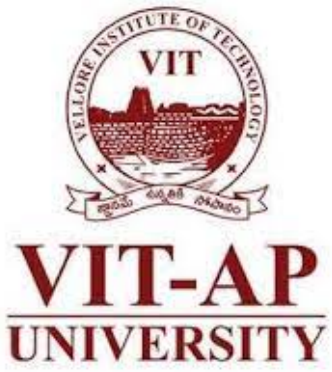


ADVANCE DATA ANALYTICS
PROJECT REPORT

By

SREEDATH SOMI SETTY-18BCE7294
S SRIGOKUL BHARADWAJ-18BCE7218
A.B.V.K. SOUMITH-18BCD7027

Advanced data Analytics:

The dataset we used is Diabetes(by Pima india)

https://www.kaggle.com/uciml/pima-indians-diabetes-database

he datasets consist of several medical predictor (independent) variables and one target (dependent) variable, Outcome. Independent variables include the number of pregnancies the patient has had, their BMI, insulin level, age, and so on.

Reason to take this dataset:

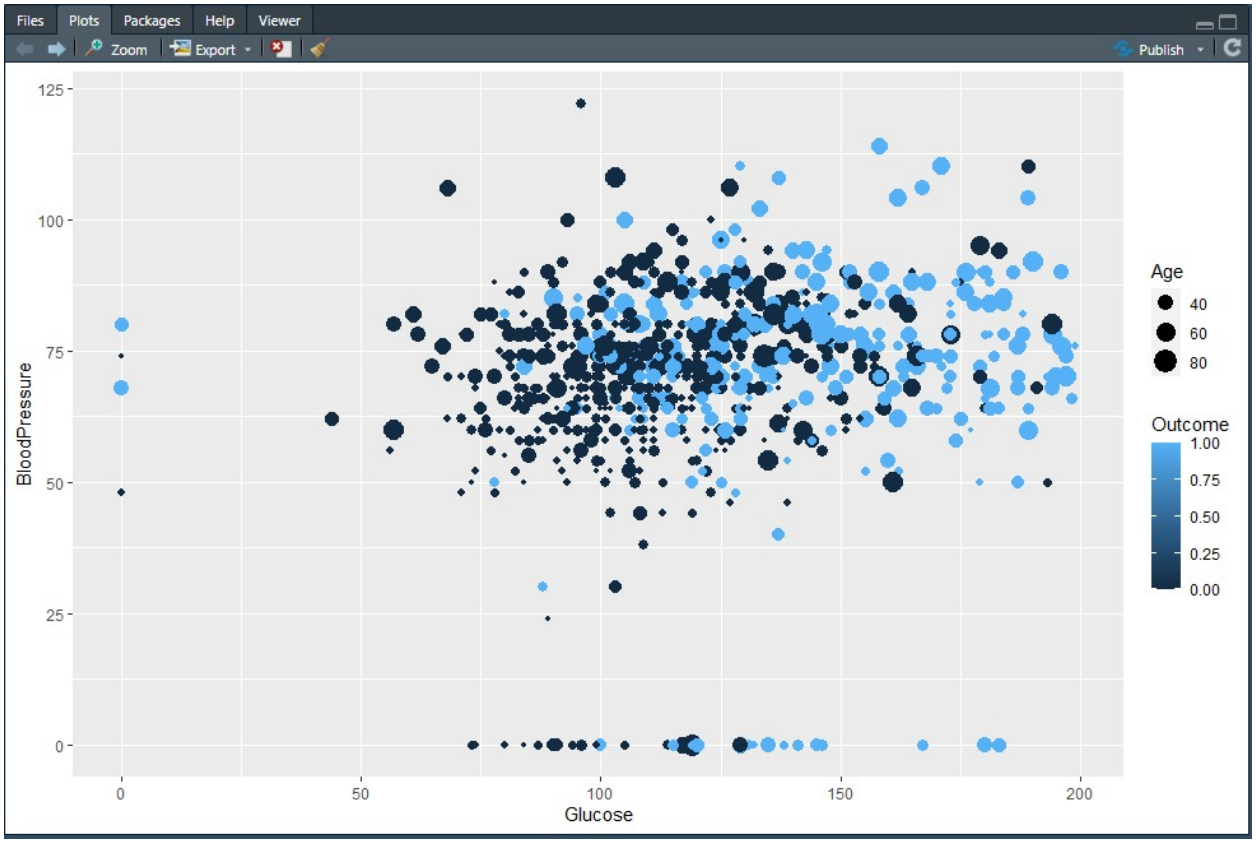
Has 768 rows and 9 columns and is suitable for almost all data analysis algorithms and models.

Libraries used:

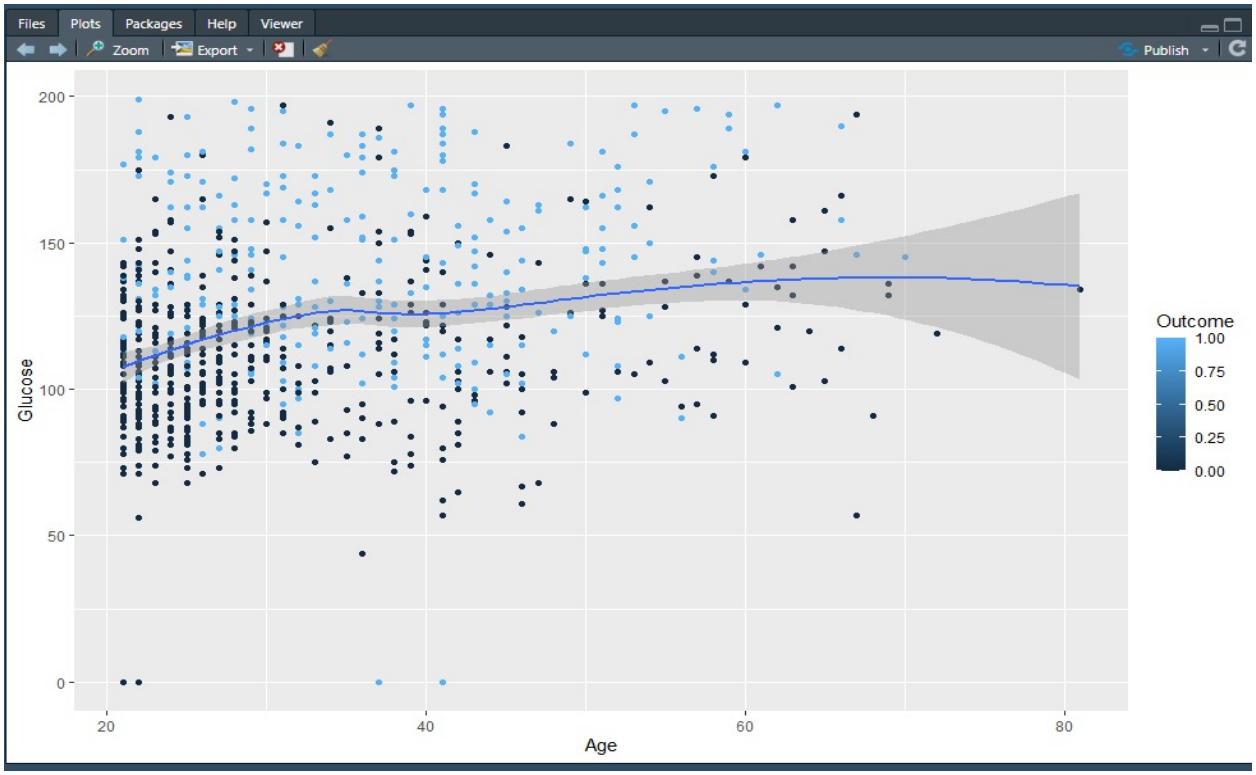
library(factoextra)
library(tidyverse)
library(ggplot2)
library(caret)
library(dplyr)
library(BSDA)
library("Hmisc")
library(corrplot)-correlation
library(readr)
library("e1071")-svm
library("partykit")-Conditional Inference tree
library(cluster)
library(factoextra)

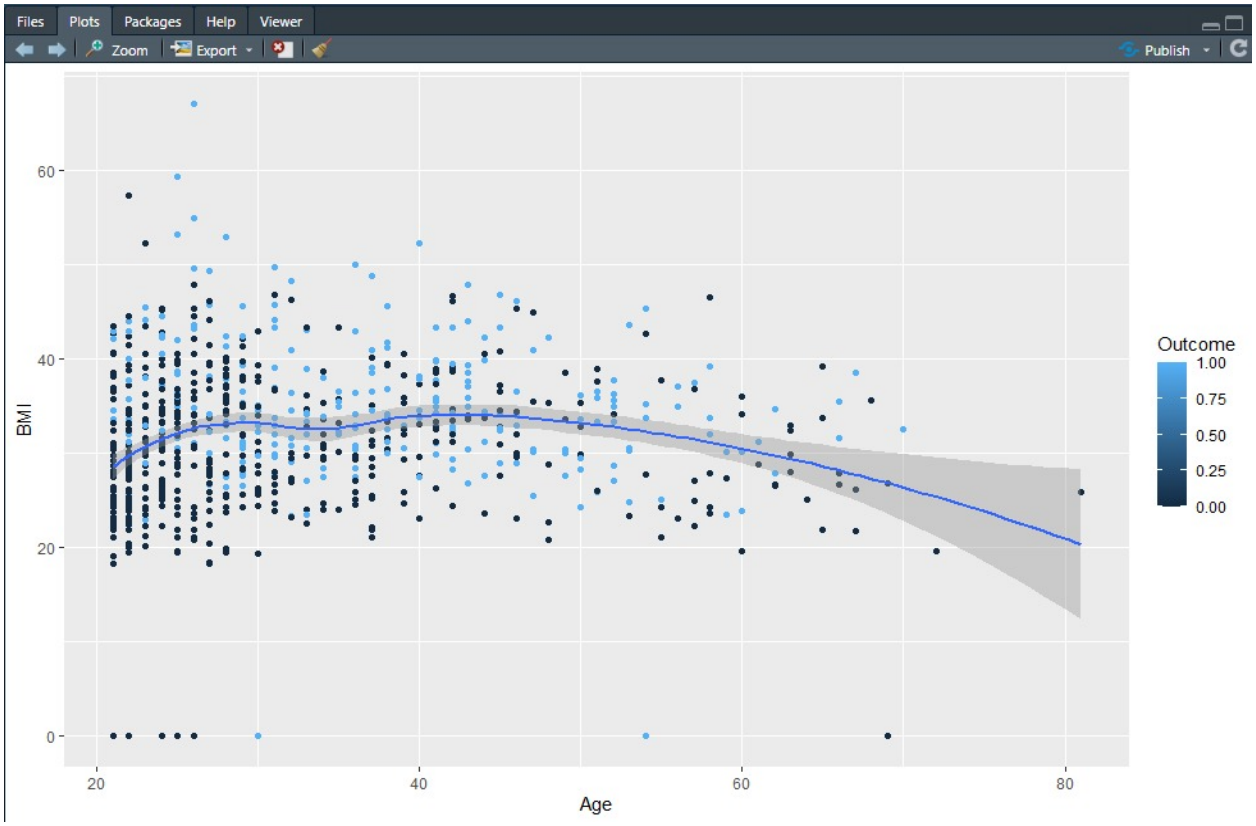
#Preprocessing the data
preProcess_range_modeltr <- preProcess(data, method='range')
trainData <- predict(preProcess_range_modeltr,newdata = data)
levels(trainData$Outcome) <- c("Class0","Class1")
summary(trainData)

ggplot(data,aes(x = Glucose, y = BloodPressure, col = Outcome ,size = Age))+geom_point()



```
C:/Users/SREEDATH/Desktop/ADA_Project/
> p3 <- ggplot(data, aes(x = Age, y=Glucose ,col =Outcome))+geom_point()
> p3+geom_smooth(method = "loess")
`geom_smooth()` using formula 'y ~ x'
>
> p4 <- ggplot(data, aes(x = Age, y=BMI ,col =Outcome))+geom_point()
> p4+geom_smooth(method = "loess")
`geom_smooth()` using formula 'y ~ x'
```




Files
Plots
Packages
Help
Viewer
Zoom
Export
Publish
BMI
60
40
20
0
20
40
60
80
Age
Outcome
1.00
0.75
0.50
0.25
0.00

HYPOTHESIS TESTING

T-TEST

1. Two-sample Assuming Unequal Variances

EXCEL:

t-Test: Two-Sample Assuming Unequal Variances		
	26.6	*33.6*
Mean	30.31162325	35.14831461
Variance	59.22500519	52.94002534
Observations	499	267
Hypothesized Mean Differe	0	
df	571	
t Stat	-8.590985212	
P(T<=t) one-tail	4.12279E-17	
t Critical one-tail	1.647526586	
P(T<=t) two-tail	8.24558E-17	
t Critical two-tail	1.964127246	

R: t-test for BMI and outcome

```
C:/Users/SREEDATH/Desktop/ADA_Project/
> #t-test
>
> #t- test for BMI and Outcome
>
> df1 <- data%>%select(BMI,Outcome)%>%filter(Outcome == '0')
> df2 <- data%>%select(BMI,Outcome)%>%filter(Outcome == '1')
> dfbmi1<- df1$BMI
> dfbmi2<- df2$BMI
> t.test(dfbmi1,dfbmi2)

        Welch Two Sample t-test

data:  dfbmi1 and dfbmi2
t = -8.6193, df = 573.47, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -5.940864 -3.735811
sample estimates:
mean of x mean of y
 30.30420  35.14254
```

