Classification of Arrhythmia Using Conjunction of Machine Learning Algorithms and ECG Diagnostic Criteria

ANISH BATRA
Delhi Technological University
Department of Applied Mathematics
Main Bawana Road, New Delhi
INDIA
anishbatra95@gmail.com

VIBHU JAWA
Delhi Technological University
Department of Applied Mathematics
Main Bawana Road, New Delhi
INDIA
vibhu.jawa@gmail.com

Abstract: This paper proposes a classification technique using conjunction of Machine Learning Algorithms and ECG Diagnostic Criteria which improves the accuracy of detecting Arrhythmia using Electrocardiogram (ECG) data. ECG is the most widely used first line clinical instrument to record the electrical activities of the heart. The data-set from UC Irvine (UCI) Machine Learning Repository was used to implement a multi-class classification for different types of heart abnormalities. After implementing rigorous data preprocessing and feature selection techniques,different machine learning algorithms such as Neural Networks, Decision trees, Random Forest, Gradient Boosting and Support Vector Machines were used. Maximum experimental accuracy of 84.82% was obtained via the conjunction of SVM and Gradient Boosting. A further improvement in accuracy was obtained by validating the factors which were important for doctors to decide between normal and abnormal heart conditions.The performance of classification is evaluated using measures such as confusion matrix, kappa-score, confidence interval, Area under curve (AUC) and overall-accuracy.

Key–Words: Machine Learning, Arrhythmia Classification, ECG, Neural Networks, SVM, Gradient Boosting

1 Introduction

One of the most prominent method to diagnose heart diseases is using Electrocardiogram(ECG) signals. ECG signals comprises of P wave,T wave, and QRS complex.

In the normal ECG signal of a heart, the main parameters which are examined include the duration, the shape and the correlation between the P wave, QRS complex, and T wave components and R-R interval. The variation in these parameters is used to identify the type of illness of the heart. The beat phases which are irregular are generally called arrhythmia and some arrhythmias can be fatal for the patient.

Various computer-based automatic ECG interpretation systems are currently being developed to diagnose arrhythmia in time, and various machine-learning based models are applied to these systems.

In this study, a conjunction of Machine Learning Algorithms and ECG Diagnostic criteria were used to detect various types of arrhythmia.

2 Related Work

Many methods for automated arrhythmia detection and classification have been developed in the past to make the monitoring task easier.[1, 2].

The methods include self-organizing map , Wavelet transformation[3], Logically Weighted Regression (LWR) methods, fuzzy cmeans clustering techniques, and machine learning classifiers.

A review of various machine learning classifiers specifically Logistic Regression, Decision Trees, Naive-Bayes and RBF Networks and their performances compared with other types of classifiers can be found in [4, 5].

Most of the above mentioned classifiers have their experimental accuracies in the range of 60%-76% .

The approach developed in the research defined in this paper was to evaluate various standard machine learning algorithms applied to classify cardiac arrhythmias and develop a model using amalgamation of various machine learning techniques and ECG diagnostic criteria to develop a method with an improved accuracy and efficiency.

3 Data Set

The Arrhythmia dataset found at the UCI Machine Learning Repository [6] (Bache and Lichman, 2013; Guvenir, 1997) was used in the presented research paper.

The dataset consists of 452 records with 279 features per record. Every record is assigned to 1 of the 16 classes: a class label of 1 defines regular ECG patterns while a class label between 2 to 16 is used for identifying different types of arrhythmia.

The datasets class distribution is shown in figure 1.

