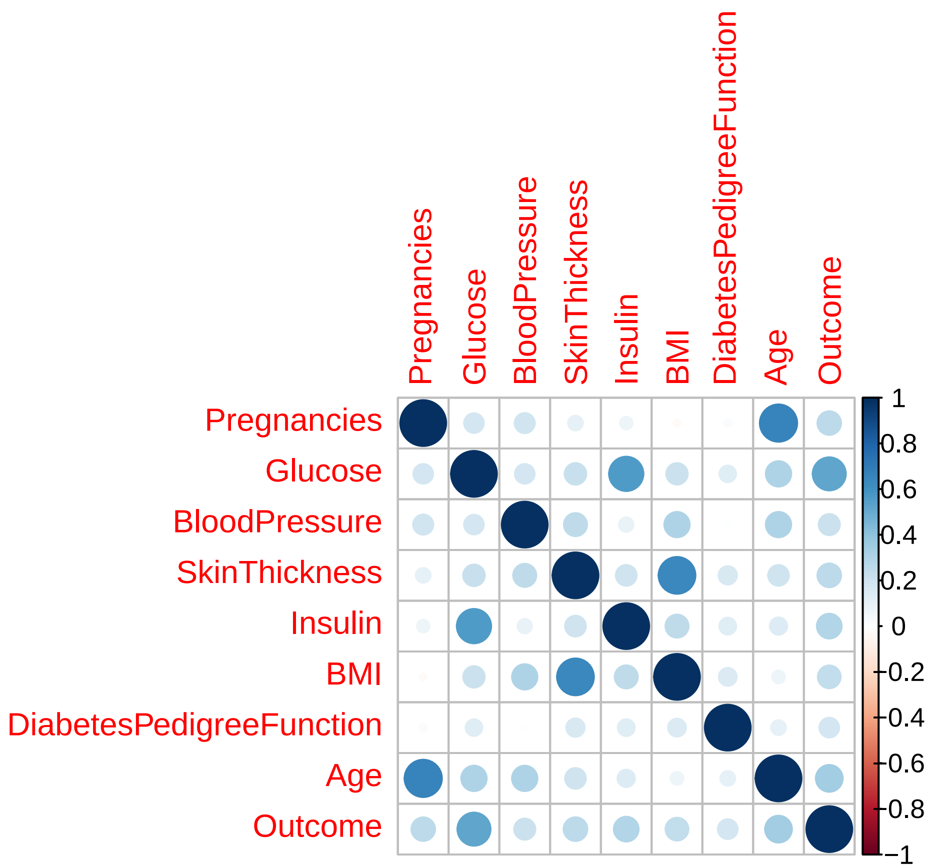
```
##
##   0   1
## 698 337
```

Below is the correlation matrix:

```
##                          Pregnancies   Glucose BloodPressure SkinThickness
## Pregnancies               1.00000000 0.1832721   0.191951482     0.1045123
## Glucose                   0.18327215 1.0000000   0.182876059     0.2213106
## BloodPressure             0.19195148 0.1828761   1.000000000     0.2534418
## SkinThickness             0.10451226 0.2213106   0.253441759     1.0000000
## Insulin                   0.07541846 0.5606352   0.098906877     0.2050320
## BMI                      -0.02396266 0.2183979   0.304532005     0.6450924
## DiabetesPedigreeFunction  0.02501807 0.1303609   0.009800844     0.1603260
## Age                       0.66135850 0.3069743   0.305666143     0.2035449
## Outcome                   0.26472192 0.5216071   0.212382415     0.2674751
##                             Insulin         BMI DiabetesPedigreeFunction
## Pregnancies              0.07541846 -0.02396266              0.025018072
## Glucose                  0.56063519  0.21839786              0.130360892
## BloodPressure            0.09890688  0.30453201              0.009800844
## SkinThickness            0.20503196  0.64509235              0.160325991
## Insulin                  1.00000000  0.25326605              0.134679254
## BMI                      0.25326605  1.00000000              0.151761817
## DiabetesPedigreeFunction 0.13467925  0.15176182              1.000000000
## Age                      0.14275534  0.07855708              0.105085468
## Outcome                  0.29171177  0.24712905              0.185973403
##                                 Age   Outcome
## Pregnancies              0.66135850 0.2647219
## Glucose                  0.30697425 0.5216071
## BloodPressure            0.30566614 0.2123824
```