2. Two-Sample Assuming Unequal Variances Glucose and out come EXCEL:

t-Test: Two-Sample Assuming Unequal Variances Glucose and out come		
	85	*148*
Mean	110.0300601	141.2322097
Variance	683.4790143	1023.803019
Observations	499	267
Hypothesized Mean	0	
df	459	
t Stat	-13.6775704	
P(T<=t) one-tail	2.90103E-36	
t Critical one-tail	1.648180137	
P(T<=t) two-tail	5.80206E-36	
t Critical two-tail	1.965145755	

R: t-test for Glucose and outcome

```
C:/Users/SREEDATH/Desktop/ADA_Project/
> #t test for Glucose and Outcome
>
> df1glu <- data%>%select(Glucose,Outcome)%>%filter(Outcome == '0')
> df2glu <- data%>%select(Glucose,Outcome)%>%filter(Outcome == '1')
> dfglu1<- df1glu$Glucose
> dfglu2<- df2glu$Glucose
> t.test(dfglu1,dfglu2)

        Welch Two Sample t-test

data:  dfglu1 and dfglu2
t = -13.752, df = 461.33, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -35.74707 -26.80786
sample estimates:
mean of x mean of y
 109.9800  141.2575
```

3. Two-Sample Assuming Unequal Variences DiabetesPedigreeFunction and Output

EXCEL:

t-Test: Two-Sample Assuming Unequal Variances DiabetesPedigreeFunction and output

	0.351	*0.627*
Mean	0.429891784	0.550213483
Variance	0.089619169	0.139147011
Observations	499	267
Hypothesized Mean Difference	0	
df	452	
t Stat	-4.545306927	
P(T<=t) one-tail	3.52555E-06	
t Critical one-tail	1.64823176	
P(T<=t) two-tail	7.05111E-06	
t Critical two-tail	1.965226215	

R; t-test for DiabeticPedigreeFunction

```
C:/Users/SREEDATH/Desktop/ADA_Project/
> #t-test for DiabetesPedigreeFunction
>
> df1pf <- data%>%select(DiabetesPedigreeFunction,Outcome)%>%filter(Outcome == '0')
> df2pf <- data%>%select(DiabetesPedigreeFunction,Outcome)%>%filter(Outcome == '1')
> dfpf1<- df1pf$DiabetesPedigreeFunction
> dfpf2<- df2pf$DiabetesPedigreeFunction
> t.test(dfpf1,dfpf2)

        Welch Two Sample t-test

data:  dfpf1 and dfpf2
t = -4.5768, df = 454.51, p-value = 6.1e-06
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.17262065 -0.06891135
sample estimates:
mean of x mean of y
 0.429734  0.550500
```

Z-TEST

EXCEL:

z-Test: Two Sample for Means		
	Variable 1	*Variable 2*
Mean	30.3042	35.14253731
Known Variance	0.5	0.5
Observations	500	268
Hypothesized Mean Difference	0	
z	-127.3822045	
P(Z<=z) one-tail	-4.9006133	
z Critical one-tail	1.281551566	
P(Z<=z) two-tail	-4.776025	
z Critical two-tail	1.644853627	

R: z-test for BMI and Outcome

```
C:/Users/SREEDATH/Desktop/ADA_Project/
> #z-test for BMI and Outcome
> x<-dfbmi1
> y<-dfbmi2
> zlol<-z.test(x,sigma.x=0.5,y,sigma.y=0.5,conf.level = 0.90)
> zlol

        Two-sample z-Test

data:  x and y
z = -127.82, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
90 percent confidence interval:
 -4.900600 -4.776075
sample estimates:
mean of x mean of y
 30.30420  35.14254
```

VARIOUS CORRELATION MEASURES

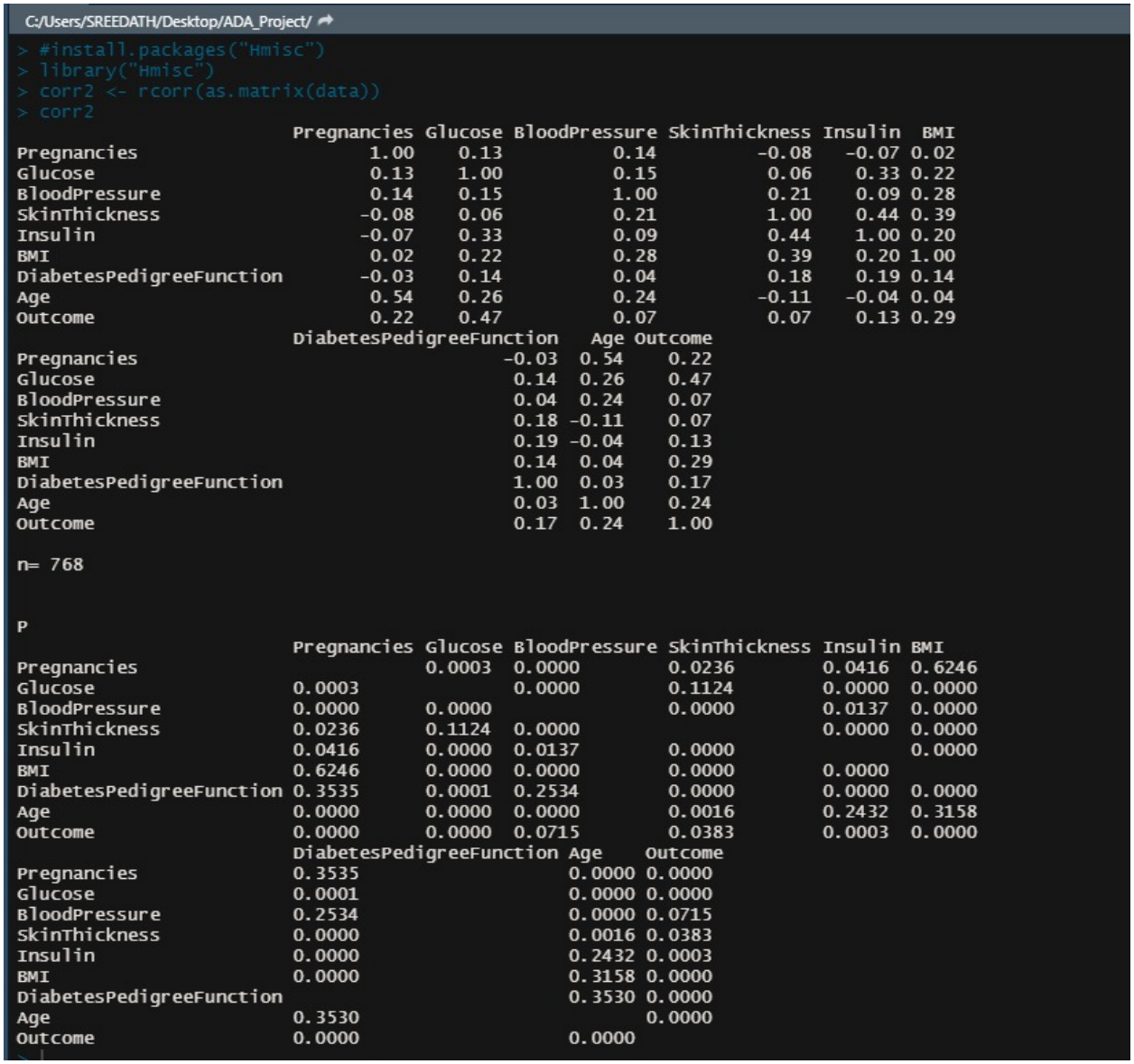
EXCEL:

Covarriance	*Pregnancies*	*Glucose*	*BloodPressure*	*SkinThickness*	*Insulin*	*BMI*	*DiabetesPedigreeFunction*	*Age*
Pregnancies	11.33927239							
Glucose	13.92897034	1020.917						
BloodPressure	9.20254008	94.308	374.1594493					
SkinThickness	-4.384324816	29.20111	63.94602458	254.1419				
Insulin	-28.51804945	1219.346	198.120107	801.934394	13263.89			
BMI	0.469162496	55.65443	42.94869944	49.30958049	179.5411	62.07905		
DiabetesPedigreeFunction	-0.03737724	1.45298	0.264292994	0.970869744	7.057479	0.366926	0.109635697	
Age	21.54253303	98.95379	54.4524587	-21.35318332	-57.0689	3.355954	0.130601412	138.123

Correlation between patients with diabetes negative	*Pregnancies*	*Glucose*	*BloodPressure*	*SkinThickness*	*Insulin*	*BMI*	*DiabetesPedigreeFunction*	*Age*
Pregnancies	1							
Glucose	-0.054591456	1						
BloodPressure	0.126962706	0.068699	1					
SkinThickness	-0.079164979	0.037618	0.225247183	1				
Insulin	-0.078562597	0.261368	0.089366746	0.45655799	1			
BMI	-0.159070938	0.050418	0.133950969	0.312058274	0.055112	1		
DiabetesPedigreeFunction	-0.069194984	0.026474	0.034522373	0.273899938	0.101565	0.136761	1	
Age	0.444987328	0.098565	0.262684235	-0.092011848	0.023944	-0.18801	-0.088118752	1

Correlation between patients with diabetes positive	*Pregnancies*	*Glucose*	*BloodPressure*	*SkinThickness*	*Insulin*	*BMI*	*DiabetesPedigreeFunction*	*Age*
Pregnancies	1							
Glucose	0.098684522	1						
BloodPressure	0.133095858	0.19279456	1					
SkinThickness	-0.1183399	0.01601513	0.187071619	1				
Insulin	-0.13198606	0.35295698	0.074626484	0.412789817	1			
BMI	0.016495068	0.131749	0.363178092	0.438605941	0.254202	1		
DiabetesPedigreeFunction	-0.07995055	0.09554795	0.027291539	0.095181158	0.227385	0.070664	1	
Age	0.572776163	0.22801775	0.214693877	-0.163788322	-0.14923	0.03607	0.041665041	1

R:



