Predicting Risk of Cardiac Arrhythmia Using Machine Learning Algorithms

Adwiteeya Chaudhry Ayush Bhardwaj Paras Chaudhary
Indraprastha Institute of Information Technology, Delhi
adwiteeya18126, ayush18134, paras18167 @iiitd.ac.in

Abstract

At this moment, cardiovascular diseases represent the first mortality cause in humans. So, it is becoming more and more critical for machines to predict such disasters before they happen. In challenging times like these, when health is the only asset a person could wish for, we believe it would be appropriate to apply what we learn in this domain. What motivates us is the learning we could achieve while implementing the various learning models on our data. How to predict the threat of heart related diseases in real life is of great significance, both to research and application. We want to contribute to the early detection and diagnosis of cardiovascular diseases in the medical field through Machine Learning. This paper aims at understanding the applications of machine learning in the medical domain. We apply the various classical models that we are going to learn throughout this course and find how accurate these models are, and which model works best for the problem at hand.
GitHub Repository : Predicting Heart Attacks

1. Introduction

This paper aims at a better understanding and application of machine learning in medical domain. In this paper, we modify six classical models for multi-class problems such as: Logistic Regression, Naive Bayes and SVM, and then implement them to predict cardiac arrhythmia based on patients' medical records. First, we use all features provided to build the models. Afterwards, we intend to implement some feature selection methods to get only the relevant features given the large amount of features that we have and to improve the accuracy of prediction. Later, by comparing the accuracy between using all features and using features selected, we want to imply that feature selection can significantly enhance the performance of our models.

In the medical industry, machine learning algorithms can be used to diagnose some serious diseases. In this paper, we apply machine learning algorithms to predict cardiac arrhythmia based on a patient's medical record. We use the UCI Arrhythmia Data Set for both training and testing. We are provided with 452 clinical records of patients. Each record contains 279 attributes, such as age, sex, weight and information collected from ECG signals. The diagnosis of cardiac arrhythmia is divided into 16 classes. Class 1 refers to a normal case. Class 2 to 15 represent different kinds of cardiac arrhythmia, such as Ischemic Changes, Old Anterior Myocardial Infarction, Supra-ventricular Premature Contraction, Right Bundle Branch Block and etc. Class 16 refers to the rest.

Our objective is to classify a patient into one class according to his or her clinical measurements. This paper applies algorithms like Logistic Regression, Naive Bayes, and SVM algorithms to this realistic problem, compares their accuracy, and modifies the models to get the best possible results.

2. Literature Survey

2.1. Predicting cardiac Arrhythmia

This paper applied machine learning algorithms to predict cardiac arrhythmia based on a patient's medical record. They used clinical records of patients. Each record contains attributes such as age, sex, weight and information collected from ECG signals. They classified a patient into 16 different classes according to their clinical measurements. Models used were Naive Bayes, Logistic regression and SVM algorithms.

2.2. Image-Based Cardiac Diagnosis

Cardiac imaging plays an important role in the diagnosis of cardiovascular disease. In this paper, They describe the ML techniques and the procedures required to successfully design, implement, and validate new ML tools for image-based diagnosis. It tells about the risk of getting cardiovascular diseases using their cardiac images and other medical conditions.Models used were Logistic regression, Random forest, SVM, Clustering, Convolutional Neural Network.

3. Dataset Description

We are using UCI Arrhythmia Data Set for both training and testing. The Dataset contains 452 records of patients with each record containing 279 attributes like age, sex, weight other medical informations. Preprocessing and discretization The dataset contains 279 features. Out of these, 4 features had plenty of missing values. Therefore, these were removed. 1 record having missing value for Heart Rate was also removed. While analyzing the dataset, we observed that some features like the age, sex etc. had discrete values and the other ones had continuous values. One of the models used for classification is using the Multinomial Naive bayes algorithm. Therefore, data was converted into the discrete values and each feature was divided into 10 intervals.

