A Novel Approach to Classify Cardiac Arrhythmia Using Different Machine Learning Techniques

Parag Jain[1], Arjun Babu C S[1], Sahana Mohandoss[1], Nidhin Anisham[1], Shivakumar Gadade[1],
Dr Srinivas A[1], Rajasekar Mohan[1]

[1] PES University, Banashankari, Bengaluru, Karnataka 560085
paragjainpes@gmail.com, arjun1300@gmail.com, sahanamohandoss@gmail.com,
nidhin.anisham@utdallas.edu, shivugadade@gmail.com, dean-engg@dsu.edu.in,
rajasekarmohan@pes.edu

Abstract. One of the major causes of deaths around the world is cardiovascular disease. Arrhythmia is one of such disease, in which the heart beats in an abnormal rhythm or rate. The detection and classification of various cardiac arrhythmias is challenging task for doctors. When this is not done accurately or not done in time, the patient's life can be at a great risk, as few arrhythmias are serious and they can even cause potentially fatal symptoms. This paper illustrates a simple yet effective solution to help doctors in the critical diagnosis of cardiac arrhythmias. The solution utilizes a variety of machine learning algorithms, in the classification of arrhythmia, using the data set obtained from the UCI machine learning repository. Implementing the solution can provide a much needed early diagnosis that proves to be critical in saving many human lives.

Keywords: Machine Learning, ECG recordings, Cardiac Arrhythmia, Ensemble Methods, Hard voting, Healthcare, Feature Selection

1 Introduction

In healthy human being the heart beats at a rate of 60 to 100 beats per minute in a periodic sinus rhythm, which is maintained by the heart's electrical system. When there are problems with this electrical system, the heart chambers will beat in a random way or the heart will beat too fast or too slow. These conditions are collectively called as cardiac arrhythmia. The history and ECG test are crucial in diagnosis of the patients suspected with arrhythmias [1]. A typical electrocardiogram (ECG) tracing consists of P wave, QRS complex and T wave which repeats in a sequence. A normal ECG tracing is shown in the Fig. 1.

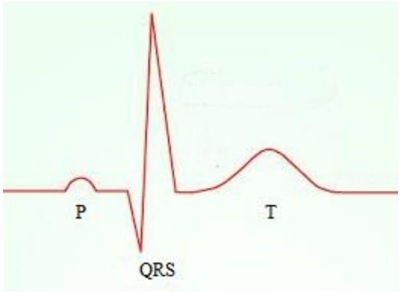

Fig. 1. Normal ECG tracing

A Cardiologist evaluate the ailments based on the various parameters like the shape, duration, amplitude, RR, PR, QT intervals, etc. of the waves [2]. Due to the massive amount of information involved, miscalculation of the beats in ECG tracing, the visual interpretation by pattern recognition of the ECG tracing is prone to errors and concluding the diagnosis to a specific type of arrhythmia becomes a difficult task. Some arrhythmias maybe just be slightly uncomfortable while few arrhythmias such as ventricular fibrillation are deadly, if no treatment is provided in time [3]. Therefore, it becomes pivotal to evaluate the exact type of arrhythmia the patient is affected with. The aim of this paper is to classify the arrhythmia dataset into one of the 16 classes by a trained machine learning system, where the first class represents absence of an arrhythmia.

2 Related Works

In the early days, arrhythmia detection was carried out using conventional statistical methods like heart rate variability (HRV) analysis [4]. Variations in the indicators of HRV, like duration of successive RR intervals and multiple derived statistical parameters such as root mean square difference and standard deviations, point to the existence of an arrhythmia [4]. The arrhythmia dataset [5] was created and classification was proposed in [6]. They developed a new supervised inductive learning algorithm, VFI5 for classification. A couple of machine learning algorithms have been investigated in the same classification problem [2]. It was found that feature selection using gradient boosting technique and the model trained with SVM, gave the best results comparatively. Principle Component Analysis (PCA) method was used for feature selection and detection of arrhythmia was done using various SVM based methods like One Against One, Decision Directed Acyclic Graph, Fuzzy Decision Function and One Against All in [7]. Cardiac arrhythmia diagnosis was carried out by techniques such as Fisher Score and Least Squares-SVM with Gaussian radial basis function and 2Dgrid search parameters in [8]. In [9], an arrhythmia prediction was accomplished by combination of methods like dimensionality reduction by PCA and clustering by Bag of Visual Words on different models, based on SVM, Random Forest (RF), kNN and Logistic Regression. The arrhythmia dataset was classified by selecting significant features using wrapper method around RF and normalizing it in [10]. Further it was used to implement several classifiers such as Multi-Layer Perceptron, NB, kNN, RF and SVM.