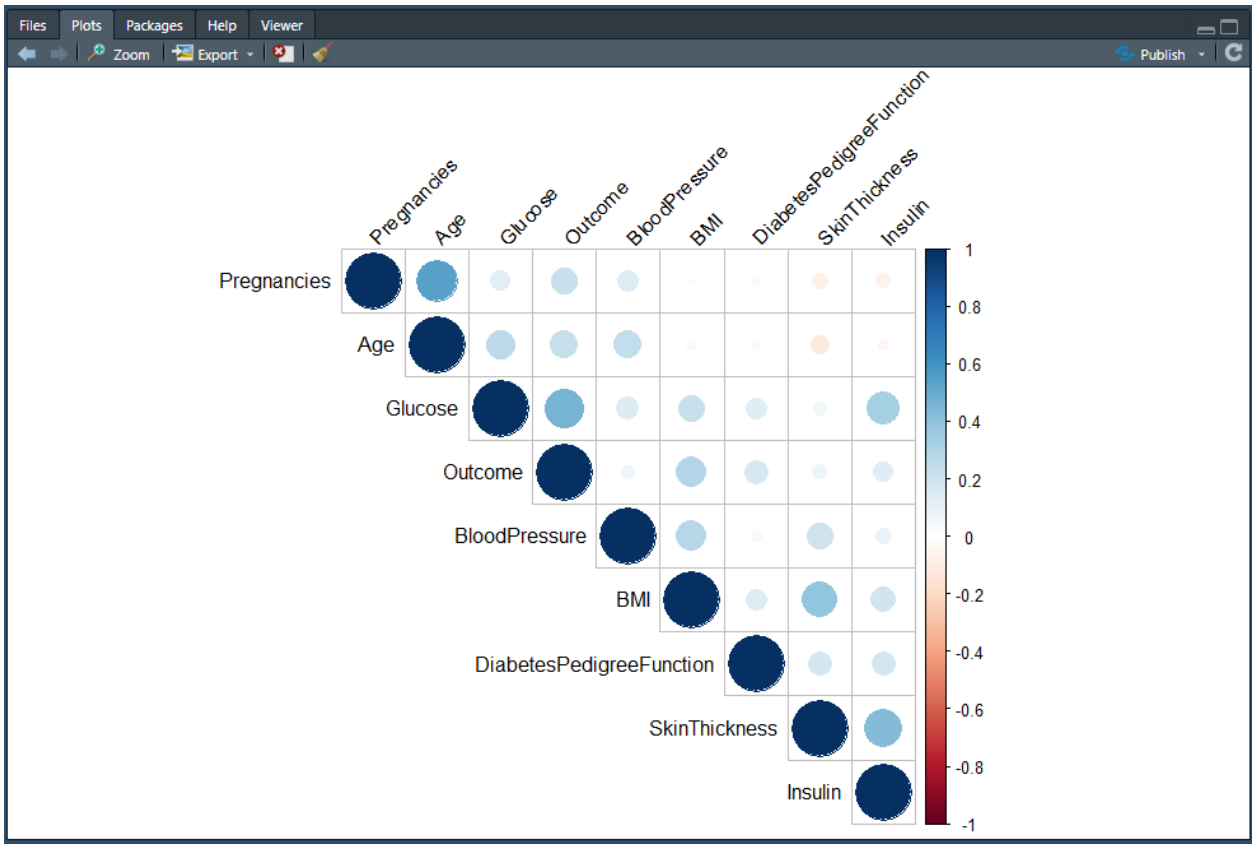
```
C:/Users/SREEDATH/Desktop/ADA_Project/
> #install.packages("Hmisc")
> library("Hmisc")
> corr2 <- rcorr(as.matrix(data))
> corr2
                          Pregnancies Glucose BloodPressure SkinThickness Insulin  BMI
Pregnancies                      1.00    0.13          0.14         -0.08   -0.07 0.02
Glucose                          0.13    1.00          0.15          0.06    0.33 0.22
BloodPressure                    0.14    0.15          1.00          0.21    0.09 0.28
SkinThickness                   -0.08    0.06          0.21          1.00    0.44 0.39
Insulin                         -0.07    0.33          0.09          0.44    1.00 0.20
BMI                              0.02    0.22          0.28          0.39    0.20 1.00
DiabetesPedigreeFunction        -0.03    0.14          0.04          0.18    0.19 0.14
Age                              0.54    0.26          0.24         -0.11   -0.04 0.04
Outcome                          0.22    0.47          0.07          0.07    0.13 0.29
                          DiabetesPedigreeFunction   Age Outcome
Pregnancies                                -0.03  0.54    0.22
Glucose                                     0.14  0.26    0.47
BloodPressure                               0.04  0.24    0.07
SkinThickness                               0.18 -0.11    0.07
Insulin                                     0.19 -0.04    0.13
BMI                                         0.14  0.04    0.29
DiabetesPedigreeFunction                    1.00  0.03    0.17
Age                                         0.03  1.00    0.24
Outcome                                     0.17  0.24    1.00

n= 768


P
                          Pregnancies Glucose BloodPressure SkinThickness Insulin BMI
Pregnancies                           0.0003  0.0000        0.0236        0.0416  0.6246
Glucose                   0.0003              0.0000        0.1124        0.0000  0.0000
BloodPressure             0.0000      0.0000                0.0000        0.0137  0.0000
SkinThickness             0.0236      0.1124  0.0000                      0.0000  0.0000
Insulin                   0.0416      0.0000  0.0137        0.0000                0.0000
BMI                       0.6246      0.0000  0.0000        0.0000        0.0000
DiabetesPedigreeFunction  0.3535      0.0001  0.2534        0.0000        0.0000  0.0000
Age                       0.0000      0.0000  0.0000        0.0016        0.2432  0.3158
Outcome                   0.0000      0.0000  0.0715        0.0383        0.0003  0.0000
                          DiabetesPedigreeFunction Age    Outcome
Pregnancies               0.3535                   0.0000 0.0000
Glucose                   0.0001                   0.0000 0.0000
BloodPressure             0.2534                   0.0000 0.0715
SkinThickness             0.0000                   0.0016 0.0383
Insulin                   0.0000                   0.2432 0.0003
BMI                       0.0000                   0.3158 0.0000
DiabetesPedigreeFunction                           0.3530 0.0000
Age                       0.3530                          0.0000
Outcome                   0.0000                   0.0000
```

Extracting the correlation coefficients

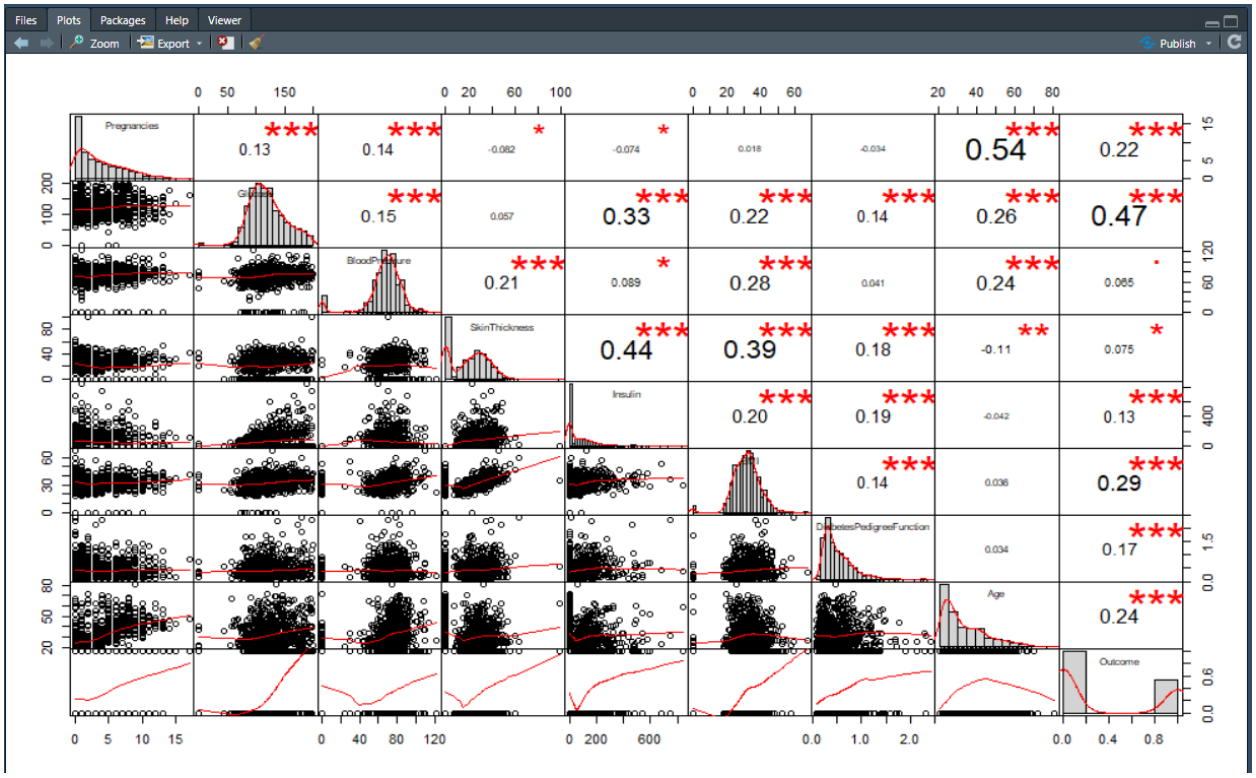
```
C:/Users/SREEDATH/Desktop/ADA_Project/
> # Extract the correlation coefficients
> corr2$r
                         Pregnancies    Glucose BloodPressure SkinThickness     Insulin        BMI
Pregnancies              1.00000000 0.12945867    0.14128198   -0.08167177 -0.07353461 0.01768309
Glucose                  0.12945867 1.00000000    0.15258959    0.05732789  0.33135711 0.22107107
BloodPressure            0.14128198 0.15258959    1.00000000    0.20737054  0.08893338 0.28180529
SkinThickness           -0.08167177 0.05732789    0.20737054    1.00000000  0.43678257 0.39257320
Insulin                 -0.07353461 0.33135711    0.08893338    0.43678257  1.00000000 0.19785906
BMI                      0.01768309 0.22107107    0.28180529    0.39257320  0.19785906 1.00000000
DiabetesPedigreeFunction -0.03352267 0.13733730    0.04126495    0.18392757  0.18507093 0.14064695
Age                      0.54434123 0.26351432    0.23952795   -0.11397026 -0.04216295 0.03624187
Outcome                  0.22189815 0.46658140    0.06506836    0.07475223  0.13054795 0.29269466
                         DiabetesPedigreeFunction         Age    Outcome
Pregnancies                           -0.03352267  0.54434123 0.22189815
Glucose                                0.13733730  0.26351432 0.46658140
BloodPressure                          0.04126495  0.23952795 0.06506836
SkinThickness                          0.18392757 -0.11397026 0.07475223
Insulin                                0.18507093 -0.04216295 0.13054795
BMI                                    0.14064695  0.03624187 0.29269466
DiabetesPedigreeFunction               1.00000000  0.03356131 0.17384407
Age                                    0.03356131  1.00000000 0.23835598
Outcome                                0.17384407  0.23835598 1.00000000
> # Extract p-values
> corr2$P
                          Pregnancies      Glucose BloodPressure SkinThickness      Insulin          BMI
Pregnancies                        NA 3.219491e-04  8.541846e-05  2.360795e-02 4.162094e-02 6.246376e-01
Glucose                  3.219491e-04           NA  2.169507e-05  1.124141e-01 0.000000e+00 5.891412e-10
BloodPressure            8.541846e-05 2.169507e-05            NA  6.606687e-09 1.368350e-02 1.776357e-15
SkinThickness            2.360795e-02 1.124141e-01  6.606687e-09            NA 0.000000e+00 0.000000e+00
Insulin                  4.162094e-02 0.000000e+00  1.368350e-02  0.000000e+00           NA 3.219695e-08
BMI                      6.246376e-01 5.891412e-10  1.776357e-15  0.000000e+00 3.219695e-08          NA
DiabetesPedigreeFunction 3.535346e-01 1.345878e-04  2.533744e-01  2.856179e-07 2.402264e-07 9.197970e-05
Age                      0.000000e+00 1.150191e-13  1.752065e-11  1.558278e-03 2.431822e-01 3.158330e-01
Outcome                  5.065126e-10 0.000000e+00  7.151390e-02  3.834770e-02 2.861865e-04 0.000000e+00
                         DiabetesPedigreeFunction          Age      Outcome
Pregnancies                          3.535346e-01 0.000000e+00 5.065126e-10
Glucose                              1.345878e-04 1.150191e-13 0.000000e+00
BloodPressure                        2.533744e-01 1.752065e-11 7.151390e-02
SkinThickness                        2.856179e-07 1.558278e-03 3.834770e-02
Insulin                              2.402264e-07 2.431822e-01 2.861865e-04
BMI                                  9.197970e-05 3.158330e-01 0.000000e+00
DiabetesPedigreeFunction                       NA 3.529797e-01 1.254607e-06
Age                                  3.529797e-01           NA 2.209966e-11
Outcome                              1.254607e-06 2.209966e-11           NA
>
```

```
> symnum(corr, abbr.colnames = FALSE)
                         Pregnancies Glucose BloodPressure SkinThickness Insulin BMI
Pregnancies              1
Glucose                             1
BloodPressure                               1
SkinThickness                                         1
Insulin                             .                 .             1
BMI                                                   .                     1
DiabetesPedigreeFunction
Age                      .
Outcome                             .
                         DiabetesPedigreeFunction Age Outcome
Pregnancies
Glucose
BloodPressure
SkinThickness
Insulin
BMI
DiabetesPedigreeFunction 1
Age                                           1
Outcome                                           1
attr(,"legend")
[1] 0 ' ' 0.3 '.' 0.6 ',' 0.8 '+' 0.9 '*' 0.95 'B' 1
>
```

```
C:/Users/SREEDATH/Desktop/ADA_Project/
> #install.packages("corrplot")
>
> library(corrplot)
>
> # correlogram plot
> corrplot(corr, type = "upper", order = "hclust",
+          tl.col = "black", tl.srt = 45)
>
```