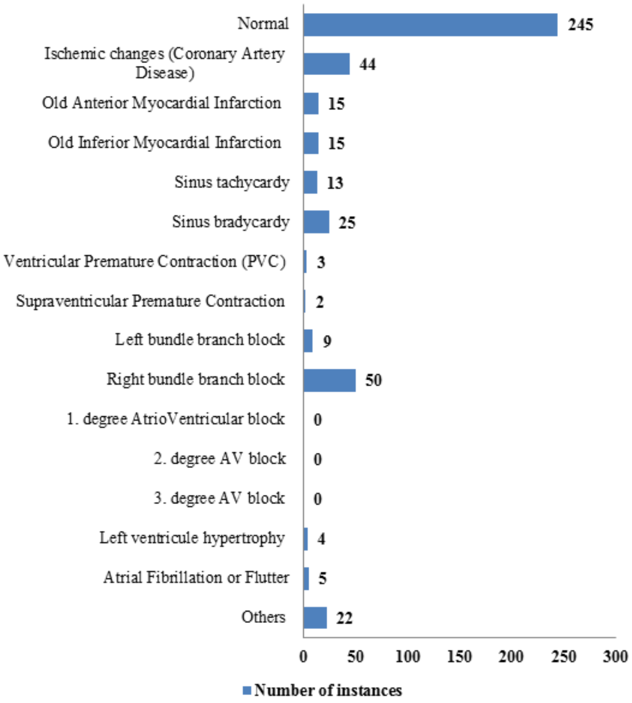


Figure 1: Data Set Description

The data was divided randomly into training and test-set as follows:

- Training Set: 75%
- Testing Set: 25%

4 Analysis and Data preprocessing

4.1 Data Preprocessing

The following steps were undertaken to pre-process the data-set:

- The missing values in features were identified and replaced with their arithmetic means over all the observations for that variable.
- Invariant features:

  The categorical features that were indicating either all 0s or all 1s were removed. These features did not contribute in a significant manner in determining the type of arrhythmia as they were pointing to the same category most of the time. Their removal, did not affect the accuracy but minimized the feature space and simplified the database. To serve this purpose, the summary function in R was used to check which features had the same min and max value; meaning that these features were invariant. This reduced the feature set from 279 to 198.

- Data Normalization:
  The algorithms we are employing for classification expect a normalized dataset, all the parameters that were not discrete in nature were normalized using the Z-score normalization technique. Z-scores are expressed in terms of standard deviations from their means. The distribution of z-scores obtained has a mean of 0 and a standard deviation of 1.

- Principal Component Analysis:
  Principal Component Analysis is a statistical procedure, which by using an orthogonal transformation converts a set of possibly correlated variables into a set of linearly uncorrelated variables called the principal components. Therefore, the number of principal components is either equal or less than the number of original variables. After applying the above mentioned steps, PCA was used convert the set of observations with 95% correlation into a set of less than 100 values of linearly uncorrelated variables.

We then proceeded towards analyzing the data using ECG diagnostic criteria.

4.2 Data analysis using ECG diagnostic Criteria

As some of the classes in the dataset have very few instances, an accurate machine learning training model cannot be achieved for them. Thus to identify these scarce classes accurately, the criteria given by ECG analysis from a medical point of view were used.[7]
Left Bundle Branch Block or class 9 of arrhythmia given in the data-set can be detected using the following diagnostic criteria:

- QRS duration of >120ms.
- Dominant S wave in V1
- Broad monophasic R wave in lateral leads (I, aVL, V5-V6)
- Absence of Q waves in lateral leads (I, V5-V6)

- Prolonged R wave peak time >60ms in left precordial leads (V5-6).

After applying the above mentioned filters, an accuracy of 100% for class 9 was observed in the results. We tried applying the same technique to other arrhythmia classes too, but due to the limitations in the dataset and lack of clear-cut criteria for identification of those particular classes, an accuracy of 100% could not be achieved. The algorithms and the above mentioned technique gave similar accuracies for classes other than 9, so this technique was avoided for them.

5 Models Used and Results

5.1 Neural Networks

These are information-processing models that are influenced, as the name suggests, by the way the nervous system of the human processes information. An important aspect of this information paradigm is its inimitable structure. A large number of highly interconnected processing paradigms (commonly referred to as neurons) work in tandem to solve very specific problems. Akin to people, Neural Networks also learn by example.[8]

Neural networks are essentially simple mathematical models which define a function:

$$f : X \to Y \tag{1}$$

or a distribution over X or both X and Y, but sometimes the models are also intimately associated with some learning algorithm or a learning rule.

Better accuracy was observed after pre-processing the data-set using Principal Component Analysis technique.

The neural networks model was trained several times varying the number of hidden layers as well the decay factor (parameter to effectively limit the amount of free parameters in the model, thus avoiding overfitting), fine-tuning the model for maximum accuracy. The best results were obtained with 1 hidden layer and 13 neurons.

The results of the Neural Network are displayed in table 1.