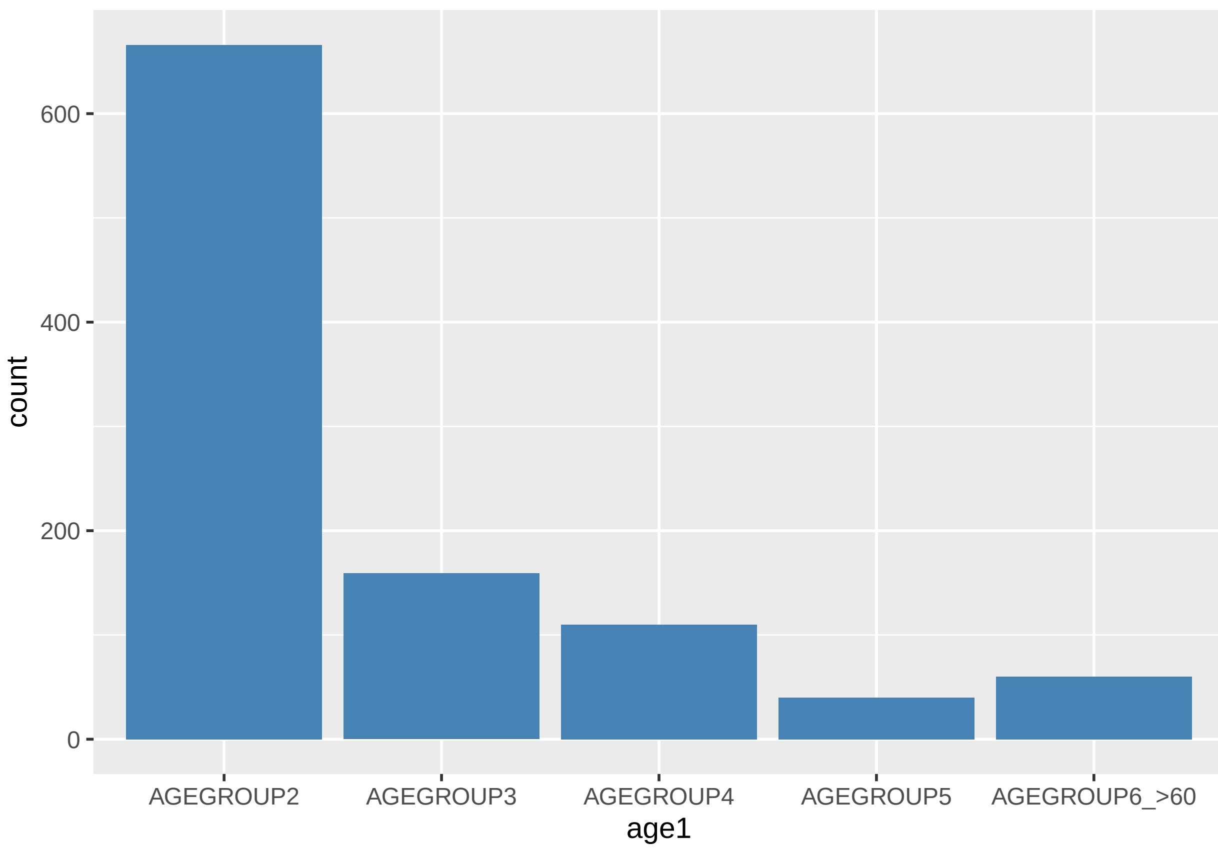
```
## SkinThickness            0.20354493 0.2674751
## Insulin                  0.14275534 0.2917118
## BMI                      0.07855708 0.2471290
## DiabetesPedigreeFunction 0.10508547 0.1859734
## Age                      1.00000000 0.3420619
## Outcome                  0.34206192 1.0000000
```