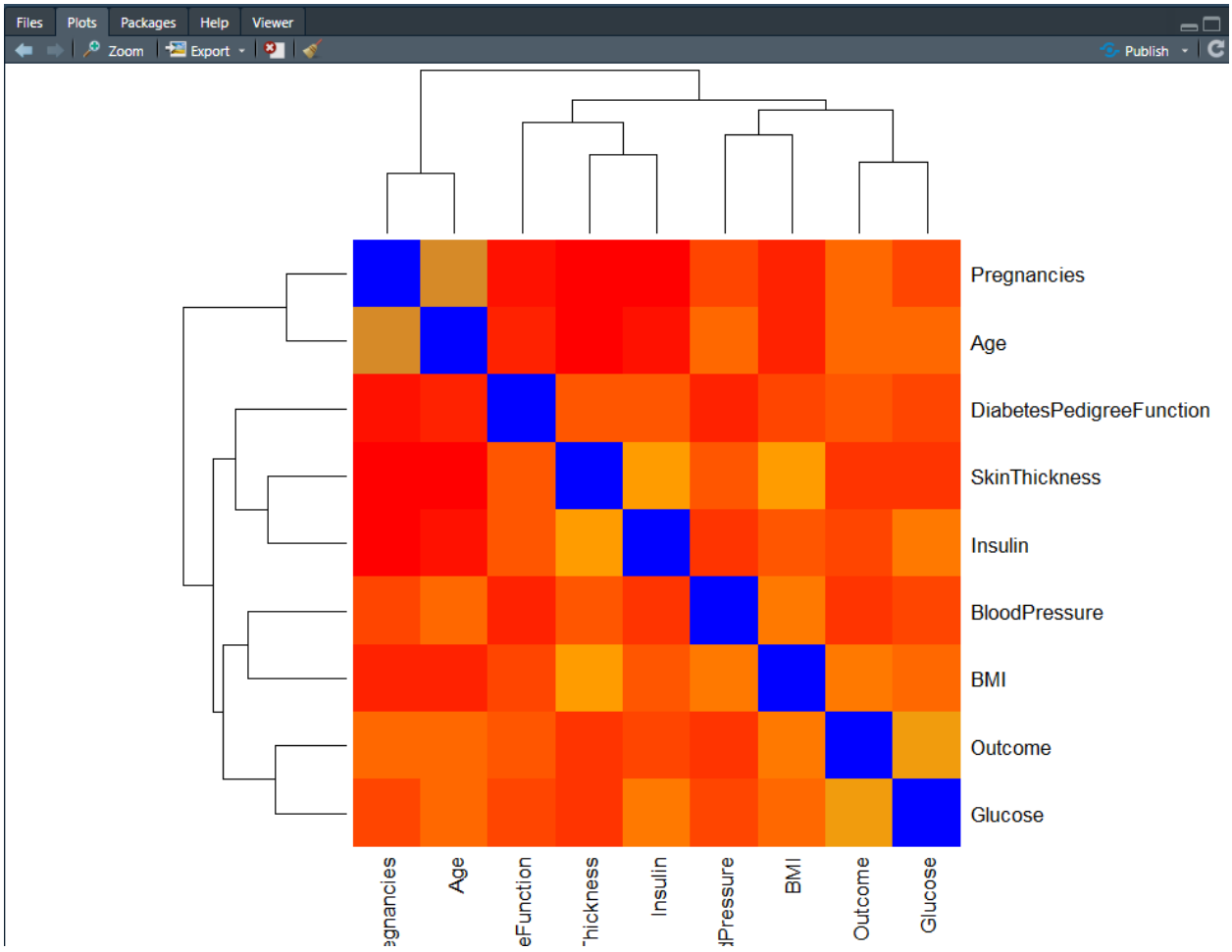


```
C:/Users/SREEDATH/Desktop/ADA_Project/
> corrplot(corr, type = "upper", order = "hclust",
+          tl.col = "black", tl.srt = 45)
> #Use chart.Correlation():
> #install.packages("PerformanceAnalytics")
> library("PerformanceAnalytics")
> #Let's produce the matrix of scatterplots and histogram
> chart.Correlation(data, histogram=TRUE, pch=19)
>
```



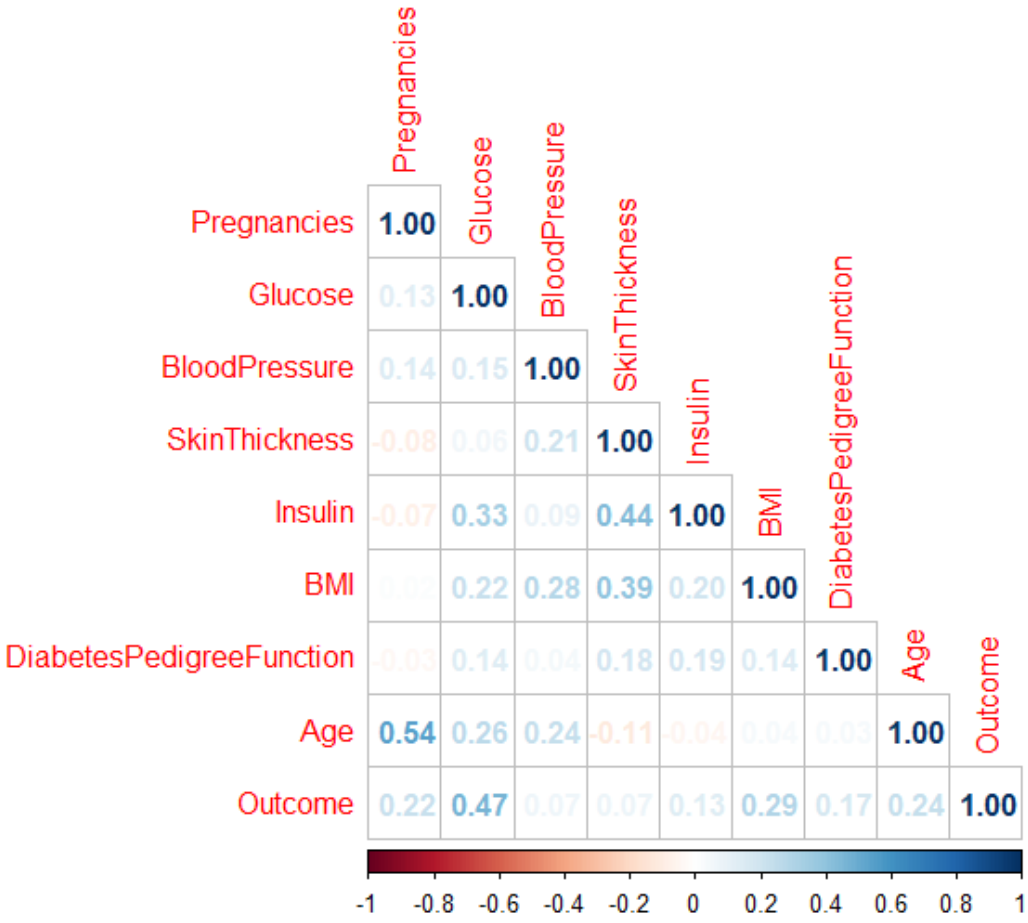
Using Heatmap getting some colors

```
> #Using heatmap
> # Get some colors
> col<- colorRampPalette(c("red", "orange", "blue"))(20)
> heatmap(x = corr, col = col, symm = TRUE)
>
```



Computing the matrix of correlations between the variables using corrplot

```
> #we compute the matrix of correlations between the variables using corrplot()
> corrplot(cor(data[, -10]), type = "lower", method = "number")
>
```


Pregnancies
Glucose
BloodPressure
SkinThickness
Insulin
BMI
DiabetesPedigreeFunction
Age
Outcome
Pregnancies 1.00
Glucose 0.13 1.00
BloodPressure 0.14 0.15 1.00
SkinThickness -0.08 0.06 0.21 1.00
Insulin -0.07 0.33 0.09 0.44 1.00
BMI 0.02 0.22 0.28 0.39 0.20 1.00
DiabetesPedigreeFunction -0.03 0.14 0.04 0.18 0.19 0.14 1.00
Age 0.54 0.26 0.24 -0.11 -0.04 0.04 0.03 1.00
Outcome 0.22 0.47 0.07 0.07 0.13 0.29 0.17 0.24 1.00
-1 -0.8 -0.6 -0.4 -0.2 0 0.2 0.4 0.6 0.8 1

ANOVA

EXCEL:

Anova: Single Factor

SUMMARY

Groups	*Count*	*Sum*	*Average*	*Variance*
Pregnancies	768	2953	3.845052083	11.35405632
Glucose	768	92847	120.8945313	1022.248314
BloodPressure	768	53073	69.10546875	374.6472712
SkinThickness	768	15772	20.53645833	254.4732453
Insulin	768	61286	79.79947917	13281.18008
BMI	768	24570.3	31.99257813	62.15998396
DiabetesPedigreeFun	768	362.401	0.471876302	0.109778638
Age	768	25529	33.24088542	138.3030459

ANOVA

Source of Variation	*SS*	*df*	*MS*	*F*	*P-value*	*F crit(tells significant level of confidence between all the coloumns and output)*
Between Groups	9319275.91	7	1331325.13	703.2664055	0	2.011076749
Within Groups	11615812.92	6136	1893.059472			
Total	20935088.83	6143				

Anova: Single Factor

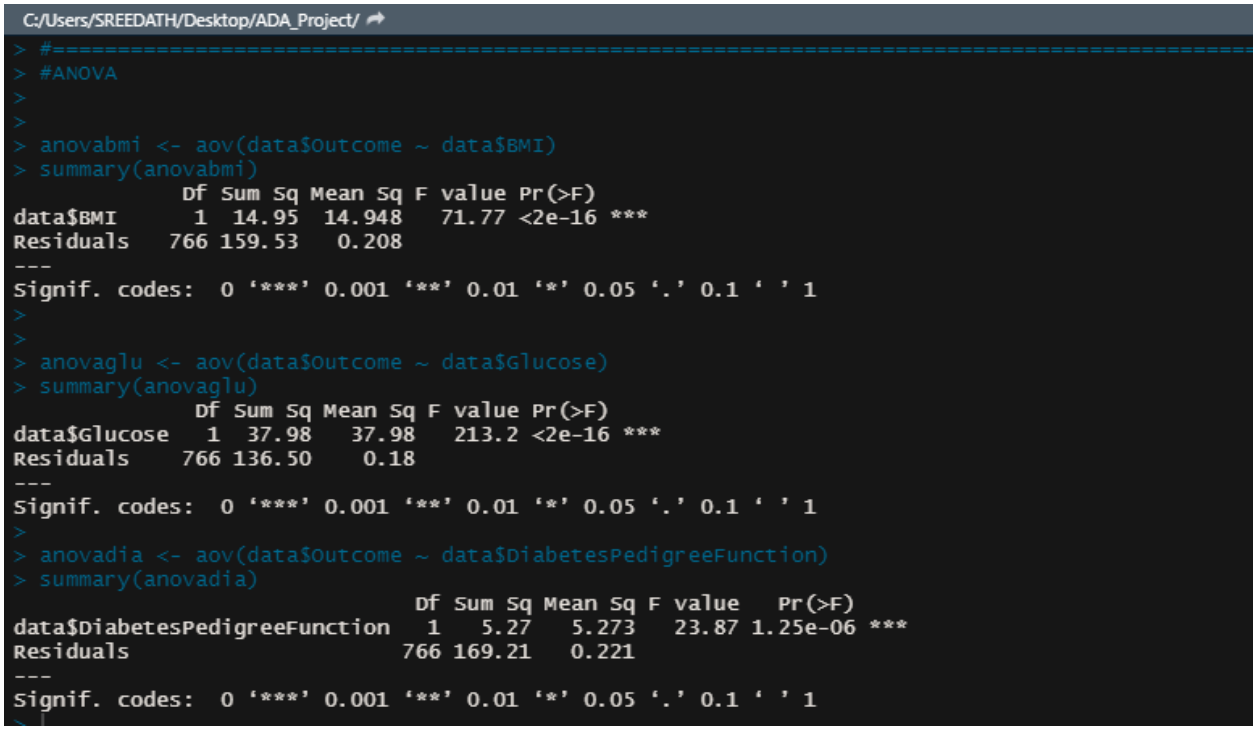
SUMMARY

Groups	*Count*	*Sum*	*Average*	*Variance*
Outcome	768	268	0.348958	0.227483
BMI2	768	24570.3	31.99258	62.15998

ANOVA

ce of Varia	*SS*	*df*	*MS*	*F*	*P-value*	*F crit*
Between G	384506.4	1	384506.4	12326.4	0	3.847528
Within Grc	47851.19	1534	31.19373			
Total	432357.6	1535				

R:



TIME SERIES ALGORITHMS

#5.Time series algorithm

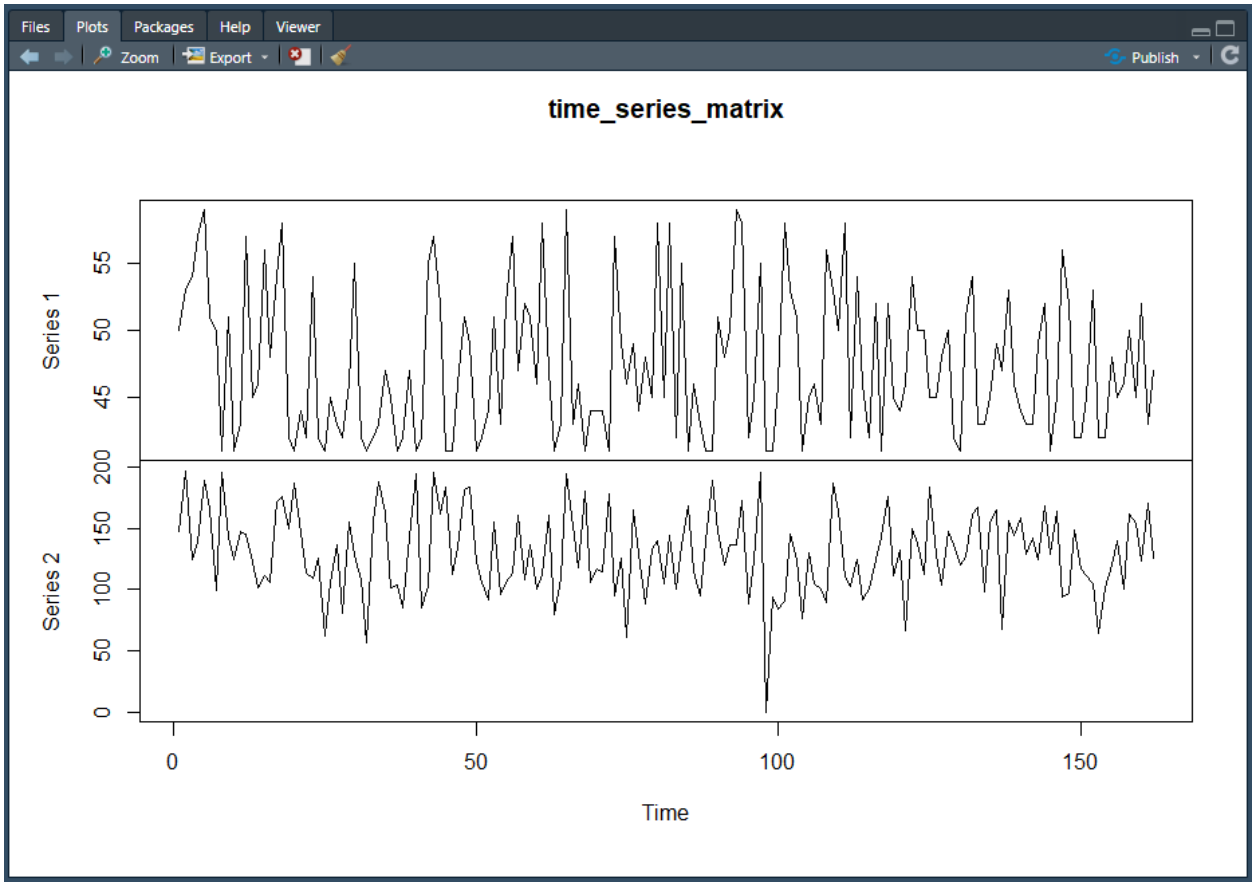
#Print time series analysis basis on Age (40-60) as per glucose value.

```
> age_40_60 = subset(data,Age>40&Age<60)
> time_series_age = ts(age_40_60$Age,start=1,end=162,frequency = 1)
> time_series_glucose = ts(age_40_60$Glucose,start=1,end=162,frequency = 1)
> age_glucose_matrix = matrix(c(time_series_age,time_series_glucose),nrow=162)
> time_series_matrix = ts(age_glucose_matrix,start=1,end=162,frequency = 1)
> plot(time_series_matrix)
>
```