Table 1: Neural Network Results

Parameters	All features	Feature Selection with PCA
Overall Accuracy	0.6792	0.7570
Confidence Inteval	(0.5816, 0.7666)	(0.6646,0.8347)
Kappa	0.4283	0.6
AUC	0.632	0.7620

5.2 Decision Trees

In layman terms decision trees are simply tools that help choose between several courses of action. They let us investigate all possible outcomes by laying out all possible options.

Decision trees implicitly achieve feature selection, and are also able to tackle non-linear relationships between various parameters. They provide a mechanism to quantify the possibility of a particular event/class in terms of the probability of occurrence.

A decision tree is a linear structure where the outcome of the event is given by the leaf node, and the conditions along the path gradually filter out and map the outcome to one particular possibility.[9]

The decision tree can be viewed as decision rules, where the final result is found in the contents of the leaf node, and the conditions along the path of the tree form a conjunction in the if clause.

In general, the rules are somewhat like:

If condition1 and condition2 and condition3 then outcome.

The decision tree, as shown in figure 2 was obtained when the dataset was trained using this algorithm.

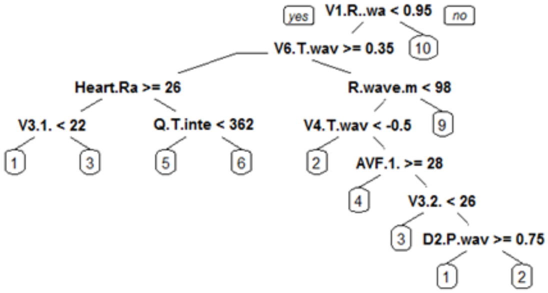


Figure 2: Decision Tree

The results on the testing data are presented in table 2.

Table 2: Decision Tree Results

Parameters	All features	Feature Selection with PCA
Overall Accuracy	79.459	58.04
Confidence Interval	(0.708 ,0.865)	(0.483,0.67)
Kappa	0.670	0.277
AUC	0.836	0.752

5.3 Random Forest

Random forest is a collaborative learning technique for regression, classification and other tasks, which functions by constructing a number of decision trees at the time of the training, and outputting the class which is the mode of the classes (in case of classification) or mean prediction (in case of regression) of the individual trees. It is an amalgamation of prediction trees (decision trees) where the outcome of each tree depends on the value of a random vector sampled independently with the same amount of distribution for all of the trees in the forest.

Random Forests attempts to alleviate the problems of high variance and high bias, often found in single decision trees, by averaging to find a balance between the two boundaries.

In case of classification problems, given a set of random predictor variables and simple trees, the Random Forest algorithm establishes a margin function, which measures the degree to which the average number of votes for the actual class surpasses the average vote for any other class present in the dependent variable. Random Forests grows a number of classification trees. If we want to classify a new object, we put the input vector down each of the trees; every tree giving out a classification. Thus every tree votes for a particular class. The class having the most number of votes is chosen as the outcome of the random forest. [10]

The average predictions of the trees are taken to be the outcome prediction of the Random Forest:

$$RFPredictions = \frac{1}{k}\sum_{1}^{k} treeresponse^{kth} \qquad (2)$$

Where the index k runs over each tree individually in the forest. The Dot-chart of variable importance as measured by the random forest model on the data-set is shown in figure 3.

As can be observed from figure 4, the error becomes constant after the number of trees reaches 450. The results obtained for the random forest model are shown in figure 3. Type of random forest: classification

Number of trees: 451

No. of variables tried at each split: 14

OOB estimate of error rate: 27.35%

Table 3: Random Forest results

Parameters	All features	Feature Selection with PCA
Overall Accuracy	0.785	0.633
Confidence Interval	(0.6981,0.8576)	(0.5376,0.7229)
Kappa-Value	0.642	0.246
AUC	0.885	0.564

5.4 Gradient Boosting

Gradient boosting is a machine learning technique for classification and regression problems, which typically uses decision trees to produce a model in the form of an ensemble of weak prediction models. We used the xgboost package in R to implement gradient boosting. XGBoost is short for eXtreme gradient boosting, which produces a prediction model as that of a collaboration of weak prediction models. Like other boosting methods, it creates the model in a gradual fashion, and it generalizes them by allowing optimization of a random differentiable loss function.[11]

In many supervised learning problems there is an output variable y and a vector of input variables x which are associated together through a joint probability distribution P(x, y).