3 The Dataset and its Pre-processing

Dataset: We use the dataset from UCI repository [5], which contains records of 452 patients with 279 different attributes. Every record contains 4 personal details of patients like age, weight, gender and height and 275 derived attributes of the ECG waves such as amplitude, width, vector angle, so on which can be found in [5]. Also each record has the conclusion of an expert cardiologist, which represents the class of arrhythmia. Class 01 indicates a normal ECG, classes 02 – 15 indicate various types of arrhythmias while class 16 indicates the remaining unclassified ones.
Pre-processing: We remove records with abnormal values such as height of 500 cm, 780 cm, age of 0, etc. The missing values represented by '?' are replaced with the median value of that feature. The variance of the features was visualized using WEKA [11]. Further all the features with standard deviation close to zero are eliminated, as they have a very little effect on the final result. The pre-processing yields a clean dataset of 163 features and 420 records.

4 System Description

Supervised machine learning algorithms have been used for the classification problem. All of them are implemented in python. We then form different models by training each of the below algorithms with the testing dataset.

4.1 Naïve Bayes (NB)

NB is a based on the Bayes' theorem. It assumes that the value of a particular feature is independent of the value of any other feature [12]. In NB the class with the highest posterior probability, is chosen as the predicted class. Posterior probability is given as,

$$posterior\ probability = \frac{prior\ probability \times likelihood}{evidence} \tag{1}$$

where prior probability of a class is the ratio of the numbers of samples of that class to the total number of samples, evidence is the sum of the likelihoods of all classes. Before the likelihood of a class is calculated, P(A|C) i.e., conditional probability of each attribute of that class in the training sample is calculated. It is given as,

$$P(A|C) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{-(x-\mu)^2}{2\sigma^2}} . \tag{2}$$

where x is the value of that attribute, σ is the variance, μ is the mean of all the values of that attribute. Likelihood is the product of the conditional probabilities of all attributes of that class.

4.2 Decision Trees (DT)

DT is a classifier in the form of a tree structure. We implement a DT using the ID3 algorithm. In the ID3 algorithm, the attribute for splitting the data samples is decided by the information gain. The information gain which describes how well a given attribute separates the training sample into the given classes, is given as,

$$Gain(S,A) = E(S) - \sum_{v \in Values(A)} \left\{ \frac{|S_x|}{|S|} \times E(S) \right\}. \tag{3}$$

where S_x is the subset of S for which attribute A has value x and Entropy E(S) is given as,

$$E(S) = \sum_{i=1}^{c} -p_i \log_2 p_i . \tag{4}$$

where p_i is the proportion of S belonging to class i and S is the total number of samples. The data samples are split, based on the attribute with the highest information. The process continues until the entropy becomes zero [13].

4.3 k-Nearest Neighbors (kNN)

The kNN algorithm classifies the instances based on their similarity. kNN is a type of lazy learning algorithm, where all the class labels of the nearest neighbors, from the training dataset are stored and all computation are postponed until the classification [14]. The prediction class is determined based on the majority of k nearest neighbors, of the test instance. In this work a 'k' value of 3 is used. If two samples p and q have 'n' number of attributes each, then the Euclidean distance d(p,q) is given as,

$$d(p,q) = \sqrt{\sum_{i=1}^{n} (p_i - q_i)^2} . \tag{5}$$

The Euclidean distance of all the training samples are sorted and the nearest neighbors are determined based on the k.

4.4 Support Vector Machine (SVM)

SVM is an algorithm which works by creating hyperplanes, which separate the different classes in space. We use the 'one-versus-all' approach. Here one class is taken to form a hyperplane, so that it separates that class from the rest of the classes. This is done for all the classes and the test label is predicted. It does the classification based on parameters C, gamma and a specified kernel. In SVM linear classifier kernel function is a dot product of data point vectors given as,

$$K(\vec{x_i}, \vec{x_j}) = \vec{x_i}^T \vec{x_j} . \tag{6}$$

We have tried different kernels, C and gamma values and the best accuracies were obtained with radial basis function (rbf) kernel [15] for C and gamma values of 100 and 1000 respectively.

4.5 Voting Feature Interval (VFI)

Classification in each VFI algorithm is based on a majority voting of all class predictions, made by each feature. A feature makes its prediction based on the projections of all the training instances on that feature. In VFI each feature is given an equal weightage. This paper makes use of all five variations of VFI algorithm. [16]
VFI1: The algorithm constructs feature intervals for all the feature of each class. The sum of all the votes, for all the features, for each class is calculated. The class with the highest number of votes is identified as the prediction class.
VFI2: This algorithm differs from VFI1, in finding the lower bounds of the intervals. The end points are selected as the mid points, instead of the lower bounds.
VFI3: VFI3 is again a modification of VFI1, in determining the class counts. This is done so as to consider the three lower bound types of the range intervals.
VFI4: VFI4 is similar to VFI3 but if the highest and lowest points of a feature are same for a class, a point interval is constructed instead of a range interval.
VFI5: VFI5 is similar to VFI4 however it constructs point intervals for all endpoints and all values between the distinct endpoints as range intervals, excluding the endpoints.