Age between 40 and 60

Since the no of people between 40 and 60 is 162 we kept from 1(start) to 162(end)

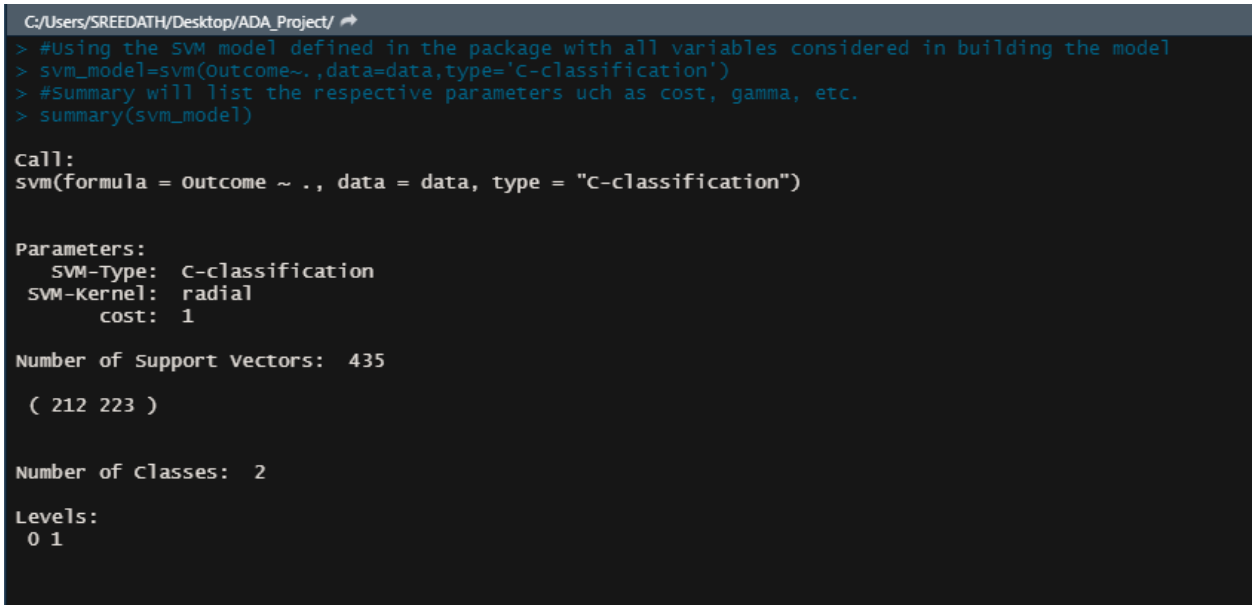
This helps us to predict with time series matrix



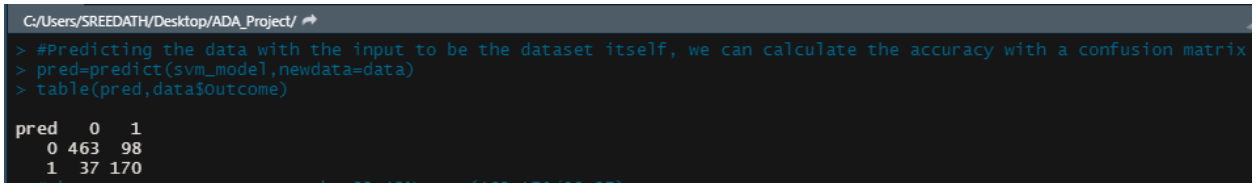

```
> time_series_age
Time Series:
Start = 1
End = 162
Frequency = 1
  [1] 50 53 54 57 59 51 50 41 51 41 43 57 45 46 56 48 54 58 42 41 44 42 54 42 41 45 43 42 46 55 42 41 42 43 47 45
 [37] 41 42 47 41 42 55 57 52 41 41 46 51 49 41 42 44 51 43 52 57 47 52 51 46 58 47 41 43 59 43 46 41 44 44 44 41
 [73] 57 49 46 49 44 48 45 58 45 58 42 55 41 46 43 41 41 51 48 50 59 58 42 45 55 41 41 46 58 53 51 41 45 46 43 56
[109] 53 50 58 42 54 46 42 52 41 52 45 44 46 54 50 50 45 45 48 50 42 41 51 54 43 43 45 49 47 53 46 44 43 43 49 52
[145] 41 45 56 52 42 42 45 53 42 42 48 45 46 50 45 52 43 47
> time_series_glucose
Time Series:
Start = 1
End = 162
Frequency = 1
  [1] 148 197 125 139 189 166  99 196 143 125 147 145 122 102 111 106 171 176 150 187 146 114 109 126  62 106 136
 [28]  81 155 128 108  57 156 188 163 102 104  85 143 194  85 103 196 162 184 112 134 181 184 122 106  92 155  96
 [55] 106 114 161 108 136 100 112 161  80 109 194 152 118 180 106 117 115 178  95 126  61 165 129  88 133 140 105
 [82] 144 100 137 168 115  95 140 189 148 120 136 137 173  89 125 195   0  94  84  91 145 125  76 130 105 100  90
[109] 187 164 110 103 125  92 100 124 143 176 111 132  67 150 138 112 183 128 104 147 136 120 127 162 167  98 154
[136] 165  68 156 144 158 129 142 125 168 129 164  94  97 149 117 111 105  65 102 120 140 100 162 154 123 170 126
```

SVM

#Using the SVM model defined in the package with all variables considered in building the model



#Predicting the data with the input to be the dataset itself, we can calculate the accuracy with a confusion matrix



```
C:/Users/SREEDATH/Desktop/ADA_Project/
> #The accuracy turns out to be 82.42%-----(463+170/98+37)
> #Now let's tune the SVM parameters to get a better accuracy on the training dataset
> svm_tune <- tune(svm, train.x=data, train.y=data$Outcome,
+                  kernel="radial", ranges=list(cost=10^(-1:2), gamma=c(.5,1,2)))
> print(svm_tune)

Parameter tuning of 'svm':

- sampling method: 10-fold cross validation

- best parameters:
 cost gamma
   10   0.5

- best performance: 0.03473523
```

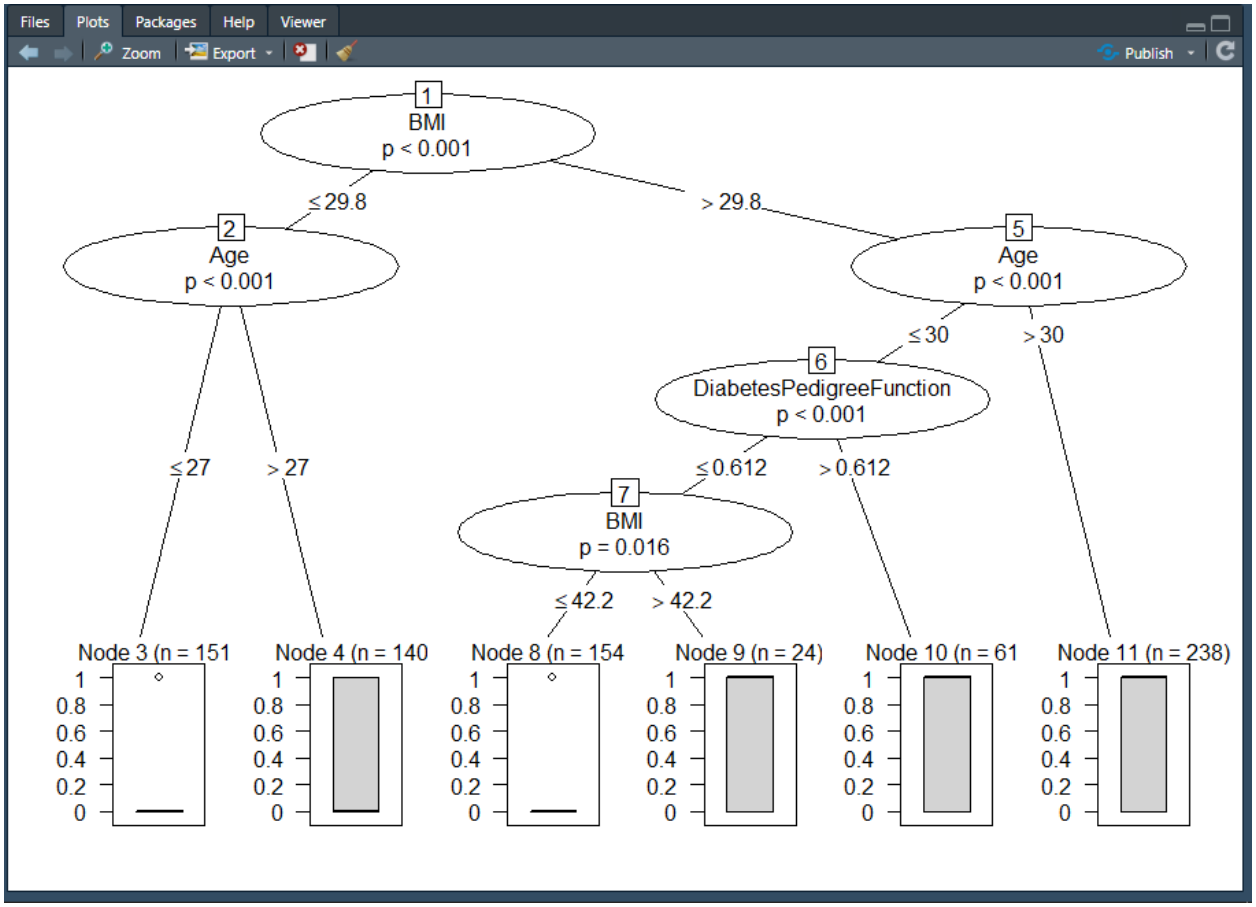
```
C:/Users/SREEDATH/Desktop/ADA_Project/
> #Gives an optimal cost to be 10 and a gamma value of 0.5
>
> svm_model_after_tune <- svm(Outcome ~ ., data=data, type='C-classification',kernel="radial", cost=10, gamma=0.5)
> summary(svm_model_after_tune)

Call:
svm(formula = Outcome ~ ., data = data, type = "C-classification", kernel = "radial", cost = 10,
    gamma = 0.5)


Parameters:
   SVM-Type:  C-classification
 SVM-Kernel:  radial
       cost:  10

Number of Support Vectors:  504

 ( 232 272 )


Number of Classes:  2

Levels:
 0 1
```

And finally the prediction is very accurate and we can see the pred and data(outcome) table where only 8(2+6)values are deviated.
760/768=98.95% accurate

```
C:/Users/SREEDATH/Desktop/ADA_Project/
> #The results show us that there is an improved accuracy of about 98%, results are obtained in the form of a confusion
 matrix
> pred <- predict(svm_model_after_tune,data)
> system.time(predict(svm_model_after_tune,data))
   user  system elapsed
   0.03    0.00    0.04
> table(pred,data$Outcome)

pred   0   1
   0 498   6
   1   2 262
>
```

CONDITIONAL INFERENCE TREE

```
C:/Users/SREEDATH/Desktop/ADA_Project/
> #Conditional Inference tree
> install.packages("partykit")
Error in install.packages : Updating loaded packages
> library("partykit")
> modelct = lm(Outcome~Pregnancies+Glucose+BloodPressure+SkinThickness+Insulin+BMI+DiabetesPedigreeFunction+Age,data=dat
a)
> x = c(coef(modelct)[2],coef(modelct)[3],coef(modelct)[4],coef(modelct)[5],coef(modelct)[6],coef(modelct)[7],coef(model
ct)[8],coef(modelct)[9])
> max_3 = sort(x,decreasing = T)
> max_3 = max_3[1:3]
> modelct = lm(Outcome~BMI+DiabetesPedigreeFunction+Age,data=data) #check for the values in max using above given charac
terstics of modelct
> tree = ctree(Outcome~BMI+DiabetesPedigreeFunction+Age,data)
> plot(tree)
```