Using a training set (x1,y1),.,(xn,yn) with known values of x and the corresponding values of y, the goal is

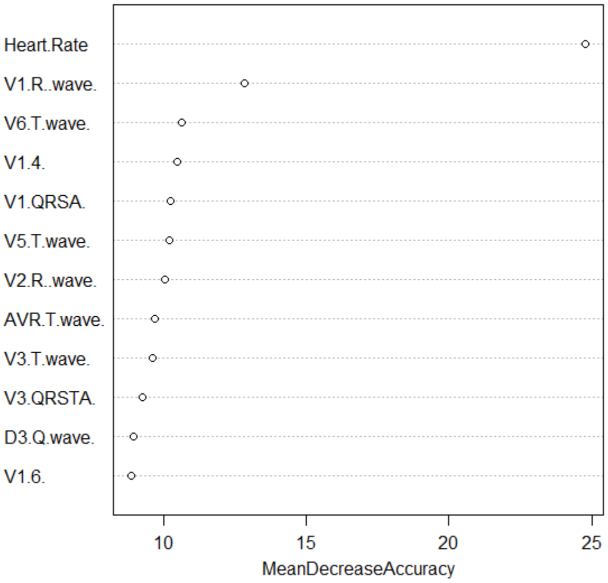


Figure 3: Variable importance dot-chart

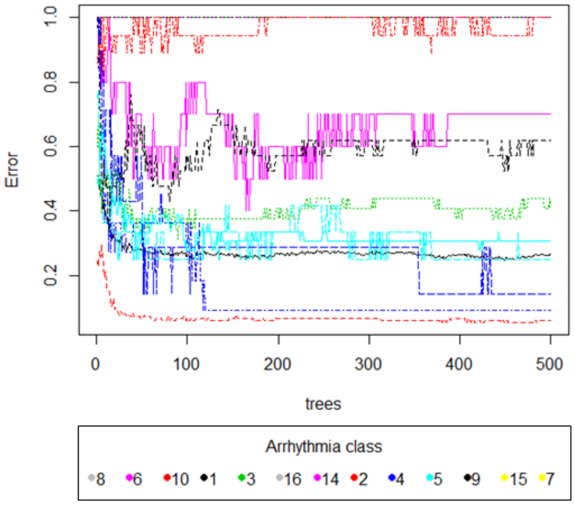


Figure 4: Number of trees vs Error

to find an approximation $F\hat{(x)}$ to a function $F\acute{(x)}$ which minimizes the expected value of a specified loss function L(y,F(x)):

$$F^* = arg\, min_F\, Emin_{x,y}\left[L(y,F(x))\right] \qquad (3)$$

The top 12 features achieved on running the gradient boosting on the model are displayed in figure 5. And the results obtained are shown in table 4.

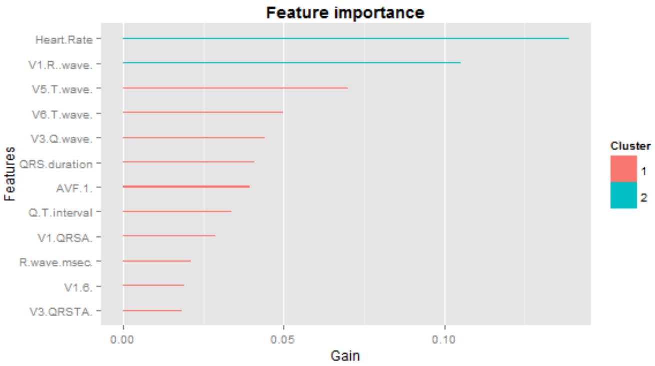


Figure 5: Top features of Gradient Boosting

Table 4: Gradient Boosting Results

Parameters	All features
Overall Accuracy	80.36
Confidence Inteval	(0.7178 , 0.872)
P-Value	0.682
AUC	0.837

The following parameter settings were used for the Gradient Boosting algorithm:

- 5 fold cross validation
- objective- multi:softmax
- eval_metric- mlogloss
- Max number of iterations- 20

5.5 Support Vector Machines

SVMs are supervised learning algorithms, which examine data and identify patterns, and are used for the classification problems and regression analysis. An SVM algorithm attempts to classify new examples into either of the two given categories, based on the set of training data making it a non-probabilistic linear classifier. It represents the training set points as points in space, mapped in a manner such that the separate categories are separated by as wide a gap as possible. New data points are then plotted on the same space such that they belong to either of the two categories. [?]