**Correlation Plot of different variables**

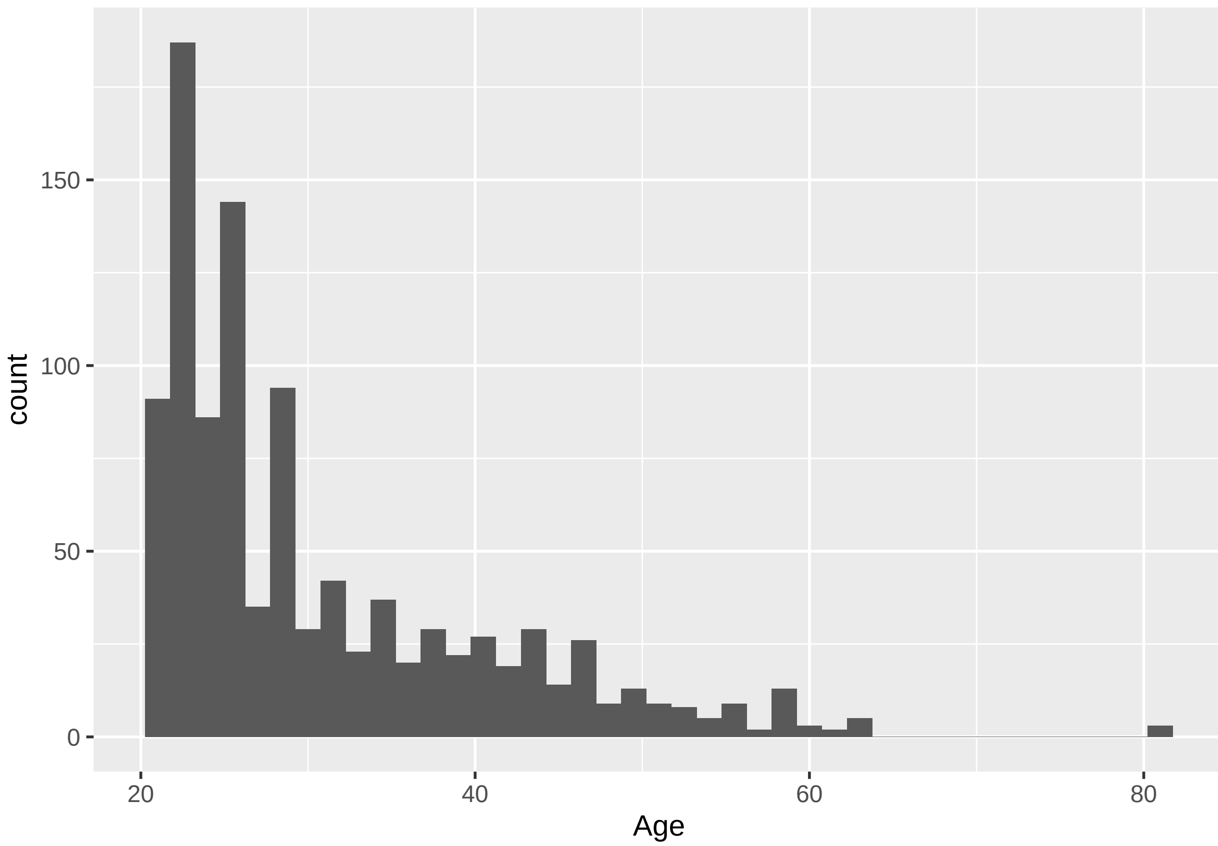


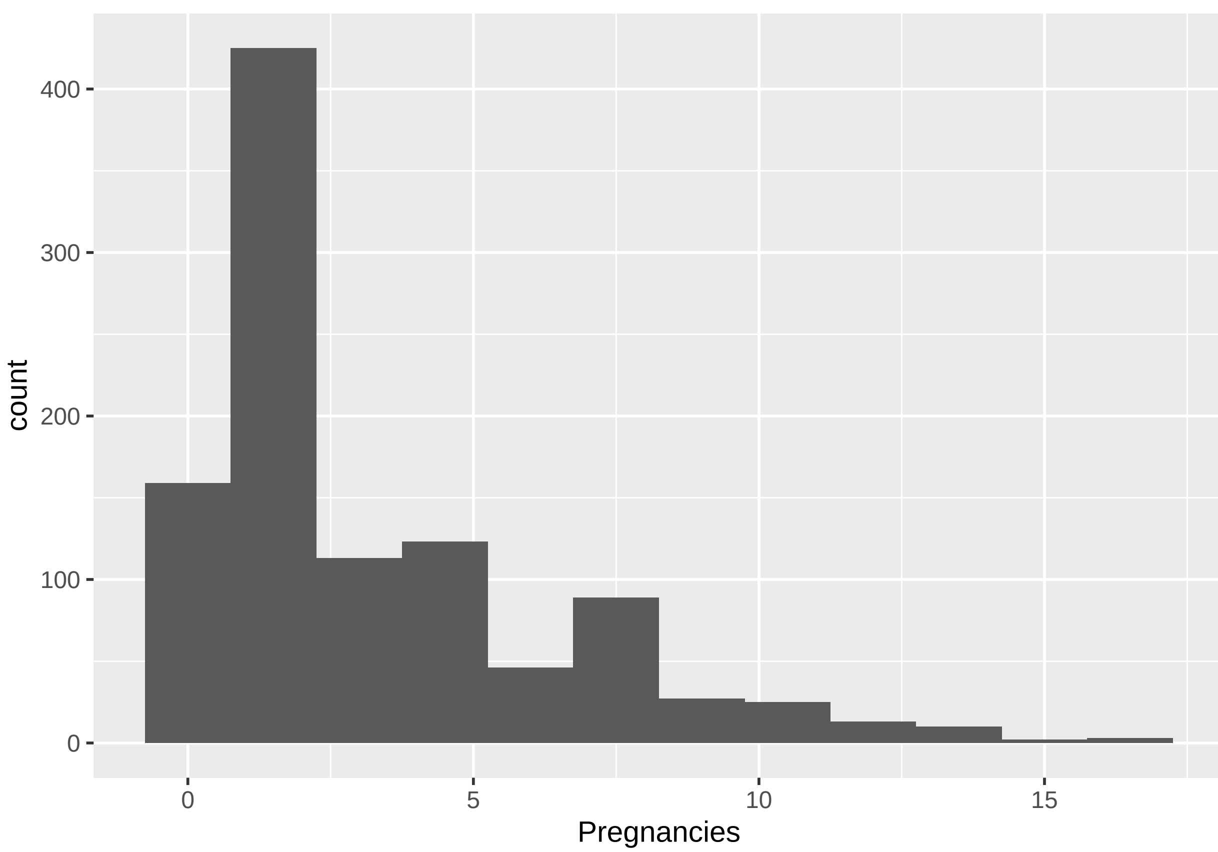

**Age distribution of data with different age groups**

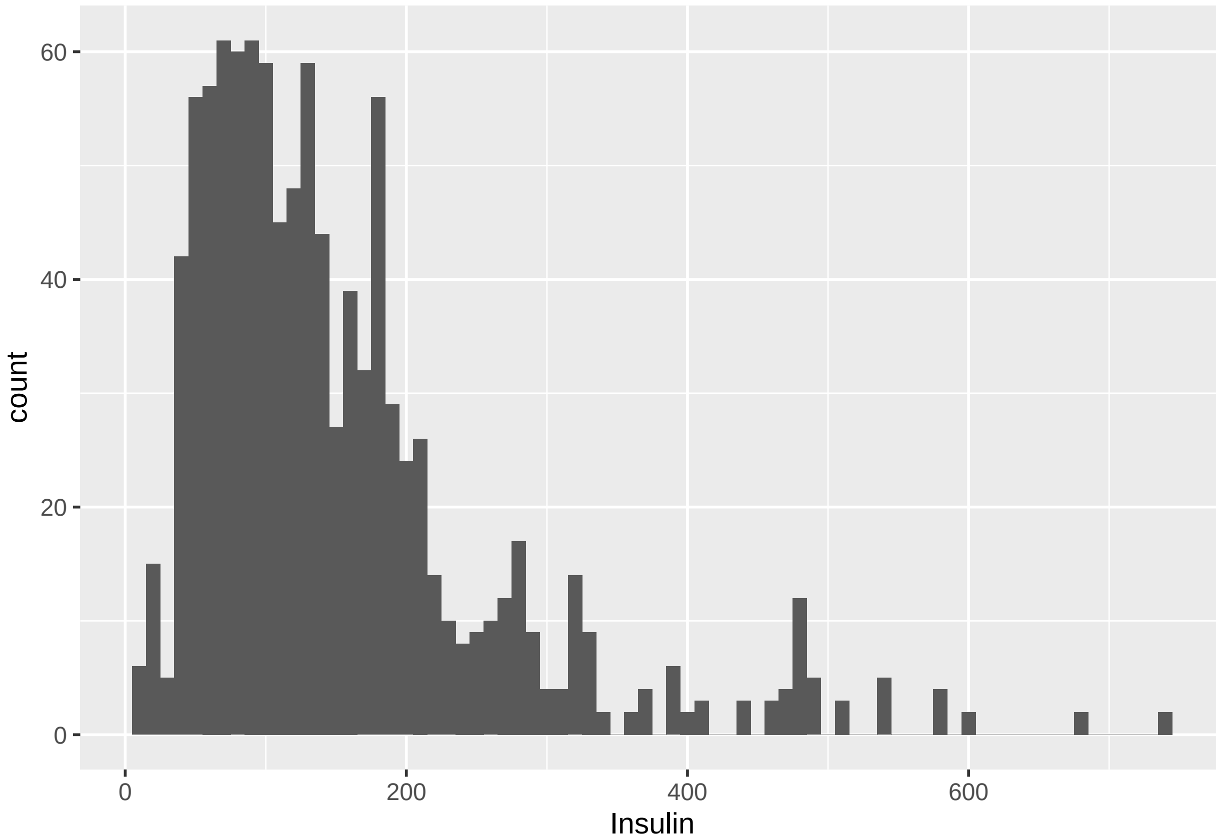
```
##
##      AGEGROUP2      AGEGROUP3      AGEGROUP4      AGEGROUP5  AGEGROUP6_>60
##            666            159            110             40             60
```