LOGISTIC REGRESSION

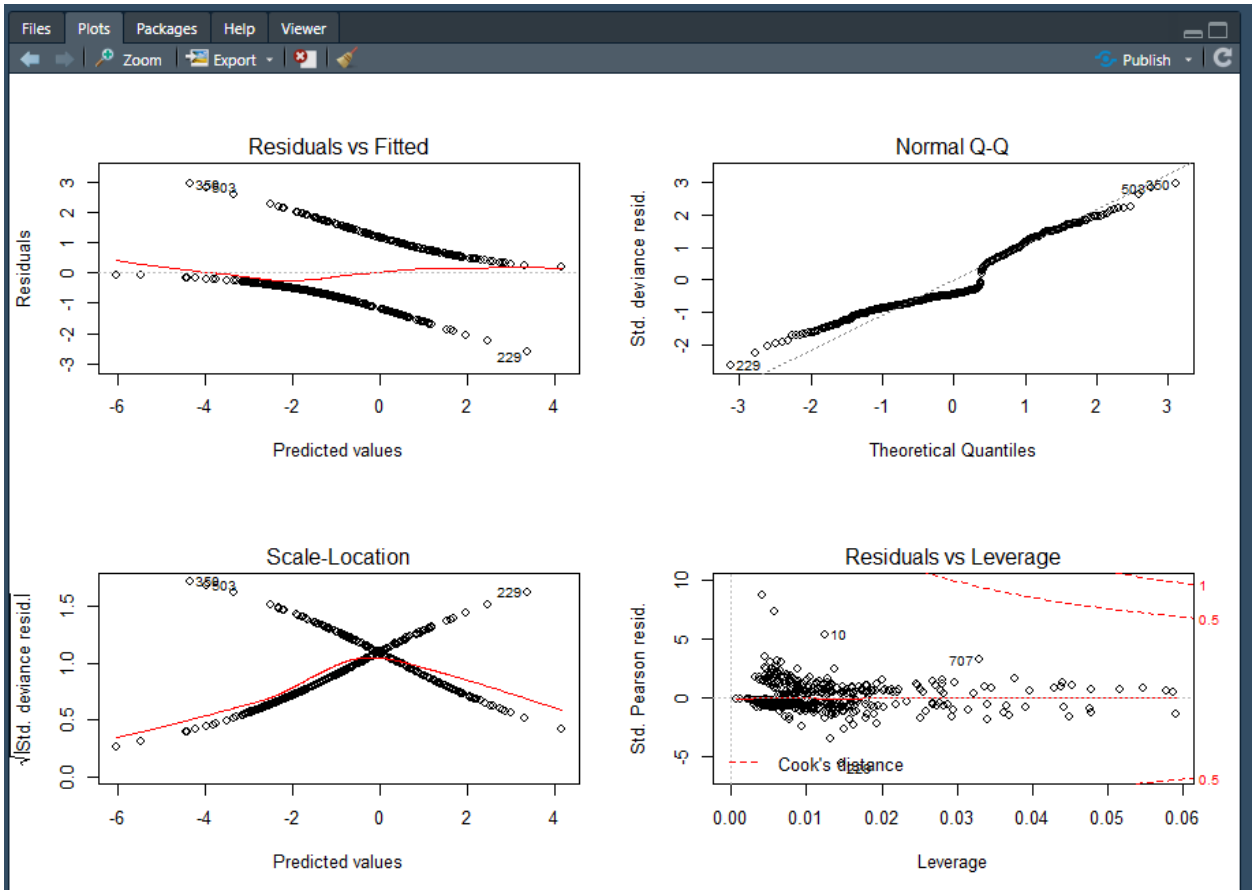
```
C:/Users/SREEDATH/Desktop/ADA_Project/
> #=============================================================================================
> #Logistic Regeression
>
>
> set.seed(123)
> n <- nrow(data)
> train <- sample(n, trunc(0.70*n))
> data_training <- data[train, ]
> data_testing <- data[-train, ]
> # Training The Model
> glm_fm1 <- glm(Outcome ~., data = data_training, family = binomial)
> summary(glm_fm1)

Call:
glm(formula = Outcome ~ ., family = binomial, data = data_training)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.2424  -0.7256  -0.4283   0.7341   2.9311

Coefficients:
                          Estimate Std. Error z value Pr(>|z|)
(Intercept)             -8.405409   0.841872  -9.984  < 2e-16 ***
Pregnancies              0.103471   0.037973   2.725  0.00643 **
Glucose                  0.035730   0.004563   7.830 4.89e-15 ***
BloodPressure           -0.012707   0.006057  -2.098  0.03590 *
SkinThickness            0.003563   0.008088   0.440  0.65959
Insulin                 -0.001710   0.001060  -1.613  0.10671
BMI                      0.088735   0.017954   4.942 7.72e-07 ***
DiabetesPedigreeFunction 0.696250   0.334761   2.080  0.03754 *
Age                      0.017015   0.011066   1.538  0.12415
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 694.17  on 536  degrees of freedom
Residual deviance: 509.76  on 528  degrees of freedom
AIC: 527.76

Number of Fisher Scoring iterations: 5
```

Here we can clearly see the skin thickness,insulin and age are not significant that is >0.1 so remove them and model again.

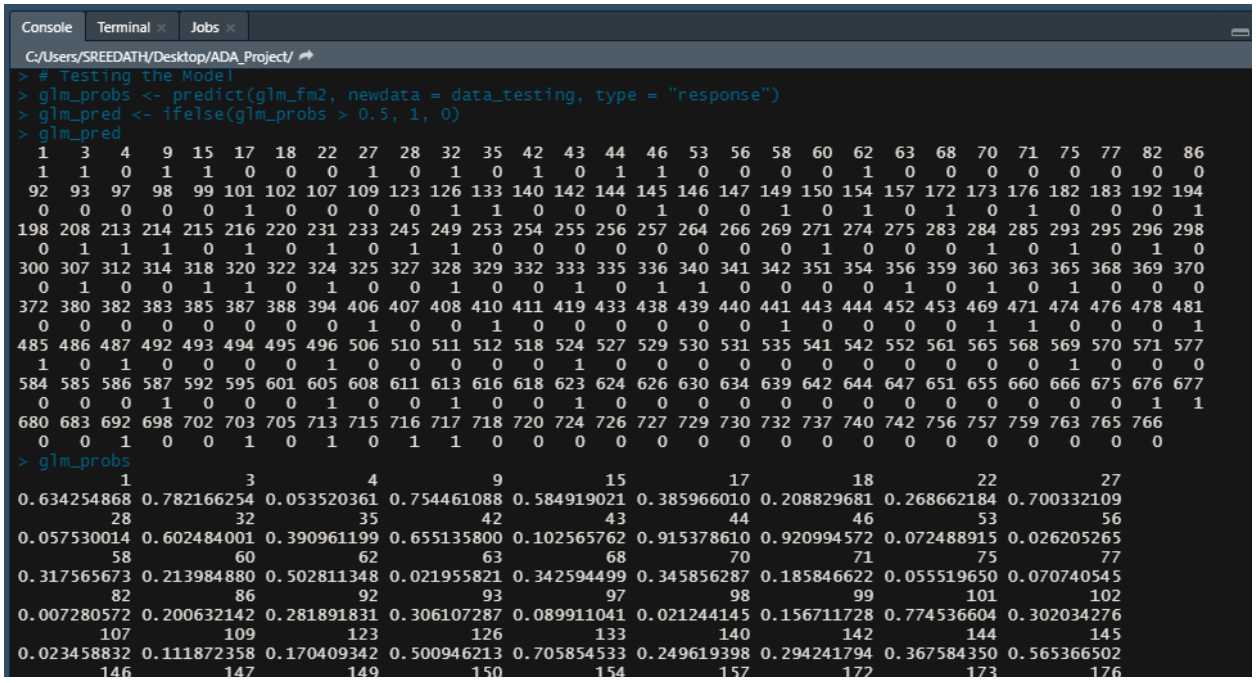
```
C:/Users/SREEDATH/Desktop/ADA_Project/
> glm_fm2 <- update(glm_fm1, ~. - SkinThickness - Insulin - Age )
> summary(glm_fm2)

Call:
glm(formula = Outcome ~ Pregnancies + Glucose + BloodPressure +
    BMI + DiabetesPedigreeFunction, family = binomial, data = data_training)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.6175  -0.7389  -0.4472   0.7157   2.9445

Coefficients:
                          Estimate Std. Error z value Pr(>|z|)
(Intercept)              -7.799784   0.775631 -10.056  < 2e-16 ***
Pregnancies               0.138138   0.032715   4.223 2.42e-05 ***
Glucose                   0.034314   0.004101   8.367  < 2e-16 ***
BloodPressure            -0.011448   0.005844  -1.959   0.0501 .
BMI                       0.084610   0.016826   5.028 4.94e-07 ***
DiabetesPedigreeFunction  0.676771   0.330840   2.046   0.0408 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 694.17  on 536  degrees of freedom
Residual deviance: 515.01  on 531  degrees of freedom
AIC: 527.01

Number of Fisher Scoring iterations: 5
```

Plotting the model2 without skin thickness,insulin and age.

```
> par(mfrow = c(2,2))
> plot(glm_fm2)
```



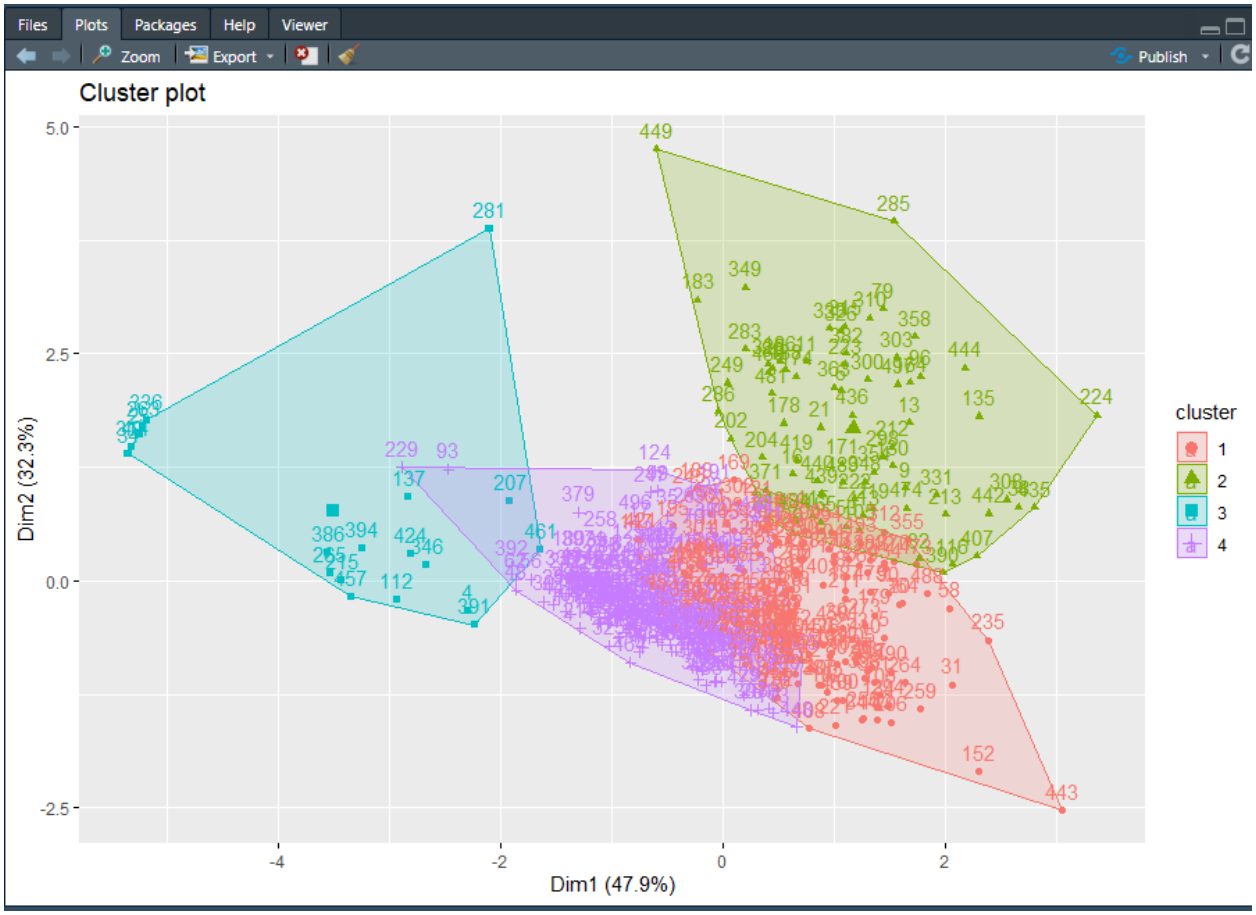
Pred vectors :