We trained the SVM model using the following feature selection techniques:

- All the features in our dataset
- The top 60 features given by the Gradient Boosting model.
- The top features given by the decision tree model.
- The features given by Principal Component Analysis.

The results of the Support Vector Machine are depicted in table 5 and figure 6.

Table 5: SVM Results using different algorithms for feature selection

Parameters	All features	PCA	Boosting	RF
Overall Accuracy	0.7760	0.73209	0.83040	0.7768
C.Interval Lower	0.6884	0.64019	0.7378	0.6884
C.Interval Upper	0.85	0.8114	0.8947	0.85
Kappa	0.6401	0.5632	0.7401	0.6522
AUC	0.7476	0.733	0.801	0.714

The algorithm gave best results with the following parameter settings:

- Kernel- Radial
- Cost- 100
- Gamma- 0.001

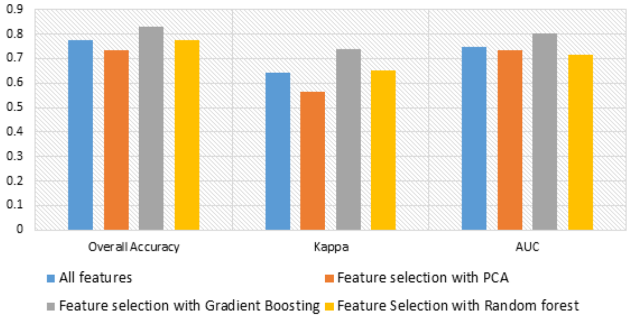


Figure 6: Accuracy of SVM for Different Feature Selection Methods

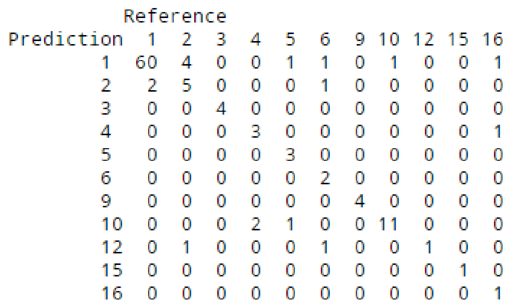
6 Conclusion

We observed maximum accuracy after training the SVM model using the top 60 features selected by the gradient boosting technique. An accuracy of 83.04%

was observed using this combination.
We then applied ECG diagnostic criteria as mentioned in section 4.2 to further increase the overall-accuracy to 84.82%. Maximum kappa value of 0.7706 was obtained using the same technique; whereas Random Forest Algorithm gave the best value for AUC of 0.885.
The results of the above mentioned modelling method is displayed in table 6; the confusion matrix in figure 7 and statistics by class in figure 8.

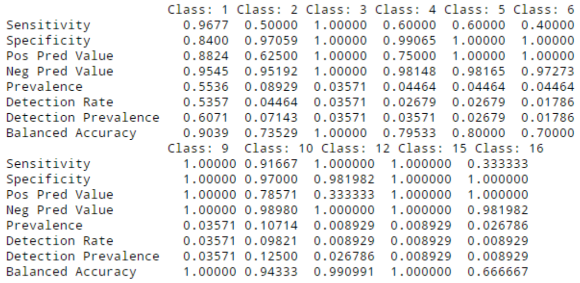
Table 6: Conjunction of Gradient Boosting and SVM results

Parameters	Without ECG Analysis	With ECG Analysis
Overall Accuracy	0.8304	0.8481
Confidence Inteval	(0.7378,0.8947)	(0.7681, 0.909)
P-Value	0.740	0.7705
AUC	0.801	0,8436



```
                 Reference
Prediction  1  2  3  4  5  6  9 10 12 15 16
        1  60  4  0  0  1  1  0  1  0  0  1
        2   2  5  0  0  0  1  0  0  0  0  0
        3   0  0  4  0  0  0  0  0  0  0  0
        4   0  0  0  3  0  0  0  0  0  0  1
        5   0  0  0  0  3  0  0  0  0  0  0
        6   0  0  0  0  0  2  0  0  0  0  0
        9   0  0  0  0  0  0  4  0  0  0  0
        10  0  0  0  2  1  0  0 11  0  0  0
        12  0  1  0  0  0  1  0  0  1  0  0
        15  0  0  0  0  0  0  0  0  0  1  0
        16  0  0  0  0  0  0  0  0  0  0  1
```