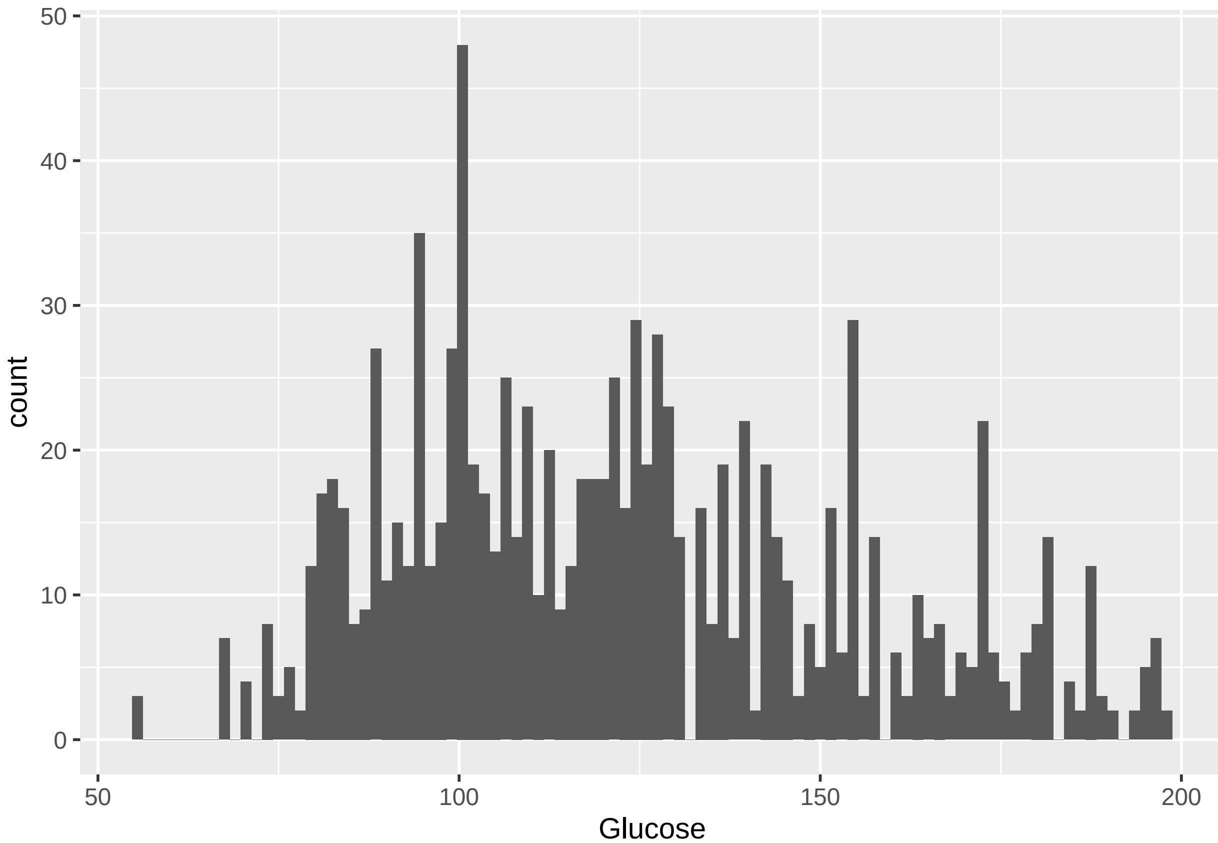

600
400
200
0
count
AGEGROUP2
AGEGROUP3
AGEGROUP4
AGEGROUP5
AGEGROUP6_>60
age1

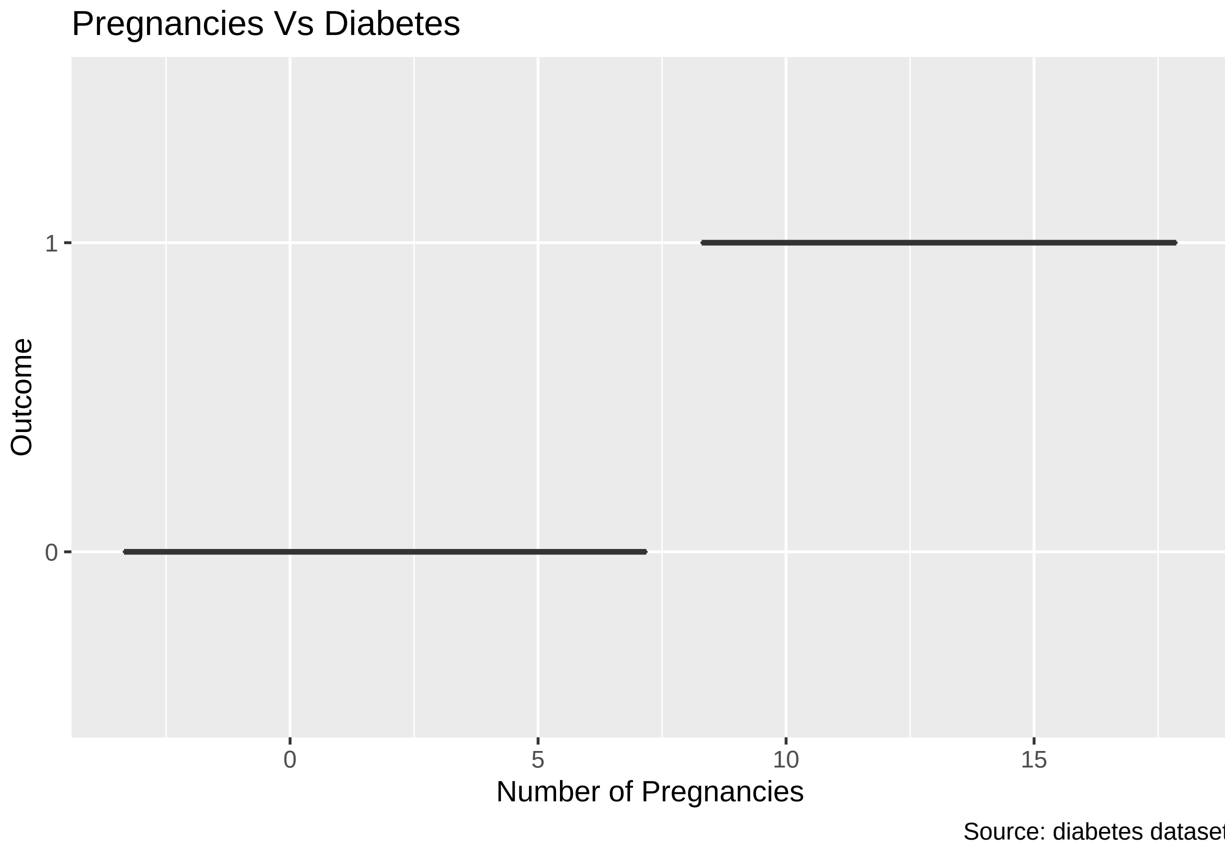
**Data Visualization**

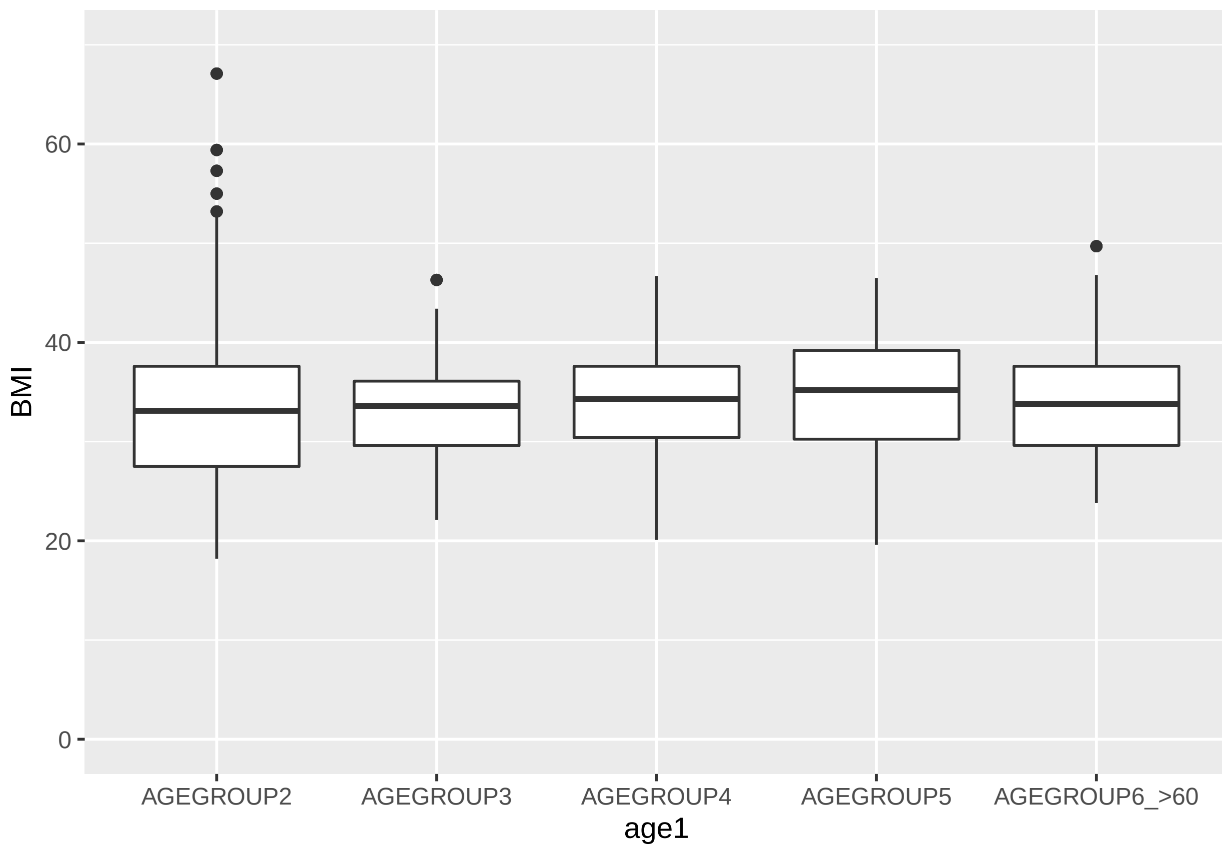



60
40
20
0
count
0
200
400
600
Insulin


50
40
30
20
10
0
count
50
100
150
200
Glucose


Pregnancies Vs Diabetes
1
0
Outcome
0
5
10
15
Number of Pregnancies
Source: diabetes dataset




