International Research Journal of Computer Science (IRJCS)
Issue 08, Volume 4 (August 2017)
ISSN: 2393-9842
www.irjcs.com

AN EVALUATION OF SUPERVISED MACHINE LEARNING ALGORITHMS FOR HEART DISEASE DIAGNOSIS

Akanksha Gupta
B.Tech/CSE, VIT University, India
akanksha.1997@gmail.com

Shubham Pathak
B.Tech/CSE, VIT University, India
shubham.pathak2014@vit.ac.in

Prof. Kannadasan R
SCOPE, Professor/CSE, VIT University, India
kannadasan.r@vit.ac.in

Manuscript History
Number: **IRJCS/RS/Vol.04/Issue08/AUCS10087**
DOI: 10.26562/IRJCS.2017.AUCS10087
Received: 09, August 2017
Final Correction: 14, August 2017
Final Accepted: 19, August 2017
Published: August 2017
Citation: Akanksha, G., Shubham, P. & Prof., K. (2017). AN EVALUATION OF SUPERVISED MACHINE LEARNING ALGORITHMS FOR HEART DISEASE DIAGNOSIS. IRJCS:: International Research Journal of Computer Science, IV, 33-42. doi: 10.26562/IRJCS.2017.AUCS10087
Editor: Dr.A.Arul L.S, Chief Editor, IRJCS, AM Publications, India

Abstract— In this paper, we have taken four supervised machine learning algorithms and compared their competency in terms of the accuracy achieved by them in predicting the occurrence of heart disease. We compare and contrast the state of art systems used for Heart disease diagnosis that use machine learning. We have taken a set of 14 different attributes to predict heart disease, placing regard to the presence of reversible defect in thalassemia and asymptomatic chest pain being present or not. We run the chosen algorithms on our datasets and get the collective results. We analyze these results and conclude on which the best approach in maximum cases is for accuracy is disease diagnosis.

Keywords— Machine Learning, Artificial Intelligence, Logistic Regression, Random forests, SVM

I. INTRODUCTION

In the past years a lot of diseases that might have been deemed untreatable have been cured because of early detection of the problem. Machine learning plays a strong role in this process. Machine learning comes under the field of Artificial Intelligence. [2] Machine learning enables an approach towards building sophisticated, automatic, and objective algorithms for analysis of high-dimensional and multimodal biomedical data. [1]Machine learning explores the study and construction of algorithms that can learn from and make predictions on data – such algorithms overcome following strictly static program instructions by making data-driven predictions, by building model from sample inputs (either supervised or unsupervised). Machine learning is employed tasks related to computing where designing and programming explicit algorithms with good performance is infeasible or very exhausting. We generally apply the machine learning models in cases of lack of predefined models.

In section 2 we talk about the work done related to this field and results shown by that work. Section 3 gives a brief overview of the machine learning algorithms. The methodology has been discussed in section 4. In this we have implemented machine learning approach on UCI heart disease data set and compared the results of logistic regression, random forest, boosted trees and support vector machines. In this first we imported the data then we checked for if any data is missing or not. In case we have missing data then we just remove them from our data sets. Then we created a function called convert. Magic which is used for converting the class of predictor values for the variables in our data set. Our data set consists of 14 different variables. Next, we split our data into training data set and testing data set, this is done by with the help of r library caret. After we have these two sets we can now implement above described machine learning approaches and compare their results. Section 5 comprises of the results and inferences. Based on our implementation in previous section we give out the results for the various machine learning models and describe how good was the model in predicting the heart disease with respect to other models. Then we have given summary about the best model. Also in this section, we see the importance of variables in boosted tree and based on that we make some conclusions and also describe the plot showing the importance of these variables. In section 6 we have provided the conclusion. Based on the results obtained we give an overall conclusion of our study, and also analyze the important points of the result and the facts which could have improved the accuracy and minimized the errors, thus giving us better prediction results. We gain an understanding of the proposed models can be improved.