4.6 Ensemble method

Ensemble methods takes multiple models and combines them to produce an aggregate model which performs better than any of its individual models. We use a hard voting ensemble method. It makes use of all the above algorithms to classify an unknown data record. Each algorithm predicts one of the 16 class labels. The class which is predicted the most number of times, is taken as the final predicted class. A GUI is used to display the result.

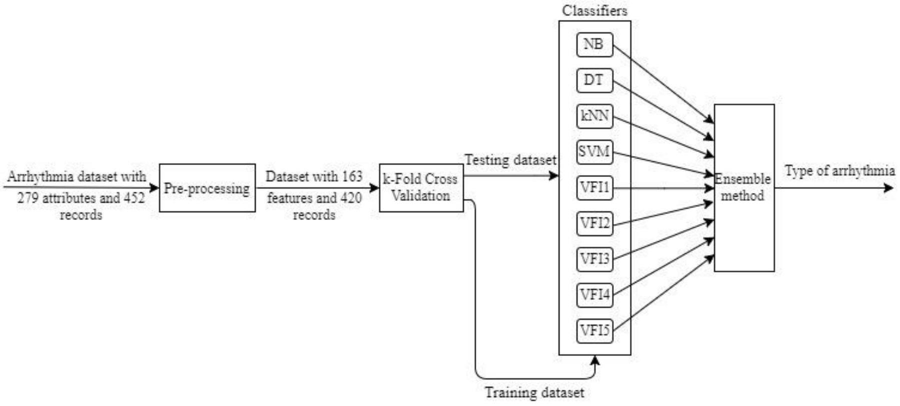

Fig. 2. Architecture design for arrhythmia classification

5 Results and Discussions

5.1 k-Fold Cross Validation

If we have a dataset that has a very low ratio of, the number of data records to the number of features, then there will be a lot of variation in the accuracy estimates for different partitions of training and testing datasets.

To mitigate this, we perform k-fold cross validation, where the original sample is randomly partitioned into k equal subsamples. One of the subsample is used as validation data for testing the model and the rest k-1 subsamples is used as training data. This cross validation is repeated k times until each of the k subsamples is used once as validation data. For a 15-fold cross validation the above models perform at its best.

5.2 Performance Analysis

We use accuracy as the performance indicator. The accuracy is calculated as number of correct predictions to number of evaluated records. The Fig. 3 shows the accuracy percentage of the various classifiers for split ratios k = 15.

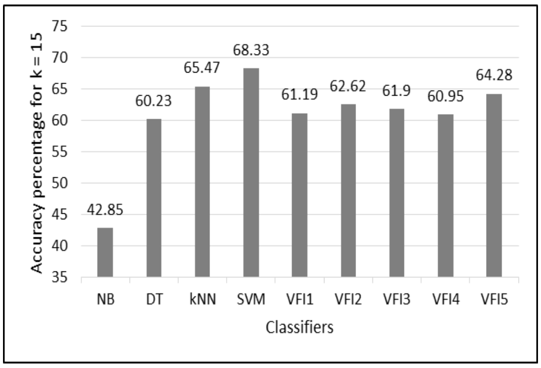

Fig. 3. Accuracy percentage of various classifiers

To summarize the figure:

- The low accuracy of the NB algorithm is due to the fact that every feature is assumed to be independent of the other and hence the interdependence of the features is not taken into account.
- The DT algorithm may overfit the data, which is called as pruning. This may cause incorrect prediction and lowers the accuracy.
- The kNN algorithm is more effective if there are more number of neighbors. Hence we can improve the accuracy further with a larger dataset.
- The SVM algorithm gave the highest accuracy of 68.33%. SVM supports different kernels which can be used to create nonlinear hyperplanes between the classes which increases the accuracy of the model.

- The VFI algorithms too consider feature independence. The accuracies were better but the training time increased as compared to NB.

5.3 Arrhythmia Classification

The hard voting ensemble method predicts with an accuracy of 90.71%. This is significant improvement in the predictions performance. Individually the model are prone to different kinds of errors like variance, noise, and bias on the data set [17]. This can result in an average performance because each individual model might over-fit, different parts of the dataset. As long as the models are reasonably diverse, informed and independent, the risk of over-fitting is reduced, as their individual mistakes are averaged out, by merging all these predictions together. The outcomes consequently tends to be substantially better. The core intuition is to develop a "strong learner" from a group of "weak learners". Ultimately this paper provides a concrete diagnosis that are highly irrefutable.