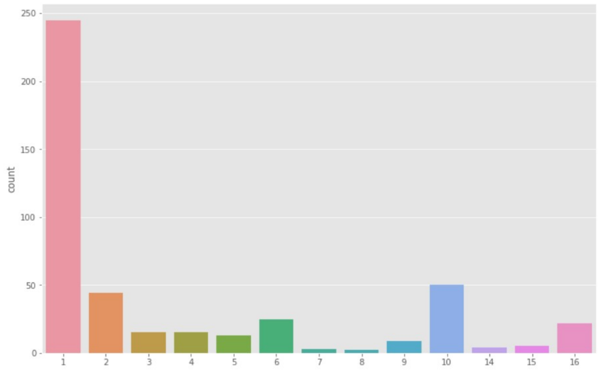

Figure 1: Class Distribution

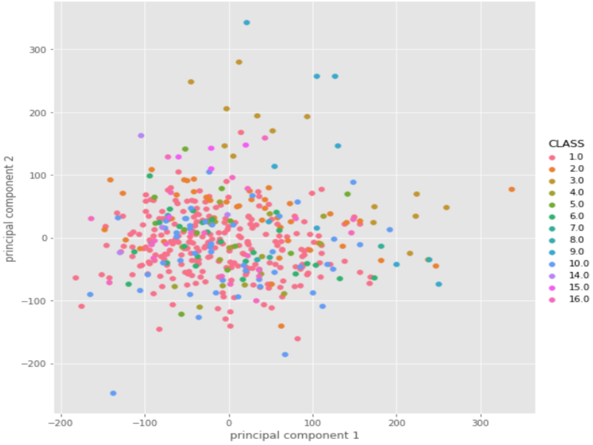

Figure 2: PCA Plot

4. Model Details

4.1. Logistic Regression

One vs all method was used to do Multiclass classification using Logistic Regression, for which 16 models of linear regression were trained and used. Multiclass classification makes the assumption that each sample is assigned to one and only one label so that helped our case where he had to classify data into 16 different classes.
Binomial logistic regression first calculates the odds of the event happening for different levels of each independent variable, and then takes its logarithm to create a continuous criterion as a transformed version of the dependent variable. The logarithm of the odds is the logit of the probability, the logit is defined as follows:

$$\operatorname{logit} p = \ln \frac{p}{1-p} \quad \text{for } 0 < p < 1\,.$$

4.2. Multinomial Naive Bayes

After data preprocessing, all the features can only take values from 1 to 10. Also, we would like to classify the training set into 16 classes. The multinomial As it takes the frequency of each class's occurrence into account, the Naive Bayes classifier is suitable for classification with discrete features. Thus, it was an obvious choice for us to implement it.
Naive Bayes is based on Bayes' theorem, where the adjective Naïve says that features in the dataset are mutually independent. The occurrence of one feature does not affect the probability of occurrence of the other feature.

To make predictions, we use Bayes formula:

$$P(y=k|x) = \frac{P(x|y=k)P(y=k)}{P(x)}$$

We predict y to be

$$argmax_k P(x|y=k)P(y=k).$$

4.3. Decision Tree

A decision tree is a flowchart-like structure in which each internal node represents a "test" on an attribute (e.g. whether a coin flip comes up heads or tails), each branch represents the outcome of the test, and each leaf node represents a class label (decision taken after computing all attributes).[2]
Decision trees are the most powerful and popular tool for classification and prediction. We used decision trees as they perform classification without requiring much computation. They are able to handle both continuous and categorical variables. Most importantly, decision trees provide a clear indication of which fields are more important for classification and which is less important

4.4. Random Forest

A random forest is a meta estimator that fits several decision trees on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting. We used Decision Trees for our Model before the Mid-sem. To add to that and improve the accuracy further, we implemented a Random Forest Classifier on our dataset.

Random forests (RF) construct many individual decision trees at training. Predictions from all trees are pooled to make the final prediction i.e. the mode of the classes for classification.

Information Gain is used for splitting the data using Entropy. It is calculated as the decrease in Entropy after the dataset is split on an attribute:

$$Gain(T, X) = Entropy(T) - Entropy(T, X)$$

Where,

- T = Target variable
- X = Feature to be split on
- Entropy(T,X) = The entropy calculated after the data is split on feature X

4.5. K - Nearest Neighbours

After pre-processing of the data, as we would like to classify the training set into 16 classes. The KNN classifier works well because for each sample it takes the most frequent class value of each class's occurrence in the neighbours (no. of neighbours predefined) of the concerned sample into account. KNN algorithm is very fast and it works even when the classes aren't linearly separable.

The accuracy of the KNN algorithm can be severely degraded by the presence of noisy or irrelevant features, or if the feature scales are not consistent with their importance. Therefore, much effort has been put into selecting or scaling features to improve classification for this project.