II. RELATED WORK

In a medical diagnosis problem, what is needed is a set of examples or attributes that are representative of all the variations of the disease. The examples need to be selected very carefully if the system is to perform reliably and efficiently. In this paper, India centric dataset is used for Heart disease diagnosis. The correct diagnosis performance of the automatic diagnosis system is estimated by using classification accuracy, sensitivity and specificity analysis. The study shows that, the SVM with Sequential Minimization Optimization learning algorithm have better choice for medical disease diagnosis application. [13]Heart disease is the leading cause of death in the world over the past 10 years. Researchers have been using several data mining techniques in the diagnosis of heart disease. Diabetes is a chronic disease that occurs when the pancreas does not produce enough insulin, or when the body cannot effectively use the insulin it produces. Most of these systems have successfully employed Machine learning methods such as Naïve Bayes and Support Vector Machines for the classification purpose. Support vector machines are a modern technique in the field of machine learning and have been successfully used in different fields of application. Using diabetics' diagnosis, the system exhibited good accuracy and predicts attributes such as age, sex, blood pressure and blood sugar and the chances of a diabetic patient getting a heart disease. [14]

This survey paper provides the comparative analysis of different machine learning algorithms for diagnosis of different diseases such as heart disease, diabetes disease, liver disease, dengue disease and hepatitis disease. It brings attention towards the suite of machine learning algorithms and tools that are used for the analysis of diseases and decision-making process accordingly. Survey highlights the advantages and disadvantages of these algorithms. Improvement graphs of machine learning algorithms for prediction of diseases are presented in detail. From analysis, it can be clearly observed that these algorithms provide enhanced accuracy on different diseases. This survey paper also provides a suite of tools that are developed in community of AI. These tools are very useful for the analysis of such problems and also provide opportunity for the improved decision-making process. [15]

The accurate diagnosis of heart diseases is one of the most important biomedical problems whose administration is imperative. In the proposed work, decision support system is made by three data mining techniques namely Classical Random Forest, Modified Random Forest and Weighted Random Forest. The classical random forests construct a collection of trees. In Modified Random Forest, the tree is constructed dynamically with online fitting procedure. The system extracts hidden Knowledge from a historical heart disease database; models are trained and validated against a test dataset. Classification matrix methods are used to evaluate the effectiveness of the models. Modified Random Forest and Weighted Random Forest both the two methods could answer complex queries, each with its own strength with respect to ease of model interpretation, access to detailed information and accuracy. When 14 attributes of UCI data set is used for Classical Random Forest then Accuracy gain is 74.19% and for Modified Random Forest it is 79.42% similarly In Weighed Random Forest it is 83.6%. When 14 attributes of UCI data is used then precision for Classical Random Forest is 73.15%, Modified Random Forest is 80.46%and Weighed Random Forest is 82.22%. Similarly Recall for Classical Random Forest is 75.24%, Modified Random Forest is 78.37%and Weighed Random Forest is 82.55%. Modified Random Forest and Weighed Random Forest both methods Results are easier to read and interpret. [16]

III. LITERATURE REVIEW

A. Machine learning: Concept and application

Machine learning developed under artificial intelligence while uncovering the vast domains of AI. In the 1950s, excursions were made to approach the problem of gaining knowledge by machine using several symbolic methods, and later, methods based on the connection principle, such as neural networks and perceptron's, were deeply studied. Consequently, several statistical learning theory (SLT)-based methods, such as support vector machines (SVMs) and decision trees (DTs), were proposed. Currently, several machine based methods, such as deep learning for big data analysis, have created attention in both academia and industry. Machine learning is a means of automating analytical model building. Using algorithms that itself learn from data provided, machine learning allows computers to find hidden knowledge without being specially programmed where to look. Machine learning shows good applicability in classification, regression and other activities related to high-dimensional data. Aimed at extracting knowledge and gaining insight from massive databases, machine learning learns from previous computations to produce reliable, repeatable decisions and results and thus has played an significant role in many fields, especially speech recognition, image recognition, bioinformatics, information security, and natural language processing (NLP) etc. [3].

As soon as electronic computers came into use in the fifties and sixties, the algorithms were developed that enabled modeling and analyzing large sets of data. From the very beginning three major branches of machine learning emerged. Classical work in symbolic learning is described by Hunt et al. (1966), in statistical methods by Nilsson (1965) and in neural networks by Rosenblatt (1962). Through the years all three domains developed advanced methods (Michie et al., 1994): statistical or pattern recognition methods, such as the k-nearest neighbors, discriminate analysis, and Bayesian classifiers, inductive learning of symbolic rules, such as top-down induction of decision trees, decision rules and inclusion of logic programs, and artificial neural networks, such as the multi-layered feed forward neural network with back propagation learning, the Kohonen's self-organizing network and the Hopfield's associative memory. For a machine learning (ML) system to be useful in solving medical diagnostic tasks, the following features are desired: good performance, the ability to appropriately deal with missing data and with noisy data, the algorithms' ability to reduce the number of results to achieve good diagnosis, and finally the ability to explain decisions [4].