Figure 7: Confusion matrix with maximum accuracy

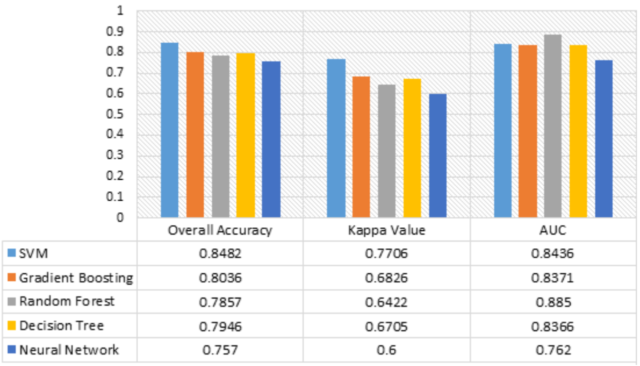


```
                     Class: 1 Class: 2 Class: 3 Class: 4 Class: 5 Class: 6
Sensitivity            0.9677  0.50000  1.00000  0.60000  0.60000  0.40000
Specificity            0.8400  0.97059  1.00000  0.99065  1.00000  1.00000
Pos Pred Value         0.8824  0.62500  1.00000  0.75000  1.00000  1.00000
Neg Pred Value         0.9545  0.95192  1.00000  0.98148  0.98165  0.97273
Prevalence             0.5536  0.08929  0.03571  0.04464  0.04464  0.04464
Detection Rate         0.5357  0.04464  0.03571  0.02679  0.02679  0.01786
Detection Prevalence   0.6071  0.07143  0.03571  0.03571  0.02679  0.01786
Balanced Accuracy      0.9039  0.73529  1.00000  0.79533  0.80000  0.70000
                     Class: 9  Class: 10 Class: 12 Class: 15 Class: 16
Sensitivity            1.00000 0.91667  1.000000  1.000000  0.333333
Specificity            1.00000 0.97000  0.981982  1.000000  1.000000
Pos Pred Value         1.00000 0.78571  0.333333  1.000000  1.000000
Neg Pred Value         1.00000 0.98980  1.000000  1.000000  0.981982
Prevalence             0.03571 0.10714  0.008929  0.008929  0.026786
Detection Rate         0.03571 0.09821  0.008929  0.008929  0.008929
Detection Prevalence   0.03571 0.12500  0.026786  0.008929  0.008929
Balanced Accuracy      1.00000 0.94333  0.990991  1.000000  0.666667
```

Figure 8: Statistics by class

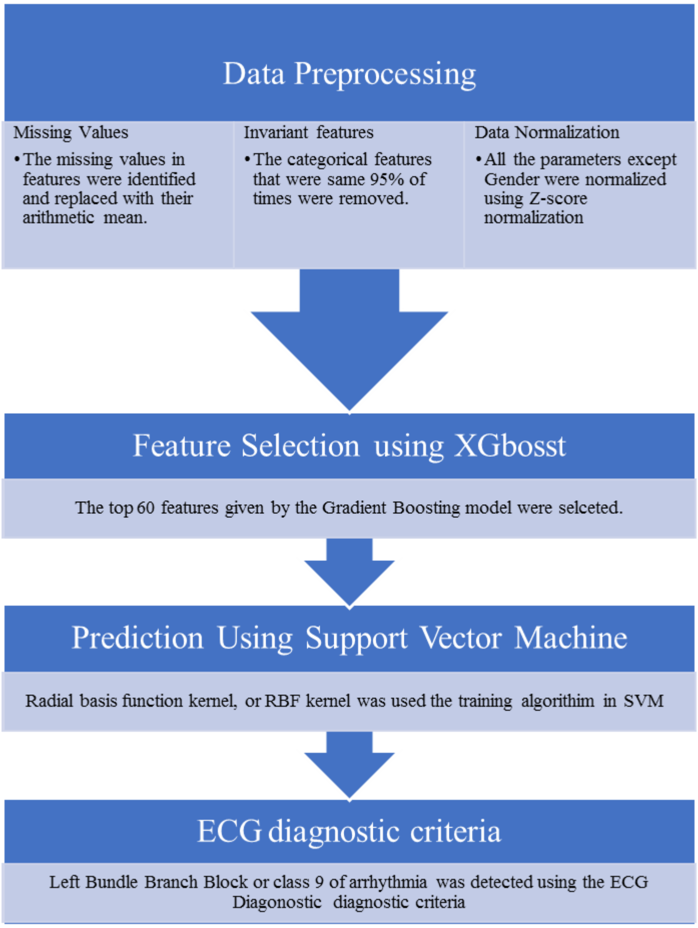
The comparison of various algorithms is displayed in figure 9.