### Apply Linear regression model on dataset

We will now apply linear regression model on the dataset and will calculate the accuracy of model prediction:

```
##
##   0   1
## 698 337
```

```
## Analysis of Deviance Table
##
## Model: binomial, link: logit
##
## Response: Outcome
##
## Terms added sequentially (first to last)
##
##
##                          Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
## NULL                                       722     914.75
## Pregnancies               1   51.924       721     862.83 5.768e-13 ***
## Glucose                   1  167.237       720     695.59 < 2.2e-16 ***
## BloodPressure             1   16.361       719     679.23 5.235e-05 ***
## SkinThickness             1   16.876       718     662.36 3.990e-05 ***
## Insulin                   1    0.010       717     662.35  0.920004
## BMI                       1    2.280       716     660.07  0.131079
```

```
## DiabetesPedigreeFunction  1   19.769       715     640.30 8.741e-06 ***
## Age                       1    2.193       714     638.10  0.138672
## age1                      4   15.996       710     622.11  0.003024 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

## Start:  AIC=648.11
## Outcome ~ Pregnancies + Glucose + BloodPressure + SkinThickness +
##     Insulin + BMI + DiabetesPedigreeFunction + Age + age1
##
##                            Df Deviance    AIC
## - Insulin                   1   622.63 646.63
## - BMI                       1   623.70 647.70
## <none>                          622.11 648.11
## - Age                       1   624.75 648.75
## - BloodPressure             1   625.85 649.85
## - Pregnancies               1   626.21 650.21
## - SkinThickness             1   627.84 651.84
## - age1                      4   638.10 656.10
## - DiabetesPedigreeFunction  1   638.46 662.46
## - Glucose                   1   708.41 732.41
##
## Step:  AIC=646.63
## Outcome ~ Pregnancies + Glucose + BloodPressure + SkinThickness +
##     BMI + DiabetesPedigreeFunction + Age + age1
##
##                            Df Deviance    AIC
## - BMI                       1   624.02 646.02
## <none>                          622.63 646.63
## - Age                       1   625.08 647.08
## + Insulin                   1   622.11 648.11
## - BloodPressure             1   626.43 648.43
## - Pregnancies               1   626.72 648.72
## - SkinThickness             1   628.22 650.22
## - age1                      4   638.35 654.35
## - DiabetesPedigreeFunction  1   638.66 660.66
## - Glucose                   1   731.82 753.82
##
## Step:  AIC=646.02
## Outcome ~ Pregnancies + Glucose + BloodPressure + SkinThickness +
##     DiabetesPedigreeFunction + Age + age1
##
##                            Df Deviance    AIC
## <none>                          624.02 646.02
## + BMI                       1   622.63 646.63
## - Age                       1   626.93 646.93
## - Pregnancies               1   627.69 647.69
## + Insulin                   1   623.70 647.70
## - BloodPressure             1   629.71 649.71
## - age1                      4   640.24 654.24
## - SkinThickness             1   637.78 657.78
## - DiabetesPedigreeFunction  1   640.70 660.70
## - Glucose                   1   735.24 755.24
```

```
## [1] 0.8108974
```

The top three most relevant features are "Glucose", "BMI" and "Number of times pregnant" because of the low p-values."Insulin" and "Age" appear not statistically significant.From the table of deviance, we can see that adding insulin and age have little effect on the residual deviance.

The below is the confustion matrix of linear regression model:

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 191  38
##          1  21  62
##
##                Accuracy : 0.8109
##                  95% CI : (0.763, 0.8528)
##     No Information Rate : 0.6795
##     P-Value [Acc > NIR] : 1.381e-07
##
##                   Kappa : 0.5454
##
##  Mcnemar's Test P-Value : 0.03725
##
##             Sensitivity : 0.9009
##             Specificity : 0.6200
##          Pos Pred Value : 0.8341
##          Neg Pred Value : 0.7470
##              Prevalence : 0.6795
##          Detection Rate : 0.6122
##    Detection Prevalence : 0.7340
##       Balanced Accuracy : 0.7605
##
##        'Positive' Class : 0
##
```

### Apply K-Nearest model on dataset

We will apply K nearest algorithm/model to predict the accuracy of model for different k-values from 5 to 30 and will draw a plot to see which k values have highest accuracy:

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 174  27
##          1  38  73
##
##                Accuracy : 0.7917
##                  95% CI : (0.7423, 0.8354)
##     No Information Rate : 0.6795
##     P-Value [Acc > NIR] : 7.155e-06
##
```

```
##                  Kappa : 0.5352
##
##  Mcnemar's Test P-Value : 0.2148
##
##             Sensitivity : 0.8208
##             Specificity : 0.7300
##          Pos Pred Value : 0.8657
##          Neg Pred Value : 0.6577
##              Prevalence : 0.6795
##          Detection Rate : 0.5577
##    Detection Prevalence : 0.6442
##       Balanced Accuracy : 0.7754
##
##        'Positive' Class : 0
##
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 182  36
##          1  30  64
##
##                Accuracy : 0.7885
##                  95% CI : (0.7389, 0.8325)
##     No Information Rate : 0.6795
##     P-Value [Acc > NIR] : 1.292e-05
##
##                  Kappa : 0.5065
##
##  Mcnemar's Test P-Value : 0.5383
##
##             Sensitivity : 0.8585
##             Specificity : 0.6400
##          Pos Pred Value : 0.8349
##          Neg Pred Value : 0.6809
##              Prevalence : 0.6795
##          Detection Rate : 0.5833
##    Detection Prevalence : 0.6987
##       Balanced Accuracy : 0.7492
##
##        'Positive' Class : 0
##
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 187  45
##          1  25  55
##
##                Accuracy : 0.7756
##                  95% CI : (0.7252, 0.8207)
##     No Information Rate : 0.6795
```

```
##     P-Value [Acc > NIR] : 0.0001145
##
##                   Kappa : 0.4562
##
##  Mcnemar's Test P-Value : 0.0231510
##
##             Sensitivity : 0.8821
##             Specificity : 0.5500
##          Pos Pred Value : 0.8060
##          Neg Pred Value : 0.6875
##              Prevalence : 0.6795
##          Detection Rate : 0.5994
##    Detection Prevalence : 0.7436
##       Balanced Accuracy : 0.7160
##
##        'Positive' Class : 0
##
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 186  43
##          1  26  57
##
##                Accuracy : 0.7788
##                  95% CI : (0.7287, 0.8237)
##     No Information Rate : 0.6795
##     P-Value [Acc > NIR] : 6.818e-05
##
##                   Kappa : 0.4684
##
##  Mcnemar's Test P-Value : 0.05408
##
##             Sensitivity : 0.8774
##             Specificity : 0.5700
##          Pos Pred Value : 0.8122
##          Neg Pred Value : 0.6867
##              Prevalence : 0.6795
##          Detection Rate : 0.5962
##    Detection Prevalence : 0.7340
##       Balanced Accuracy : 0.7237
##
##        'Positive' Class : 0
##
```

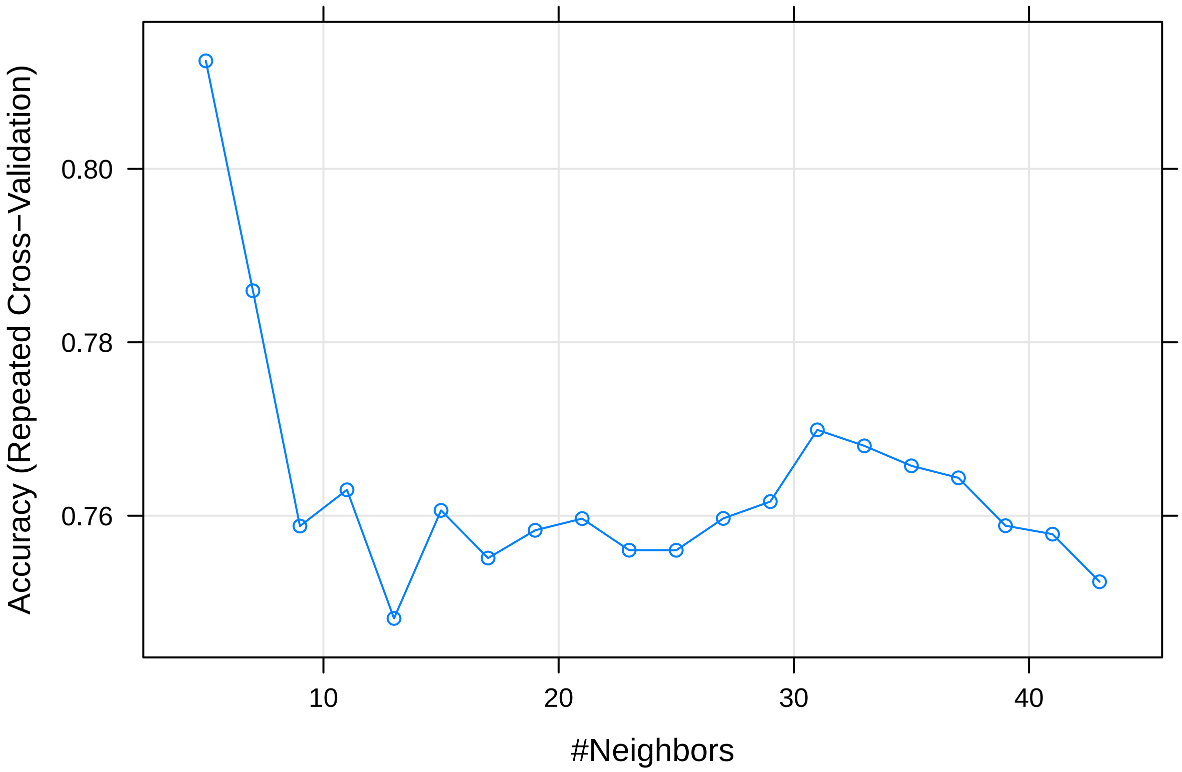
```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 185  40
##          1  27  60
##
##                Accuracy : 0.7853
```

```
##                  95% CI : (0.7355, 0.8295)
##     No Information Rate : 0.6795
##     P-Value [Acc > NIR] : 2.291e-05
##
##                   Kappa : 0.4894
##
##  Mcnemar's Test P-Value : 0.1426
##
##             Sensitivity : 0.8726
##             Specificity : 0.6000
##          Pos Pred Value : 0.8222
##          Neg Pred Value : 0.6897
##              Prevalence : 0.6795
##          Detection Rate : 0.5929
##    Detection Prevalence : 0.7212
##       Balanced Accuracy : 0.7363
##
##        'Positive' Class : 0
##
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 189  38
##          1  23  62
##
##                Accuracy : 0.8045
##                  95% CI : (0.7561, 0.847)
##     No Information Rate : 0.6795
##     P-Value [Acc > NIR] : 5.568e-07
##
##                   Kappa : 0.5326
##
##  Mcnemar's Test P-Value : 0.07305
##
##             Sensitivity : 0.8915
##             Specificity : 0.6200
##          Pos Pred Value : 0.8326
##          Neg Pred Value : 0.7294
##              Prevalence : 0.6795
##          Detection Rate : 0.6058
##    Detection Prevalence : 0.7276
##       Balanced Accuracy : 0.7558
##
##        'Positive' Class : 0
##
```