B. The types of Machine Learning Algorithms

Commonly used machine learning algorithms include: Supervised learning where the algorithm generates a function that points or maps the inputs to desired outputs. One most used method to formulize the supervised learning problem is the classification problem: the learner is required to learn a function which maps a vector into one of several classes by looking into several input-output examples of the function. Unsupervised learning is another type which models a set of inputs: labeled examples as not available. Semi-supervised learning is a combination of both labeled and unlabeled examples to generate an appropriate function or classifier. Reinforcement learning are those where the algorithm learns a policy of how to act given an observation of the world. Every action is having some impact in the environment, and the environment provides feedback that helps the learning algorithm in taking decisions. In this paper, we will be focusing on the supervised learning approach. We use the several algorithms that come under this form of Machine Learning.

C. Supervised learning Algorithms

In Supervised learning we create a classification system and then make the machine to learn this classification system by feeding it inputs related to this classification system. Digit recognition is a common example of classification learning. More generally, classification learning is appropriate for those problems where guessing a classification is of use and the classification is simple to determine. In some scenarios, it might not even be requisite to give pre-found classifications to every part of a problem if the agent can find the classifications for itself. This would be an example of unsupervised learning in a classification context. Supervised Learning [7] often leaves the probability for inputs undefined. This model is not needed as long as the inputs are available, but if some of the input values are missing, it is not possible to infer anything about the outputs. In case of unsupervised learning, all the observations are assumed to be caused by hidden variables, that is, the observations are believed to be present at the end of the progressing chain.

The algorithms used in Machine Learning of the type supervised are linear classifiers, Quadratic Classifiers, K-means clustering, Boosting, Decision Tree, Neural Networks and Bayesian Networks. In this paper, we have focused on four of the machine learning algorithms namely: logical regression and support vector machine which come under the category of linear classifiers, Random forests which are in the Decision tree and Boosting.

1. Logical Regression

Regression analysis is a method to find a functional relationship between dependent variables (response) and independent variables (predictor). In LR, the function is linear equation and dependent variable can be expressed as a function of independent variable(s) in the form of:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_n X_n$$

Where Y is dependent variable and X1 to Xn are independent variables. [9]

2. Support Vector Machine

Support Vectors are simply the co-ordinates of individual observation. For example, (45,150) is a support vector which corresponds to a female. Support Vector Machine is a frontier which best provides a distinction between Male and Females. In this, the two groups are well separate from each other; hence it is simple to find a SVM. "Support Vector Machine" (SVM) is a supervised machine learning method which is used for both classification and regression challenges. However, it is mostly used in field of classification problems. In this algorithm, we have plotted each data item as a point in n-dimensional space (where n is number of features you have) with the value of each attribute being the value of a particular (x,y) i.e. coordinates. Then, we perform classification by finding the hyper-plane that differentiates the two groups very well [9].

3. Random Forests

In Random forests we have a combination of tree predictors such that each tree depends on the values of a random vector which is sampled independently and with the same distribution for each trees in the forest. As the number of trees in the forest becomes large this generalization error converges to a limit. This error is dependent on the strength of individual trees in the forest and correlation among them. Using a random selection of features to split each node yields error rates that compare favorably to Adaboost (Freund and Schapire[1996]), but are more robust with respect to noise. Factors like error, strength and correlation are monitored using internal estimates and these are used to show the response to increasing the number of features used in the splitting. They are also used for calculating variable importance. These ideas are also applicable to regression. Random forests are an effective tool in prediction. [10]

Because of the Law of Large Numbers, they do not over fit. Injecting the right kind of randomness makes them accurate classifiers and regressors. Furthermore, the framework in terms of strength of the individual predictors and their correlations gives insight into the ability of the random forest to predict. Using out-of-bag estimation makes concrete the otherwise theoretical values of strength and correlation. [10]