The algorithm which achieved maximum accuracy is depicted in figure 10.



	Overall Accuracy	Kappa Value	AUC
SVM	0.8482	0.7706	0.8436
Gradient Boosting	0.8036	0.6826	0.8371
Random Forest	0.7857	0.6422	0.885
Decision Tree	0.7946	0.6705	0.8366
Neural Network	0.757	0.6	0.762

Figure 9: Comparison of accuracies of machine learning models



- The missing values in features were identified and replaced with their arithmetic mean.
- The categorical features that were same 95% of times were removed.
- All the parameters except Gender were normalized using Z-score normalization

The top 60 features given by the Gradient Boosting model were selceted.

Radial basis function kernel, or RBF kernel was used the training algorithim in SVM

Left Bundle Branch Block or class 9 of arrhythmia was detected using the ECG Diagonostic diagnostic criteria

Figure 10: Algorithm which achieved maximum accuracy

7 Future Work

A number of combinations of algorithms can be implemented in the hierarchical scheme. Currently, a combination of Gradient Boosting and SVM is implemented. It can be expanded to add more levels and models. Also, the current results of our models are limited by the small amount of data and the missing data in some features. Incorporating more accurate and precise patient instances may improve the accuracy results. Since we are provided with ECG raw data, methods like FFT and wavelet decomposition can be used to gain new features that cannot be

easily recognized in time domain. Then we need to give these features some physiological explanations. Moreover, we can also use deep learning methods to generate new features.
We can try to improve our classifier to be able to work on real-time data such as ECG signal. Currently, we are able to determine only class 9 entirely through ECG Diagnostic criteria; we can expand this for other class too, especially for the one with less number of instances to improve the accuracy even further.

Acknowledgements: The authors would like to thank R-Bloggers.com for providing a central hub of resources for the R-language.

References:

[1] Y. Tsai, B. Hung, and S. Tung, "An experiment on ecg classification using back-propagation neural network," in *Engineering in Medicine and Biology Society, 1990., Proceedings of the Twelfth Annual International Conference of the IEEE*. IEEE, 1990, pp. 1463–1464.

[2] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representations by error propagation," DTIC Document, Tech. Rep., 1985.

[3] P. Ghorbanian, A. Ghaffari, A. Jalali, and C. Nataraj, "Heart arrhythmia detection using continuous wavelet transform and principal component analysis with neural network classifier," in *Computing in Cardiology, 2010*, Sept 2010, pp. 669–672.

[4] Y. Ozbay, "Fast recognition of ecg arrhythmias," Ph.D. dissertation, Ph. D. Thesis, Institute of Natural and Applied Science, Selcuk University, 1999.

[5] G. Bortolan, I. Jekova, and I. Christov, "Comparison of four methods for premature ventricular contraction and normal beat clustering," in *Computers in Cardiology, 2005*. IEEE, 2005, pp. 921–924.

[6] M. Lichman, "UCI machine learning repository," 2013. [Online]. Available:

[7] D. Da Costa, W. J. Brady, and J. Edhouse, "Abc of clinical electrocardiography: Bradycardias and atrioventricular conduction block," *BMJ: British Medical Journal*, vol. 324, no. 7336, p. 535, 2002.

[8] C. Stergiou and D. Siganos, "Neural networks. 1996," 2010.

[9] W. W. Cohen, *Machine Learning Proceedings 1994: Proceedings of the Eighth International Conference*. Morgan Kaufmann, 2014.

[10] A. Liaw and M. Wiener, "Classification and regression by randomforest," *R news*, vol. 2, no. 3, pp. 18–22, 2002.

[11] A. Kantchelian, J. Tygar, and A. D. Joseph, "Evasion and hardening of tree ensemble classifiers," *arXiv preprint arXiv:1509.07892*, 2015.