6 Conclusion

We have provided a solution to detect the presence of cardiac arrhythmia and to classify it. The approach was to pre-processing the arrhythmia dataset, get the dataset split ratio using k-fold cross validation, train various models using machine learning algorithms with the help of the training set and predict the arrhythmia class from the testing set. By pre-processing the arrhythmia dataset, issues like underfitting or overfitting have been mostly avoided. In k-fold cross validation it has been found that the best results were obtained for a k value of 15. We have trained the models using NB, DT, kNN, SVM, VFI1, VFI2, VFI3, VFI4 and VFI5 algorithms using the training set. Finally the class of arrhythmia has been predict by the majority vote of these model using the hard voting ensemble method. The paper has achieved a best in class accuracy of 90.71%, which is robust and reliable enough for the doctors to provide a crucial diagnosis. In future work, this solution can be made accessible to everyone in the form of a smartphone app, as these devices already have the necessary bio sensors and computational power. Furthermore, whilst this work has not considered the execution time, we believe that taking account of concurrent methods like multithreading and batch processing could be an effective way to reduce execution time.

References

1. Harrison T, Kasper D, Hauser S, Jameson J, Fauci A, Longo D et al. Harrison's principles of internal medicine. New York [i pozostałe]: McGraw-Hill Education; 2018.
2. Anish Batra, Vibhu Jawa. (2016) Classification of Arrhythmia Using Conjunction of Machine Learning Algorithms and ECG Diagnostic Criteria. International Journal of Biology and Biomedicine, 1, 1-7

3. Harvard Health Publishing. Cardiac Arrhythmias [Internet]. Harvard Health. Available from: https://www.health.harvard.edu/a_to_z/cardiac-arrhythmias-a-to-z

4. Task Force of the European Society of Cardiology the North American Society of Pacing Electrophysiology. 1996. Heart rate variability. Circulation 93, 5 (1996), 1043–1065

5. Guvenir A. UCI Machine Learning Repository: Arrhythmia Data Set [Internet]. Archive.ics.uci.edu. 1998. Available from: https://archive.ics.uci.edu/ml/datasets/Arrhythmia

6. H. Altay Guvenir, Burak Acar, Gulsen Demiroz, Ayhan Cekin "A Supervised Machine Learning Algorithm for Arrhythmia Analysis." Proceedings of the Computers in Cardiology Conference, Lund, Sweden, 1997

7. Kohli N, Verma N. Arrhythmia classification using SVM with selected features. International Journal of Engineering, Science and Technology. 2012;3(8):122-131.

8. Yılmaz E. An Expert System Based on Fisher Score and LS-SVM for Cardiac Arrhythmia Diagnosis. Computational and Mathematical Methods in Medicine. 2013;2013:1-6.

9. Shimpi P, Shah S, Shroff M, Godbole A. A machine learning approach for the classification of cardiac arrhythmia. International Conference on Computing Methodologies and Communication (ICCMC). 2017;:603–7.

10. Mustaqeem A, Anwar SM, Majid M, Khan AR. Wrapper method for feature selection to classify cardiac arrhythmia. 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC). 2017;: 3656-9

11. Badr W. Getting Started With Weka 3-Machine Learning on GUI [Internet]. Medium. Towards Data Science; 2019 [cited 2019Dec6]. Available from: https://towardsdatascience.com/getting-started-with-weka-3-machine-learning-on-gui-7c58ab684513

12. Joshi P. Artificial intelligence with Python. Birmingham, UK: Packt Publishing; 2017

13. W. W. Cohen, Machine Learning Proceedings 1994: Proceedings of the Eighth International Conference. Morgan Kaufmann, 2014

14. Karimifard, S., A. Ahmadian, Mohammad Khoshnevisan, and M. S. Nambakhsh. "Morphological heart arrhythmia detection using hermitian basis functions and kNN classifier." In Engineering in Medicine and Biology Society, 2006. EMBS'06. 28th Annual International Conference of the IEEE, pp. 1367-1370

15. Alexandridis, Alex, Eva Chondrodima, Nikolaos Giannopoulos, and Haralambos Sarimveis. "A Fast and Efficient Method for Training Categorical Radial Basis Function Networks." IEEE Transactions on Neural Networks and Learning Systems (2016)

16. Demiroz G. Non-Incremental Classification Learning Algorithms Based on Voting Feature Intervals [Graduate]. Bilkent University; 1997

17. DeFilippi R. Boosting, Bagging, and Stacking — Ensemble Methods with sklearn and mlens [Internet]. Medium. 2018. Available from: https://medium.com/@rrfd/boosting-bagging-and-stacking-ensemble-methods-with-sklearn-and-mlens-a455c0c982de