4. Boosting

Boosting is a method of finding a highly accurate hypothesis (classification rule) by combining many "weak" hypotheses, each of which is only moderately accurate. Typically, each weak hypothesis is a simple rule which can be used to generate a predicted classification for any instance. Boosting is a general method for improving the accuracy of any given learning algorithm. The AdaBoost algorithm, does training error and generalization error; boosting's connection to game theory and linear programming; the relationship between boosting and logistic regression. It has extensions for multiclass classification problems; methods of incorporating human knowledge into boosting; and experimental and applied work using boosting. [11]

D. Deciding among the supervised machine learning algorithms

A common process for comparing and analyzing supervised ML algorithms is to perform statistical comparisons of the accuracies of trained classifiers on specific datasets (datasets should not change). If we have sufficient supply of data, we can sample a number of training sets of size N, run the two learning algorithms on each of them, and estimate the difference in accuracy for each pair of classifiers on a large test set. The average of these differences is an estimate of the expected difference in generalization error across all possible training sets of size N, and their variance is an estimate of the variance of the classifier in the total set. Our next step is to perform paired t-test to check the null hypothesis that the mean difference between the classifiers is zero. This test can produce two types of errors. Type I error is the probability that the test rejects the null hypothesis incorrectly (i.e. it finds a "significant" difference although there is none). Type II error is the probability that the null hypothesis is not rejected, when there actually is a difference. The test's Type I error will be close to the chosen significance level. In practice, however, we often have only one dataset of size N and all estimates must be obtained from this sole dataset. [12]

W use sub sampling to obtain different training sets and those that are not sampled are used for testing. This fails to meet the criteria of independence assumption for significance testing. As a result type I errors exceed the level of significance For a researcher this is problematic because it is important to be able to control Type I errors and know the probability of incorrectly rejecting the null hypothesis. Many versions of the t-test have been developed to address this problem (Dietterich, 1998), (Nadeau and Bengio, 2003).

Ideally, we would want the outcome to be independent of the particular partitioning resulting from the randomization process, because this would make it much easier to produce experimental results published in the literature. However, in practice there is always certain sensitivity to the partitioning used. To measure replicability, we have to repeat the same test number of times on the same data with different random partitioning — usually ten repetitions— and count how often the outcome is the matching with previous (Bouckaert, 2003). Intelligent systems are responsible for carrying out supervised learning tasks. [12].

IV. METHADOLOGY

In this paper, four supervised machine learning algorithms for prediction of heart disease are compared. For some algorithms, parameters are tuned and the best model is then selected based on the results. The best result, measured by AUC and accuracy are obtained by the logistic regression model (AUC 0.92, Accuracy=0.87), followed by gradient boosting Machines. We have taken a set of 14 different variables to predict heart disease, the most important parameter to predict heart failure being whether or not there is a reversible defect in thalassemia followed by whether there is occurrence of asymptomatic chest pain or not.

A. Datasets

For this we have taken nicely prepared Cleveland heart dataset which is available at UCI. The document mentions that previous predictions on the Cleveland heart disease data resulted in 74-77% accuracy. The 14 variables defined are as follows:- 1.age: Age of patient,2.sex:Sex,1for male,3.cp:chest pain,4.trestbps:resting blood pressure,5.chol:serumcholesterol,6.fbs:fasting blood pressure,7.restecg:resting elctroc.result (1 anomality),8.thalach:maximum heart rate achieved,9.exang:exercise induction angina(1 yes),10.oldpeak:ST depression induc.ex,11.slope:slope of peak exercise ST.,12.ca:number of major vessels,13.thal:no explanation provided, but probably thalassemia (3 normal; 6 fixed defect; 7 reversible defect),14.num.diagonosis/status of heart disease. Now the variable that we want to predict is num with value 0, i.e.<50% diameter narrowing and >50% diameter narrowing. Our assumption will be that every value with 0 means that heart is okay, and if the values are 1,2,3,4 then heart disease.

B. Data preparation

First we load the data set in R,ie

```
> heart.data <- read.csv("https://archive.ics.uci.edu/ml/machine-learning-databases/heart-disease/processed.cleveland.data", header = FALSE, sep = ",", na.strings = '?')
#naming the coloumns of our data set.
+ names(heart.data) <- c("age", "sex", "cp", "trestbps", "chol", "fbs", "restecg",
+                "thalach", "exang", "oldpeak", "slope", "ca", "thal", "num")
```

We can view this data by using view(heart.data).