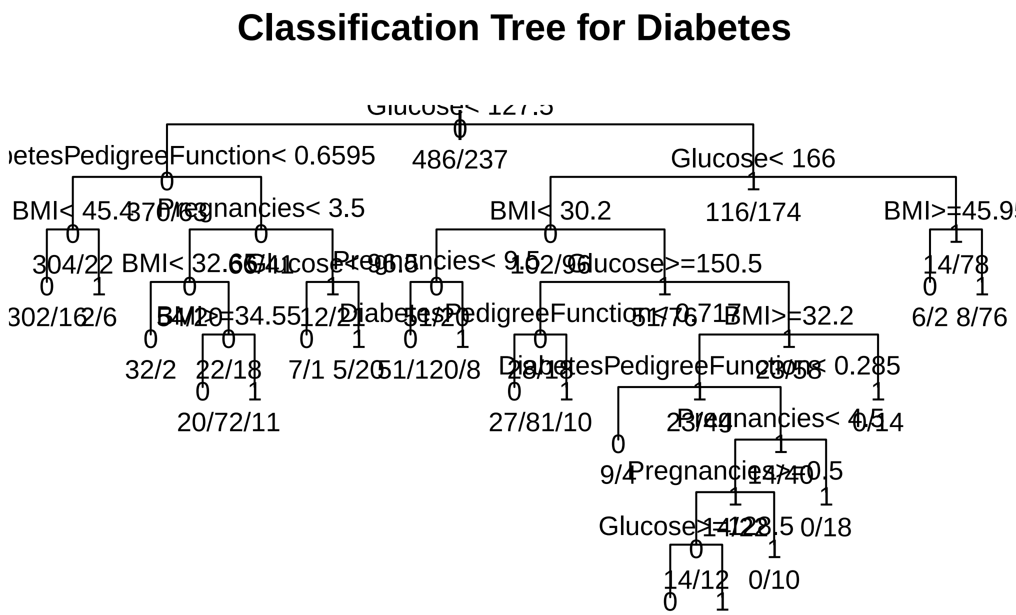
```
## k-Nearest Neighbors
##
## 723 samples
##   8 predictor
##   2 classes: '0', '1'
##
```

```
## No pre-processing
## Resampling: Cross-Validated (10 fold, repeated 3 times)
## Summary of sample sizes: 651, 652, 651, 650, 650, 651, ...
## Resampling results across tuning parameters:
##
##   k   Accuracy   Kappa
##    5  0.8124499  0.5559094
##    7  0.7859435  0.4909100
##    9  0.7588060  0.4237508
##   11  0.7629856  0.4314009
##   13  0.7481570  0.4022776
##   15  0.7606135  0.4283875
##   17  0.7551150  0.4214702
##   19  0.7583177  0.4305067
##   21  0.7596684  0.4253694
##   23  0.7560219  0.4186647
##   25  0.7560159  0.4177222
##   27  0.7596877  0.4188850
##   29  0.7616287  0.4255454
##   31  0.7698990  0.4437165
##   33  0.7680537  0.4379169
##   35  0.7657514  0.4344516
##   37  0.7643688  0.4287118
##   39  0.7588452  0.4103666
##   41  0.7578741  0.4069077
##   43  0.7523818  0.3920548
##
## Accuracy was used to select the optimal model using the largest value.
## The final value used for the model was k = 5.
```




From the plot we can see that the model with k value=5 had the highest accuracy of 81%.

**Apply Decision Tree algorithm on datasets**




```
##
## treePred   0   1
##        0 204  37
##        1   8  63
```

```
## [1] 0.8557692
```

Decision tree structure by using all features and Pima Indians dataset. From this figure, we can find in this method glucose as the root node, which can indicate the index has the highest information gain and insulin and age play important roles in this method.The above results show that the accuracy obtained through decision tree algorithms is 85%.

## CONCLUSION

Machine learning has the great ability to revolutionize the diabetes risk prediction with the help of advanced computational methods and availability of large amount of epidemiological and genetic diabetes risk dataset. Detection of diabetes in its early stages is the key for treatment. This work has described a machine learning approach to predicting diabetes levels. The technique may also help researchers to develop an accurate and effective tool that will reach at the table of clinicians to help them make better decision about the disease status.In this project, I compared the performance of Logistic Regression,KNN algorithms and Decision Tree algorithms and found that Decision tree performed better on this standard, unaltered dataset. However, there are things we can do to improve the generalization performance.

**References:**


1) https://www.academia.edu/36963831/Diabetes_Prediction_Using_Machine_Learning_Techniques
2) https://www.frontiersin.org/articles/10.3389/fgene.2018.00515/full
3) https://www.kaggle.com/paultimothymooney/predict-diabetes-with-r-starter-kernel/data