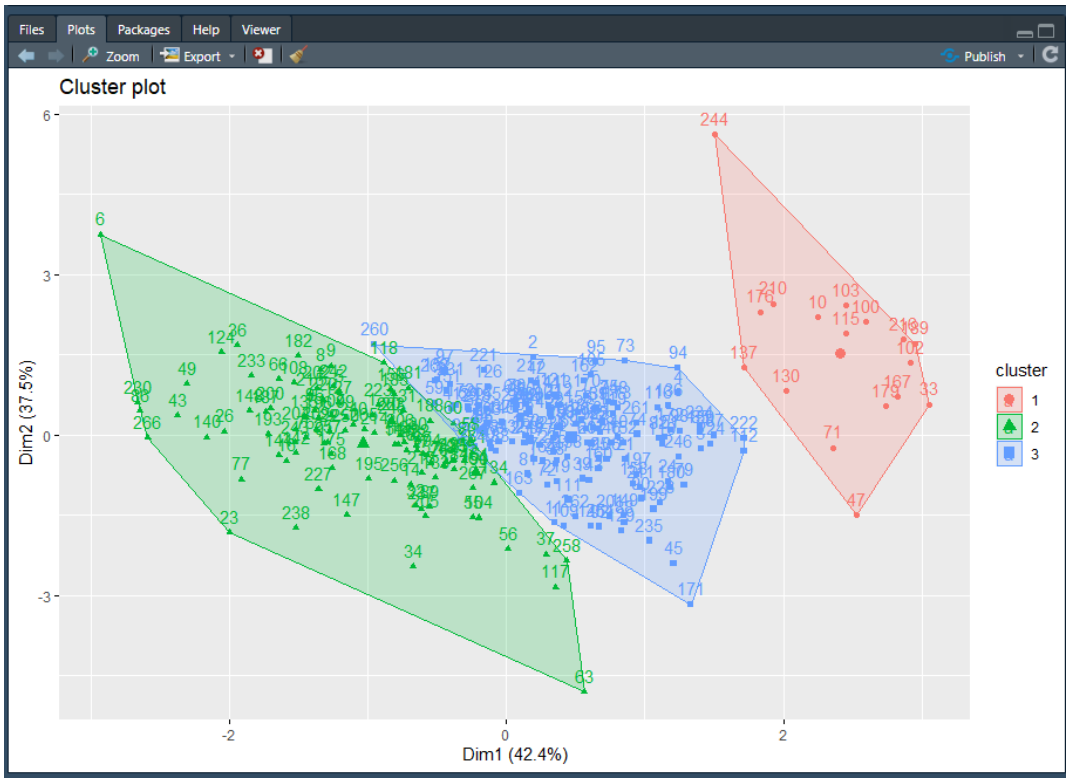
```r
confusionMatrix(glm_pred, data_testing$Outcome )
# Confusion Matrix for logistic regression

acc_glm_fit <- confusionMatrix(glm_pred, data_testing$Outcome )$overall['Accuracy']
```

K -MEANS Clustering:

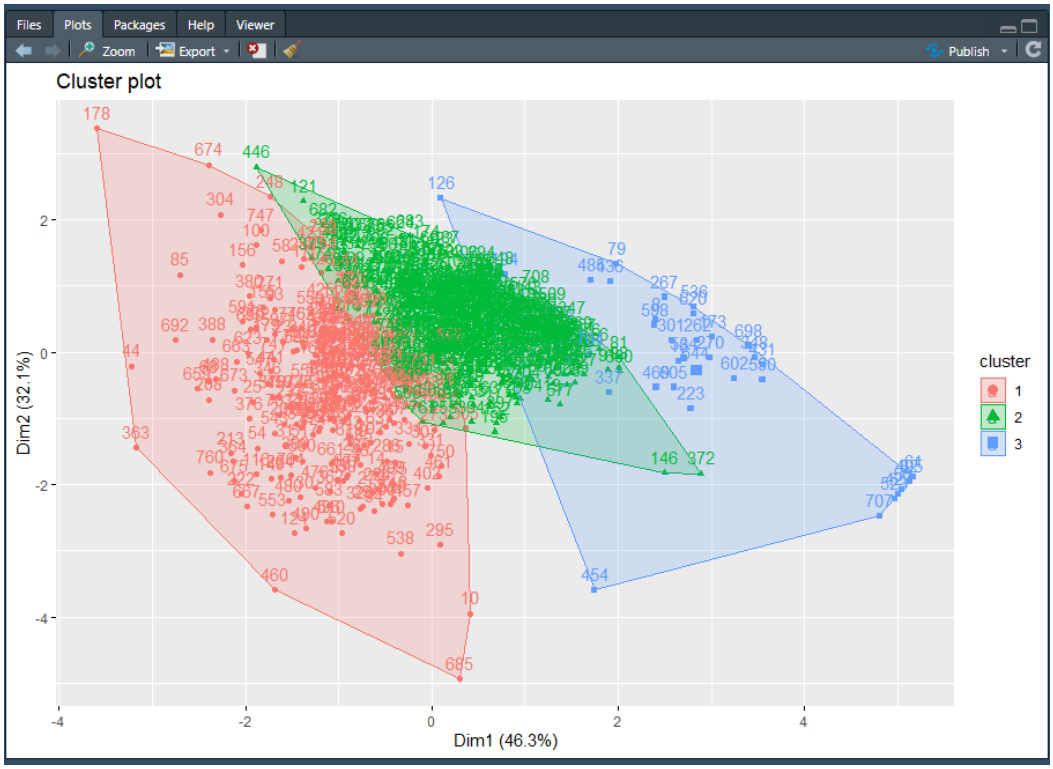
Clustering of data when outcome is 0:



Clustering of the data when Outcome is 1:



Clustering when data is not divided

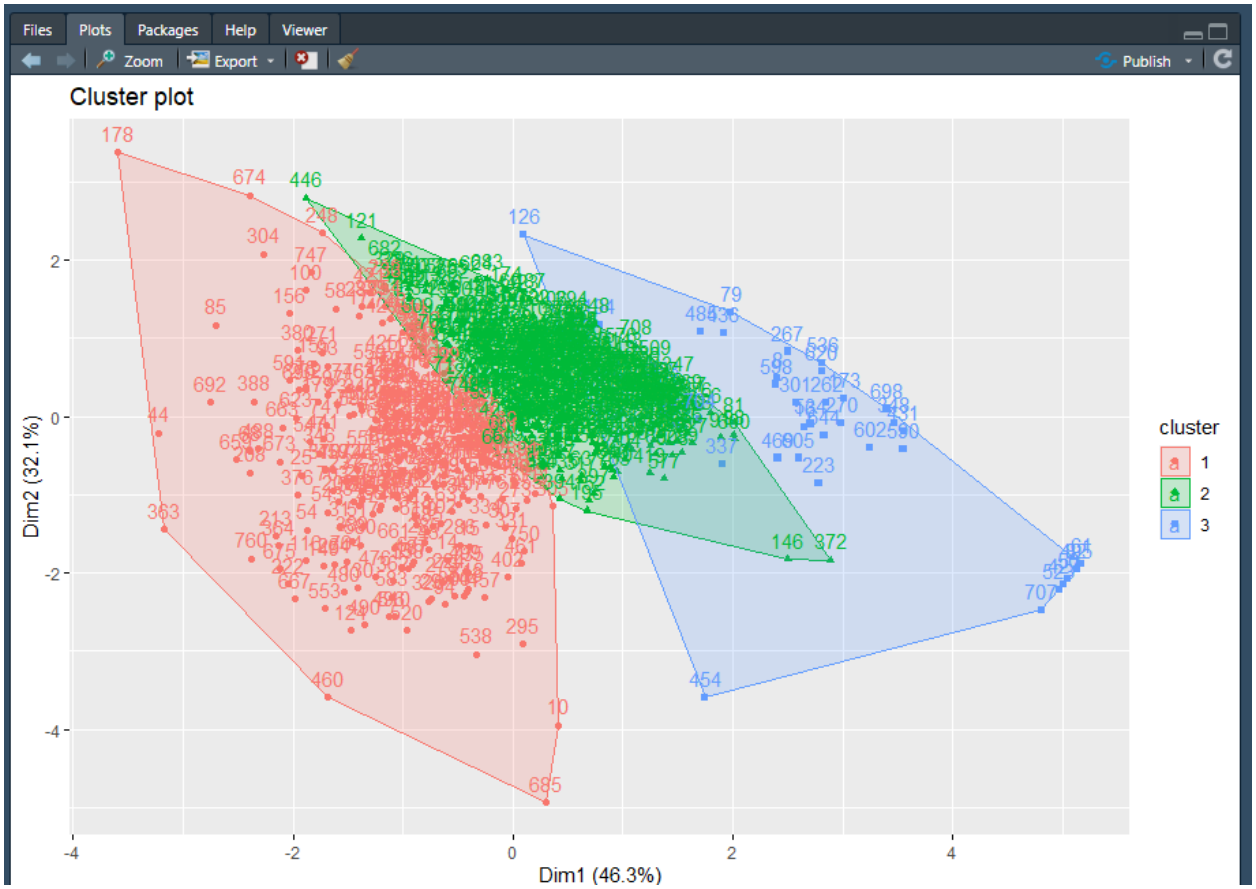


we can clearly see there is good quality clustering possible when the outcome is 1
that means studying these groups act more similar to the medicines and side effects,etc. when we dont divide the data

```
C:/Users/SREEDATH/Desktop/ADA_Project/
> #Clustering
> #k-means clustering
> #outcome is 0
>
> dfcl1 <- data%>%filter(Outcome == '0')
>   dfclusel1 <- subset(dfcl1,select = -c(Pregnancies,SkinThickness,Outcome,Insulin,DiabetesPedigreeFunction,Glucose))
> kmclu1 <- kmeans(dfclusel1, 4, nstart = 25)
> print(kmclu1)
K-means clustering with 4 clusters of sizes 187, 73, 20, 220

Cluster means:
  BloodPressure      BMI      Age
1      79.55615 33.35829 29.32620
2      78.30137 30.45205 54.23288
3       1.20000 19.12000 29.60000
4      61.25000 28.67591 25.27273

Clustering vector:
  1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29
  4   4   1   3   1   2   4   1   2   4   2   1   2   4   1   2   4   1   4   1   2   4   4   4   3   1   4   4   2
 30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58
  4   1   2   4   3   4   4   1   2   4   1   4   1   1   4   1   1   4   4   3   1   4   1   2   4   4   4   1   1
 59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87
  1   1   4   4   4   4   1   4   4   4   1   4   4   1   4   4   4   1   4   1   2   1   4   1   4   4   4   4   1
 88  89  90  91  92  93  94  95  96  97  98  99 100 101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116
  1   2   1   4   4   4   1   4   2   4   1   4   1   4   4   1   1   2   1   4   4   4   4   1   3   4   4   2   2
117 118 119 120 121 122 123 124 125 126 127 128 129 130 131 132 133 134 135 136 137 138 139 140 141 142 143 144 145
  1   4   1   4   1   4   1   4   4   1   1   4   4   2   4   4   4   1   2   4   3   2   4   1   1   1   1   1   4
146 147 148 149 150 151 152 153 154 155 156 157 158 159 160 161 162 163 164 165 166 167 168 169 170 171 172 173 174
  4   1   4   4   1   1   1   1   1   4   1   1   4   4   4   4   4   4   2   2   4   4   4   1   1   2   1   4   2
175 176 177 178 179 180 181 182 183 184 185 186 187 188 189 190 191 192 193 194 195 196 197 198 199 200 201 202 203
  4   1   1   2   1   4   1   1   2   4   1   2   1   1   1   4   4   1   4   4   1   4   4   1   4   1   1   2   4
204 205 206 207 208 209 210 211 212 213 214 215 216 217 218 219 220 221 222 223 224 225 226 227 228 229 230 231 232
  2   4   1   3   4   1   4   1   2   2   4   3   4   1   1   2   4   1   2   2   2   1   4   4   1   4   4   4   4
233 234 235 236 237 238 239 240 241 242 243 244 245 246 247 248 249 250 251 252 253 254 255 256 257 258 259 260 261
  1   4   1   1   4   4   4   4   4   4   2   4   1   4   4   4   2   1   4   4   1   1   1   1   4   4   1   4   4
262 263 264 265 266 267 268 269 270 271 272 273 274 275 276 277 278 279 280 281 282 283 284 285 286 287 288 289 290
  4   3   1   3   1   1   1   4   1   1   4   1   4   4   1   1   1   4   4   3   4   2   4   2   2   4   1   1   1
291 292 293 294 295 296 297 298 299 300 301 302 303 304 305 306 307 308 309 310 311 312 313 314 315 316 317 318 319
  4   4   4   1   1   4   4   2   4   2   1   1   2   1   4   1   4   2   1   2   4   1   4   3   2   4   1   1   1
320 321 322 323 324 325 326 327 328 329 330 331 332 333 334 335 336 337 338 339 340 341 342 343 344 345 346 347 348
  1   4   1   1   4   4   2   4   2   4   4   2   4   2   4   1   3   4   4   4   1   4   4   4   1   4   3   4   2
349 350 351 352 353 354 355 356 357 358 359 360 361 362 363 364 365 366 367 368 369 370 371 372 373 374 375 376 377
  2   1   1   1   4   2   1   4   4   2   4   4   1   1   2   1   1   4   4   1   4   1   2   1   1   1   4   4   1
378 379 380 381 382 383 384 385 386 387 388 389 390 391 392 393 394 395 396 397 398 399 400 401 402 403 404 405 406
  4   4   4   4   2   1   4   4   3   1   4   1   2   3   4   1   3   1   4   4   1   4   4   1   1   1   4   1   1
407 408 409 410 411 412 413 414 415 416 417 418 419 420 421 422 423 424 425 426 427 428 429 430 431 432 433 434 435
  2   1   4   1   4   1   2   4   1   4   1   4   2   1   1   1   4   3   1   1   4   4   4   1   4   4   4   1   2
436 437 438 439 440 441 442 443 444 445 446 447 448 449 450 451 452 453 454 455 456 457 458 459 460 461 462 463 464
  2   1   4   2   2   4   2   1   2   4   4   4   4   2   1   4   4   1   1   1   4   3   1   1   1   3   1   1   4
465 466 467 468 469 470 471 472 473 474 475 476 477 478 479 480 481 482 483 484 485 486 487 488 489 490 491 492 493
```