Next we check for any missing values and remove them,we see there are 6 missing values so just omit them.

```
> s = sum(is.na(heart.data))
+ heart.data <- na.omit(heart.data)
+ #str(heart.data)
```

C. Training/Testing of data for Validation

Here we will we splitting our dataset in two ranges one consisting of trained data(70%) and other consisting of testing data.(30%).Important thing is percentage of heart disease or not must be same in both the ranges which will be handled by r libraries(library caret) used here.

```
>library(caret)
+ set.seed(10)
+ inTrainRows <- createDataPartition(heart.data$num, p = 0.7, list = FALSE)
+ trainData <- heart.data[inTrainRows,]
+ testData <- heart.data[ - inTrainRows,]
+ nrow(trainData) / (nrow(testData) + nrow(trainData)) #checking whether really 70% -> OK
Loading required package: lattice
Loading required package: ggplot2
[1] 0.7037037
```

Next outputs are to be stored in variable AUC. AUC is the area under the ROC which represents the proportion of positive data points that are correctly considered as positive and the proportion of negative data points that are mistakenly considered as positive. We also store Accuracy which is true positive and true negative divided by all results. So we have tow variables defined as follows:-

```
> AUC = list()
+ Accuracy = list()
```

Now before we can move on considering the models one by one we need a function to convert the class of predictor values for all 14 different variables. This function we define as convert. magic and is defined as follows:-


```
>#Function needed to convert classes of predictor values
+ convert.magic <- function(obj, types) {
+    for (i in 1:length(obj)) {
+       FUN <- switch(types[i], character = as.character,
+                              numeric = as.numeric,
+                              factor = as.factor)
+       obj[, i] <- FUN(obj[, i])
+    }
+    obj
+ }
+
+ convert.names <- function(row) {
+    row = gsub("sex1", "male", row)
+    row = gsub("thal7", "reversable defect thalassemia", row)
+    row = gsub("thal6", "fixed defect thalassemia", row)
+    row = gsub("cp4", "asymptomatic chest pain", row)
+    row = gsub("cp3", "non-anginal chest pain", row)
+    row = gsub("cp2", "atypical angina chest pain", row)
+    row = gsub("oldpeak", "ST depression from exercise", row)
+    row = gsub("thalach", "maximum heart rate achieved", row)
+    row = gsub("trestbps", "resting blood pressure", row)
+    row = gsub("ca2", "2 major vessels col/b fluoro., ca2", row)
+    row = gsub("ca1", "1 major vessel col/b fluoro., ca1", row)
+    row = gsub("slope2", "flat peak exercise ST segment", row)
+    row = gsub("slope1", "upsloping peak exercise ST segment", row)
+    row = gsub("slope3", "downsloping peak exercise ST segment", row)
+    row = gsub("chol", "serum cholestoral", row)
+    row = gsub("exang", "exercise induced angina", row)
+    row = gsub("restecg2", "restec: showing left ventricular hypertrophy
+                by Estes criteria", row)
+    row = gsub("restecg1", "restec: having ST-T wave abnormality", row)
+    row = gsub("fbs1", "fasting blood sugar > 120 mg/dl", row)
+ }
```

D.Logistic Regression (One model)

Now once we have created partitions for trained and tested data sets and function for converting the classes of predictor values we can move on to comparative study between various models. First we discuss the code implemented for logistic regression.

```
set.seed(10)
+ logRegModel <- train(num ~ ., data = trainData, method = 'glm', family = 'binomial')
+ logRegPrediction <- predict(logRegModel, testData)
+ logRegPredictionprob <- predict(logRegModel, testData, type = 'prob')[2]
+ logRegConfMat <- confusionMatrix(logRegPrediction, testData[, "num"])
+ #ROC Curve
+ library(pROC)
+ AUC$logReg <- roc(as.numeric(testData$num), as.numeric(as.matrix((logRegPredictionprob))))$auc
+ Accuracy$logReg <- logRegConfMat$overall['Accuracy'] #found names with str(logRegConfMat)
+
```

