## Predicting the probability of the malignant breast cancer

#Read the data from the working directory, create your own working directly to read the dataset.

data1 <- read.csv ("C:/Users/Deep/Desktop/data/

synthetic\_cancer\_data.csv",header=TRUE,sep=",")

data2<-data.frame(data1)

# remove Sample\_No from data 2

data2 <- data2[,-1]

#perform exploratory data analysis to know about the data

# display top 6 rows of dataset to see how data look like

head (data2)

# display bottom 6 rows to see how data look like

tail(data2)

# describe the structure of data , it displays the datatype of each variable present in the data like whether that particular varibale is numeric , factor etc .

str(data2)

#display the column name of the data

names(data2)

# display the datatype

class(data2)

# Check the missing values present in the data

is.na(data2)

#to check the percentage of benign and malignant breast cancer in the data

table(data2$Outcome)/nrow(data2)

### Model Building & Interpretation on Full Data

#install randomForest package

install.packages("randomForest")

library(randomForest)

#Building Random Forests model on full data

**Program1:**

rf\_model <- randomForest(Outcome ~ ., data=data2, ntree=1500,mtry=3,importance=TRUE)

rf\_model

## Splitting Data set into Training and Testing

#Set seed in order to reproduce the sample

set.seed(2)

#splitting data set into training and testing dataset in 70:30

install.packages("caTools")

library(caTools)

sample <- sample.split(data2$Outcome,SplitRatio=0.70)

#No of observations in train dataset

train\_data <- subset(data2,sample==TRUE)

# No of observations in test dataset

test\_data <- subset(data2,sample==FALSE)

## Model Building & Interpretation on Training and Testing Data

#Building Random Forests model using training data

#install randomForest package

install.packages("randomForest")

library(randomForest)

**Program1.1:**

r\_model <- randomForest(Outcome ~ ., data=train\_data,ntree=1500,mtry=3, importance=TRUE)

print(r\_model)

plot(r\_model)

plot(margin(r\_model,test\_data$Outcome))

# Predicting the model using test data

ran\_pred <- predict(r\_model,test\_data)

#Display the confusion matrix or classification table

table(test\_data$Outcome ,ran\_pred)

# Predicting the probability matrix using test data

ran\_prob <- predict(r\_model,test\_data,type = "prob" )

ran\_prob

ran\_prob1<-data.frame(ran\_prob)