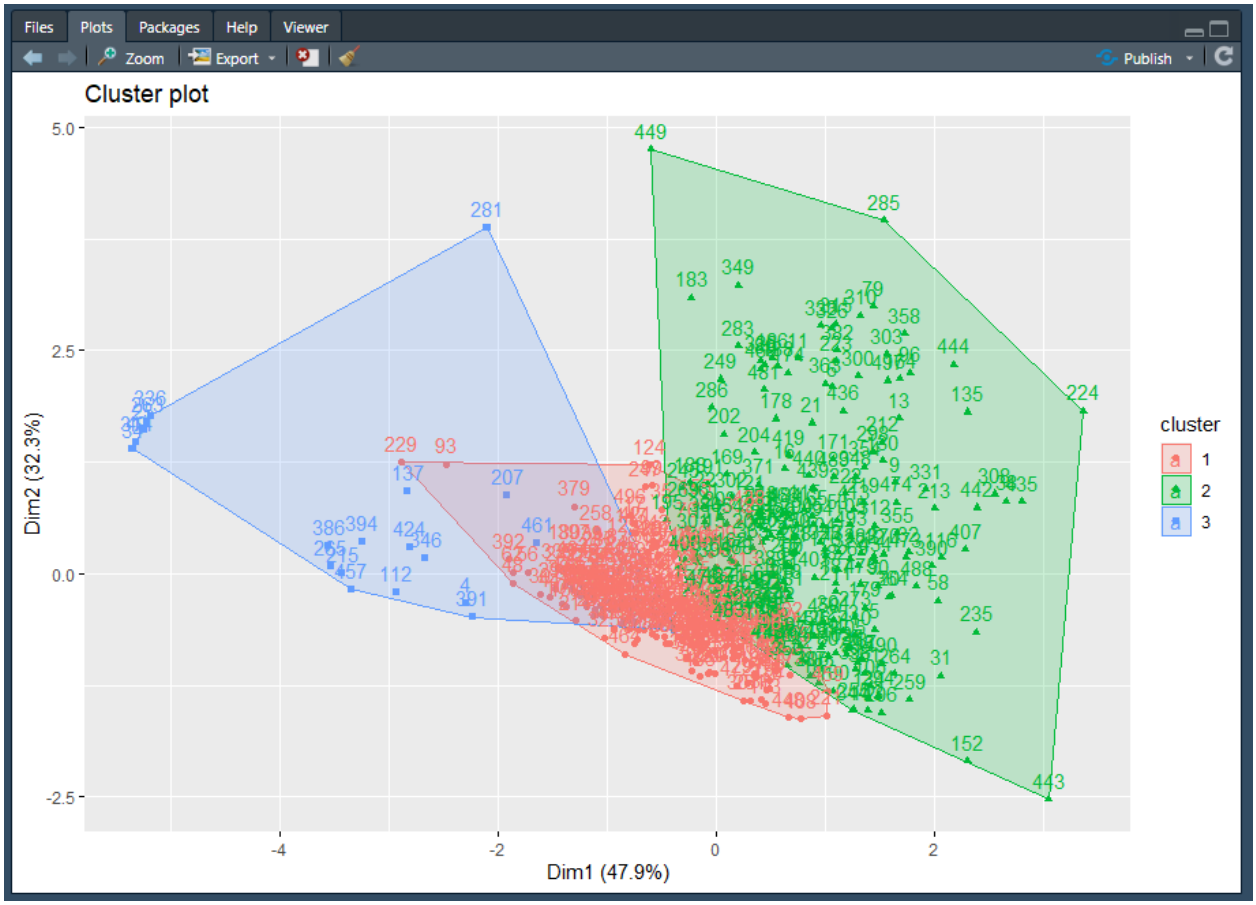
```
C:/Users/SREEDATH/Desktop/ADA_Project/
> #outcome is 1
>
> dfcl2 <- data%>%filter(Outcome == '1')
> dfclusel2 <- subset(dfcl2,select = -c(Pregnancies,SkinThickness,Outcome,Insulin,DiabetesPedigreeFunction,Glucose))
>
>
> kmclu2 <- kmeans(dfclusel2, 3, nstart = 25)
> print(kmclu2)
K-means clustering with 3 clusters of sizes 17, 106, 145

Cluster means:
  BloodPressure      BMI      Age
1      1.764706 35.30000 31.17647
2     83.660377 35.43396 46.83962
3     69.537931 34.91103 30.61379

Clustering vector:
  1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29
  2   3   3   3   2   2   3   2   2   1   3   3   3   2   3   2   3   2   3   2   3   2   2   3   3   2   3   3   3
 30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58
  2   3   2   1   2   3   2   2   3   3   3   3   3   2   3   3   3   1   2   2   3   3   3   3   2   2   2   2   2
 59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87
  3   2   3   3   2   2   3   2   2   3   3   3   1   3   3   3   3   2   2   2   3   2   3   3   3   3   3   2   3
 88  89  90  91  92  93  94  95  96  97  98  99 100 101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116
  3   2   3   2   3   3   3   3   2   3   3   2   1   3   1   1   2   3   3   2   2   3   3   3   3   3   2   1   3
117 118 119 120 121 122 123 124 125 126 127 128 129 130 131 132 133 134 135 136 137 138 139 140 141 142 143 144 145
  2   2   3   3   3   2   3   2   3   3   2   3   3   1   2   3   2   2   2   3   1   3   3   2   3   2   3   2   3
146 147 148 149 150 151 152 153 154 155 156 157 158 159 160 161 162 163 164 165 166 167 168 169 170 171 172 173 174
  3   2   2   3   3   3   3   3   2   3   2   3   3   3   3   2   3   3   3   3   2   1   2   3   3   3   3   3   3
175 176 177 178 179 180 181 182 183 184 185 186 187 188 189 190 191 192 193 194 195 196 197 198 199 200 201 202 203
  2   1   3   3   1   3   2   2   3   2   2   3   2   2   1   2   2   3   2   2   2   2   3   3   3   2   3   2   3
204 205 206 207 208 209 210 211 212 213 214 215 216 217 218 219 220 221 222 223 224 225 226 227 228 229 230 231 232
  2   2   2   3   3   2   1   3   3   2   2   2   1   3   3   3   2   3   3   2   3   3   3   2   2   3   2   3   3
233 234 235 236 237 238 239 240 241 242 243 244 245 246 247 248 249 250 251 252 253 254 255 256 257 258 259 260 261
  2   3   3   3   2   2   3   2   3   2   2   1   2   3   3   3   3   2   3   3   3   3   2   2   2   2   3   3   3
262 263 264 265 266 267 268
  3   2   2   2   2   2   3

Within cluster sum of squares by cluster:
[1]  3918.789 25544.465 24517.976
 (between_SS / total_SS =  68.2 %)

Available components:

[1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss" "betweenss"    "size"
[8] "iter"         "ifault"
>
> fviz_cluster(kmclu2, data = dfclusel2)
```

K- MEDOIDS Clustering:
When data is not divided on the basis of Outcome



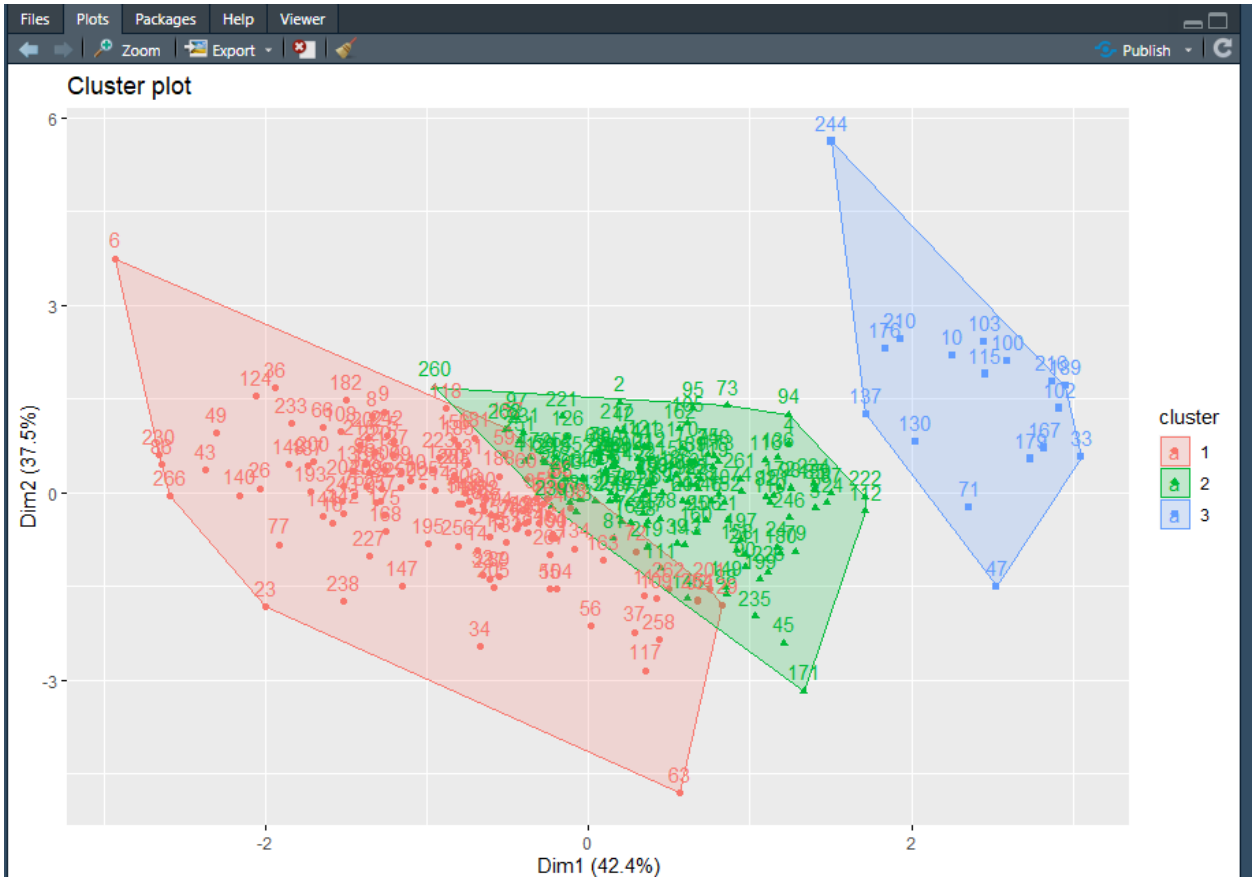

```
C:/Users/SREEDATH/Desktop/ADA_Project/
> #medoids clustering
> library(cluster)
> library(factoextra)
>
> #for normal data
> dfmed <- subset(data,select = -c(Pregnancies,SkinThickness,Outcome,Insulin,DiabetesPedigreeFunction,Glucose))
> kmed <- pam(x=dfmed,k=3)
> kmed
Medoids:
     ID BloodPressure  BMI Age
593 593            80 34.4  44
309 309            68 30.5  25
534 534             0 29.8  31
Clustering vector:
  1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29
  1   2   2   2   2   2   2   3   1   1   1   1   1   1   1   3   1   2   3   2   1   1   1   2   1   1   1   2   1
 30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58
  1   1   2   2   1   1   2   1   1   2   1   2   1   1   1   2   2   2   2   2   3   2   2   2   1   1   2   1   1
 59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87
```

Clustering of data when outcome is 0:




```
#for outcome0
dfmed1 <- subset(dfcl1,select = -c(Pregnancies,SkinThickness,Outcome,Insulin,DiabetesPedigreeFunction,Glucose))
kmed1 <- pam(x=dfmed1,k=3)
kmed1
kmed1$clustering
kmed1$medoids

summary(kmed1)
fviz_cluster(kmed1)
```

Clustering of data when outcome is 1:



We can see that k means clustering has done better job when Outcome is 1 than k medoids clustering

```
#for outcome1
dfmed2 <- subset(dfcl2,select = -c(Pregnancies,SkinThickness,Outcome,Insulin,DiabetesPedigreeFunction,Glucose))
kmed2 <- pam(x=dfmed2,k=3)
kmed2
kmed2$clustering
kmed2$medoids

summary(kmed2)
fviz_cluster(kmed2)
```