We will compare the results in results section

E. Random forests without tuning (but checked a few number of trees)

R code:

```
>library(randomForest)
+ set.seed(10)
+ RFModel <- randomForest(num ~ .,
+                 data = trainData,
+                 importance = TRUE,
+                 ntree = 2000)
+ #varImpPlot(RFModel)
+ RFPrediction <- predict(RFModel, testData)
```

```
+ RFPredictionprob = predict(RFModel, testData, type = "prob")[, 2]
+
+ RFConfMat <- confusionMatrix(RFPrediction, testData[, "num"])
+
+ AUC$RF <- roc(as.numeric(testData$num), as.numeric(as.matrix((RFPredictionprob))))$auc
+ Accuracy$RF <- RFConfMat$overall['Accuracy']
```

F. Boosted Trees with tuning (grid search).

Boosted tree model with adjusted learning rate and trees

R code:.

```
library(caret)
+ set.seed(10)
+ objControl <- trainControl(method = 'cv', number = 10, repeats = 10)
+ gbmGrid <- expand.grid(interaction.depth = c(1, 5, 9),
+                        n.trees = (1:30) * 50,
+                        shrinkage = 0.1,
+                        n.minobsinnode = 10)
+ # run model
+ boostModel <- train(num ~ ., data = trainData, method = 'gbm',
+                     trControl = objControl, tuneGrid = gbmGrid, verbose = F)
+ # See model output in Appendix to get an idea how it selects best model
+ #trellis.par.set(caretTheme())
+ #plot(boostModel)
+ boostPrediction <- predict(boostModel, testData)
+ boostPredictionprob <- predict(boostModel, testData, type = 'prob')[2]
+ boostConfMat <- confusionMatrix(boostPrediction, testData[, "num"])
+
+ #ROC Curve
+ AUC$boost <- roc(as.numeric(testData$num), as.numeric(as.matrix((boostPredictionprob))))$auc
+ Accuracy$boost <- boostConfMat$overall['Accuracy']
```

G. Support Vector Machine

R code:

```
>set.seed(10)
+ svmModel <- train(num ~ ., data = trainData2,
+                   method = "svmRadial",
+                   trControl = fitControl,
+                   preProcess = c("center", "scale"),
+                   tuneLength = 8,
+                   metric = "ROC")
+ svmPrediction <- predict(svmModel, testData2)
+ svmPredictionprob <- predict(svmModel, testData2, type = 'prob')[2]
+ svmConfMat <- confusionMatrix(svmPrediction, testData2[, "num"])
+ #ROC Curve
+ AUC$svm <- roc(as.numeric(testData2$num), as.numeric(as.matrix((svmPredictionprob))))$auc
+ Accuracy$svm <- svmConfMat$overall['Accuracy']
+
```

V. RESULTS & DISCUSSION

Here we will be comparing the AUC and accuracy among different models. To output the comparison, we have the following code in R:

```
> row.names <- names(Accuracy)
+ col.names <- c("AUC", "Accuracy")
+ cbind(as.data.frame(matrix(c(AUC, Accuracy), nrow = 5, ncol = 2,
+                            dimnames = list(row.names, col.names))))
+
```

We get the following output:


```
> row.names <- names(Accuracy)
+ col.names <- c("AUC", "Accuracy")
+ cbind(as.data.frame(matrix(c(AUC, Accuracy), nrow = 5, ncol = 2,
+                             dimnames = list(row.names, col.names))))
+
             AUC  Accuracy
logReg 0.9161585 0.8651685
RF     0.8932927 0.8089888
boost  0.9095528 0.8426966
gbm    0.9070122 0.8426966
svm     0.882622 0.7977528
```

Fig:1

From the above comparison, we say that simple logistic regression is the best model with an area under ROC of 0.92.We can predict heart disease with an accuracy of 0.87.Now we use logRegConfMat to see more clearly our results on logistic regression. We get the following output in R.

```
> logRegConfMat
Confusion Matrix and Statistics

          Reference
Prediction  0  1
         0 43  7
         1  5 34

               Accuracy : 0.8652
                 95% CI : (0.7763, 0.9283)
    No Information Rate : 0.5393
    P-Value [Acc > NIR] : 5.93e-11

                  Kappa : 0.7277
 Mcnemar's Test P-Value : 0.7728

            Sensitivity : 0.8958
            Specificity : 0.8293
         Pos Pred Value : 0.8600
         Neg Pred Value : 0.8718
             Prevalence : 0.5393
         Detection Rate : 0.4831
   Detection Prevalence : 0.5618
      Balanced Accuracy : 0.8626

       'Positive' Class : 0

>
```

Fig:2

We observe that sensitivity is 0.8958 and specificity is 0.8293.
We can summarize the logistic regression model in r as follows:

However, a direct comparison of importance of each predictor variable is not possible for the logistic regression model. But this can be added in further analysis, comparing the predictive ability of model by removing each variable separately. In terms of accuracy boosted tree model was slightly lower than logistic regression, so we will show the significance of the variables calculated by boosted tree model.