For multi-class KNN classification, upper bound error rate is given by:

$$R^* \ \leq \ R_{KNN} \ \leq \ R^* \left(2 - \frac{MR^*}{M-1}\right)$$

Where,

$$R_{KNN} = KNN \ error \ rate$$

M = Number of classes in the given problem

$$R^* = The \ Bayes \ error \ rate$$

4.6. Support-Vector Machines

In machine learning, support-vector machines are supervised learning models with associated learning algorithms that analyze data for classification and regression analysis.[2]

An SVM maps training examples to points in space so as to maximise the width of the gap between the two categories. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall.[2]

To perform SVM on multi-class problems, we can create a binary classifier for each class of the data. There are 2 possible conditions. Either the data point will belong to that class or it does not belong to that class. The classifier with the highest score is chosen as the output of the SVM.

5. Results and Analysis

For classification problems, accuracy is an important performance measurement of the classifier. To test the accuracy, we used a 10-fold cross-validation technique in our experiments. Since the results we obtain by using all features are unsatisfactory, we decide to implement feature selection to our models, as mentioned earlier, to improve the accuracy. We choose the forward search because of its easy implementation and good performance. For our six models, accuracies before feature selection are shown in the following table.

Model	Accuracy
Logistic Regression	62.765 %
Naive Bayes	61.029 %
Decision Tree	61.702 %
Random Forest	62.560 %
K - Nearest Neighbours	55.882 %
Support-Vector Machines	54.412 %

Table 1: Results

6. Feature Selection and Results

Feature selection is a technique for choosing those features in our data that contribute most to the target variable. In other words, we choose the best predictors for the target variable. Since the results we obtain by using all features are unsatisfactory, we decide to implement feature selection to our above-mentioned models to improve the accuracy. There are many kinds of heuristic search procedures used for feature selection such as forward search, backward search, and filter feature selection. We choose KBest search because of its easy implementation and good performance. For our six models, accuracy after feature selection and numbers of selected features are shown in the following table.

Model	Features	Accuracy
Logistic Regression	75	65.44 %
Naive Bayes	76	62.5 %
Decision Tree	40	63.97 %
Random Forest	126	71.32 %
K - Nearest Neighbours	50	66.176 %
Support-Vector Machines	20	66.176 %

Table 2: Results after Feature Selection

Advantages

- ***Reduces Over-fitting***: Less redundant data means less possibility of making decisions based on redundant data/noise.
- ***Improves Accuracy***: Less misleading data means modeling accuracy improves.
- ***Reduces Training Time***: Less data means that algorithms train faster.

7. Discussion and Future

Here, after running all our Models with and without Feature selection, we found that feature selection does help us in improving the accuracy of our Model and make better predictions.

There is certainly room for improvement. First, we can extract new features. 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 the 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 also use PCA to reduce the dimension of feature space and figure out what features are informative.

8. Conclusion

In our project, we think the challenge may lie in the lack of enough training examples(475) and the excessive amount of features(274). In this problem, only 475 training examples are available. Despite that, we believe that the algorithms implemented have decent accuracy and could predict Cardiac Arrhythmia. Indeed, there is room for improvement. First, we can extract new features. We can also use deep learning methods to generate new features. We can also use PCA to reduce the dimension of feature space and figure out what features are informative, something we look forward to doing in the future. So, far we have completed the implementation of all our select models along with Feature Selection. We have successfully achieved all the objectives we set in the beginning of this course. The individual contributions of the group are as shown in the table below.

Member	Contribution
Adwiteeya Chaudhry	1. Data Preprocessing & EDA 2. Implementation of MNB & RF 3. Feature Selection with MNB & RF
Ayush Bhardwaj	1. Data Preprocessing & EDA 2. Implementation of DT & SVM 3. Feature Selection with DT & SVM
Paras Chaudhary	1. Data Preprocessing & EDA 2. Implementation of LR & kNN 3. Feature Selection with LR & kNN

Table 3: Individual Contributions

References

[1] Guvenir, H. Altay, et al. ”A supervised machine learning algorithm for arrhythmia analysis.” Computers in Cardiology 1997. IEEE, 1997.

[2] Wikipedia. Decision tree — Wikipedia, the free encyclopedia.http://en.wikipedia.org/w/index.php?title=Decision

[3] Mishra, Binod Kumar, Prashant Lakkadwala, and Naveen Kumar Shrivastava. ”Novel Approach to Predict Cardiovascular Disease Using Incremental SVM.” Communication Systems and Network Technologies (CSNT), 2013 International Conference on. IEEE, 2013.

[4] Chen, Luyang, Q. Cao, S. Li and Xiao Ju. “Predicting Heart Attacks.” (2014).