For this we will call following function:

```
> boostImp = varImp(boostModel, scale = FALSE)
+ row = rownames(varImp(boostModel, scale = FALSE)$importance)
+ row = convert.names(row)
```

```
+ rownames(boostImp$importance) = row
+ plot(boostImp, main = 'Variable importance for heart failure prediction with boosted tree')
>
```

We get the following plot:

```
> summary(logRegModel)$coeff
               Estimate  Std. Error    z value     Pr(>|z|)
(Intercept) -2.83507855 3.547759084 -0.7991181 0.4242219263
age         -0.03180882 0.029637725 -1.0732545 0.2831569344
sex1         1.85291093 0.711216844  2.6052686 0.0091802253
cp2          0.44982427 1.005276129  0.4474634 0.6545405141
cp3         -0.51437449 0.860554826 -0.5977243 0.5500239377
cp4          1.64003231 0.868294403  1.8887975 0.0589189654
trestbps     0.01714664 0.015671641  1.0941190 0.2739027719
chol         0.00340254 0.004918579  0.6917730 0.4890799254
fbs1        -0.24155269 0.804218007 -0.3003572 0.7639046843
restecg2     0.25036985 0.492666693  0.5081932 0.6113178838
thalach     -0.02492582 0.014363737 -1.7353301 0.0826823533
exang1       0.52595947 0.555748185  0.9463989 0.3439451656
oldpeak      0.20695564 0.284315718  0.7279078 0.4666700014
slope2       1.72232845 0.616791265  2.7924008 0.0052318499
slope3       1.03760679 1.069301753  0.9703592 0.3318674790
ca1          2.56517830 0.696934041  3.6806615 0.0002326296
ca2          3.94566322 0.994034355  3.9693429 0.0000720711
ca3          2.16861195 1.010144229  2.1468340 0.0318065017
thal6       -0.40962979 1.003136121 -0.4083492 0.6830173544
thal7        1.58273584 0.549746570  2.8790281 0.0039890275
>
```

Fig 3:

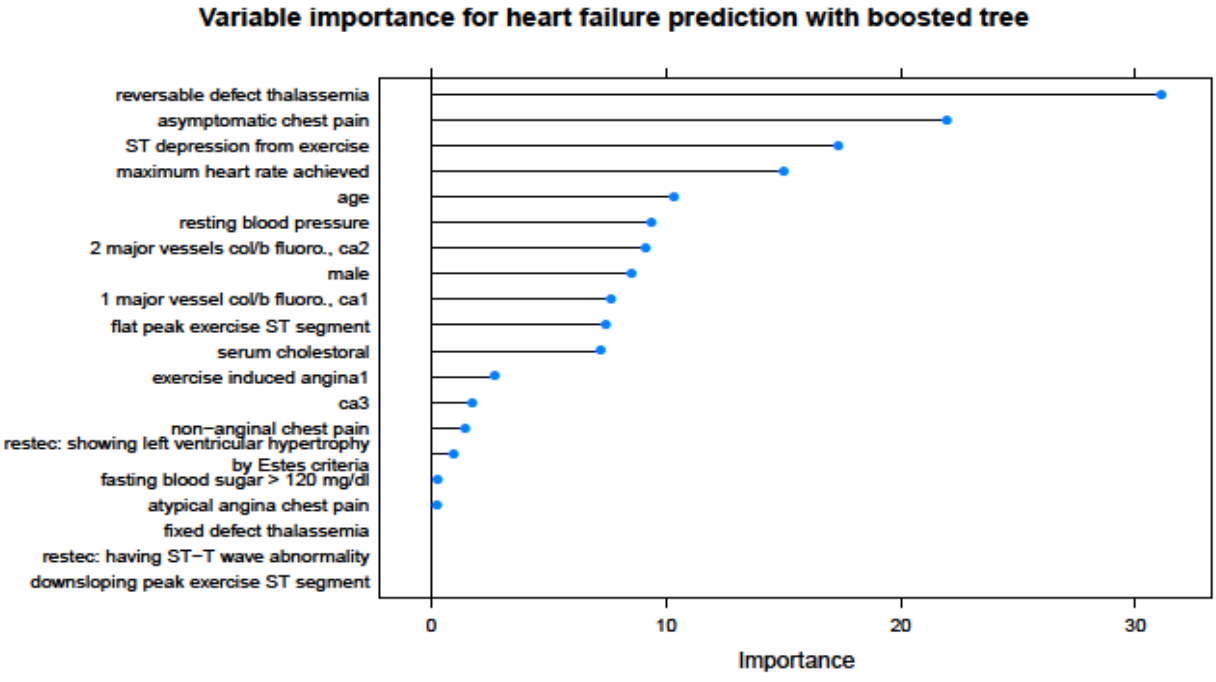

Fig 4:

VI.CONCLUSION

Performance of 4 different machine learning algorithms-logistic regression, random forests, boosted trees and support vector machines were compared on UCI heart disease data sets based 14 different predictor variables. We divided the data sets into training and testing data set, 30% of data is hold out as testing that is not seen in training data set. During the training of boosted trees and support vector machines cross validation is used to maximize the ROC (parameter tuning) and maximize the result. Logistic regression performed the best with accuracy .87 while tree based models with different tuning parameters performed slightly worse.

Boosted tree model was used to compare importance of different variables due to easier procedure and it turned out that having a reversible defect thalassemia is most important factor followed by asymptomatic chest pain. This overall analysis showed working of various machine learning algorithms on heart disease data set. Several improvements can be made by data preprocessing techniques such as use of outliers, variances etc., Choice of model, model tuning parameters and so on.

REFERENCES

1. https://en.wikipedia.org/wiki/Machine_learning
2. Sajda P, Machine learning for detection and diagnosis of disease, NCBI, 2006
3. YueLiu, WangweiJu, SiqiShi and TianluZhao, Materials discovery and design using machine learning, Journal of Materiomics, ScienceDirect, 2017.
4. Igor Kononenko, Machine Learning for Medical Diagnosis: History, State of the Art and Perspective, Science Direct, 2001.
5. Taiwo Oladipupo Ayodele,Types of Machine Learning Algorithms, INTECH, 2010
6. http://www.aihorizon.com/essays/generalai/supervised_unsupervised_machine_learning.htm
7. http://users.ics.aalto.fi/harri/thesis/valpola_thesis/node34.html
8. Erdi Tosun, Kadir Aydin and Mehmet Bilgili, Comparison of linear regression and artificial neural network model of a diesel engine fuelled with biodiesel-alcohol mixtures, Science Direct, 2016. https://www.analyticsvidhya.com/blog/2014/10/support-vector-machine-simplified/
9. Leo Breiman, Random Forests, Springer, 2001
10. Robert E. Schapire, The Boosting Approach to Machine Learning An Overview, Nonlinear Estimation and Classification, Springer, 2003
11. S.B. Kotsiantis, Supervised machine learning: A review of Classification Techniques, Informatica 31, 2007.
12. Shashikant U. Ghumbre and Ashok A. Ghatol, Heart Disease Diagnosis Using Machine Learning Algorithm, Springer, 2012.
13. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.401.7117&rep=rep1&type=pdf
14. Meherwar Fatima and Maruf Pasha, Survey of Machine Learning Algorithms for Disease Diagnostic, Journal of Intelligent Learning Systems and Applications, 2017
15. Priya R. Patil and S. A. Kinariwala, Automated Diagnosis of Heart Disease using Random Forest Algorithm, International Journal of Advance Research, Ideas and Innovations in Technology. https://github.com/mbbrigitte/Predicting_heart_disease_UCI/blob/master/heartdisease_UCI.Rmd
16. UCI Machine Learning Repository from http://archive.ics.uci.edu/ml/datasets/Heart+Disease
17. M. Karaolis, J. A. Moutiris, L. Papaconstantinou, and C. S. Pattichis(2009), "Association rule analysis for the assessment of the risk of coronary heart events," in Proc. 31st Annu. Int. IEEE Eng. Med. Biol. Soc. Conf., Minneapolis, MN, Sep. 2–6, pp. 6238–6241.

