Multiclass classification of cardiac arrhythmia from the ECG recordings

Francesca Usai
Dept. of Electrical, Computer and Biomedical Engineering, University of Pavia
via Ferrata, 6 – 27100 Pavia (Italy)

Abstract - Arrhythmias refer to abnormal irregular, too slow or too rapid heartbeats. Some types of arrhythmia do not pose an immediate danger, while others are potentially lethal, therefore an accurate and early diagnosis would be useful to save lives. The aim of this project is to apply machine learning methods to classify patients into one of the sixteen classes. The study is carried out on the dataset Cardiac Arrhythmia from the Bilkent University at UC Irvine Machine Learning Repository. This dataset contains 280 attributes, 206 of these are continuous, 73 discrete while the last attribute represents the reference class; they refer to anthropometric data, ECG parameters of 452 patients. We used two different approaches for feature selection and machine learning models including Ensemble Tree, Random Forest, Support Vector Machine and Probabilistic Neural Network. Results show that Random Forest performs better than the other classifiers, reaching an accuracy of 79.1% and the feature selection made with Genetic Algorithm proves to be a quick and accurate means of finding the best subset, in this case chosen of 26 features.

I. INTRODUCTION AND BACKGROUND

Arrhythmia is a clinical condition in which the normal frequency or regularity of the heart rhythm is lacking, or the physiological sequence of atrioventricular activation is altered. The beat may be faster (tachycardia), slower (bradycardia) or irregular (atrial fibrillation) than the average normal heart rate. There are several forms of arrhythmia, from the most harmless to the dangerous ones, some of which can even lead to death in a short time. Arrhythmias can originate from a problem of the bioelectrical system of the heart or from an insufficient supply of blood to the heart itself in some cases at the origin of an arrhythmia there are problems that do not affect the heart but other organs. For diagnosis, an arrhythmia can be detected already from the clinical examination (beats of the wrist and auscultation of the heart), but for their precise classification it is necessary to carry out an electrocardiogram and, in most cases, it may be appropriate to perform an electrocardiogram 24-hour dynamic (called cardiac Holter) that records every single heart beat and the presence of anomalies for an entire day or even for a longer time. Pacemaker cells are able to generate bioelectric potentials and this signal can spread to the surface where it is possible to record it through electrodes, so the ECG trace is the representation of the signal in the time domain. The main features of the ECG trace are: P wave, due to the depolarization of the atria, QRS complex, due to the depolarization of the ventricles and T wave, due to ventricular repolarization.

For developing an accurate diagnostic system different machine learning models have been applied in past to support medical decisions and therefore speed up the diagnosis itself. Previous studies have focused on recognizing only the presence or even the type of cardiac arrythmia; Guvenir et al. proposed the comparison of the VFI5 algorithm, Naive Bayesian classifier (NBC) and the Nearest Neighbor classifiers (NN). On the same dataset of ECG recordings, NBC and NN performed with an accuracy of 50% and 53%, respectively; whereas VFI5 achieved an accuracy of 62%. Mustaqeem et al. introduced another type of comparison adding to K-Nearest Neighbor and Naive Bayesian classifier also Random Forest (RF), Multi-layer perceptron (MLP) and Support Vector Machine (SVM) reaching precisely the latter 92% accuracy surpassing all other methods; this work compare different partitioning strategies uses a synthetic dataset obtained with an oversampling of records for a total of 2260. The aim of this paper is experiment with a new

approach to recognize four classes: Normal, Ischemia, Atrial arrythmia and Ventricular arrythmia. To do this I used models used and not previously, while as regards data preprocessing and features selection I propose different approaches.

II. METHODS

For this work I used the KNIME platform and Weka extension, which allowed me to carry out a first inspection of the dataset, the preprocessing phase up to the construction of the models. The classifier validation statistic was written in Python using the Spider environment. In the first phase, I concentrated on inspecting the dataset and preprocessing the data. First of all, I eliminated the 'Others' class because there was not enough documentation about it, then I thought about working with four classes: Normal, Ischemia, all Atrial arrhythmias and finally Ventricular arrhythmias together with Myocardial infarctions. The latter is derived from a careful research in literature in which a certain concordance between these two types of problems emerges and confirmed by a rapid vision of clustering. Secondly, I eliminated the attribute "Vector angles in degrees on front plane of J" that had 83% of the missing data, I eliminated those that had a constant value since they were not informative, I replaced the missing data with the average and finally normalized according to the Z score. As a result, we will get that the characteristics will be resized in order to have the properties of a normal distribution with zero mean and standard deviation equal to 1. If a feature has a variance of orders of magnitude greater than others, it could dominate the objective function of the algorithms and make the estimator unable to learn from other attributes. The Z-standard value is calculated as follows:

$$z = \frac{x_i - u}{\sigma}$$

where u is the average of the feature, σ is the standard deviation and xi is the value to be standardized.

Given the large dimensionality of the dataset, I then tried to eliminate very correlated attributes through the Spearman's correlation, leaving an attribute for representation even according to prior knowledge. The partition of the dataset was the following: 80% inner data and 20% outer data (test set). The inner data was in turn divided to carry out the features selection in a data set that were not those of the training, to avoid any possible overestimation and overfitting. Initially the inner data was divided into 20% and 80%, of which the first subset was used to make the features selection and the remaining part with the best subset of the previous phase was used for a 5 fold cross validation. Through this I was able to iteratively train the models in 4-fold and test on 1-fold, all to validate the best model.

I thought it necessary to carry out a features selection through two wrapper-based methods: Genetic Algorithm (GA) and Backward Features Elimination, both using Random Forests. In backward elimination, we start with all the features and remove the least significant feature at each iteration, improving model performance. We repeat it until no improvement is seen on removing the features. A genetic algorithm is a heuristic research that is inspired by the theory of natural evolution and is often used in the selection of features due to its nature as an optimizer. The natural selection process begins with the selection of the most suitable individuals from a population. They produce offspring that inherit the characteristics of their parents and will be added to the next generation. If the parents better approximate the targeted fitness function, their offspring will be better than the parents and have a better chance of surviving. This process continues to repeat itself and in the end there will be a generation with the most suitable people. The key concepts and main parameters are therefore selection of the candidates which takes place according to a certain strategy, the number of the population, the maximum number of generations, the crossover as it occurs in the chromosomes (with a probability and a certain strategy) and finally the point mutation (also with a characteristic probability).

Once the best subset of features has been chosen, the cross-validation trains and tests

iteratively 80% of the inner data, thus acting as a training and validation set. The choice of the models to train data fell on:

- SVM with RBF kernel (one-against-one and one-against-all)

Support Vector Machine, perhaps one of the most popular and talked about machine learning algorithms, is a supervised learning algorithm which can be used for classification and regression problems. SVM is based on the idea of finding a hyperplane that best separates the features into different domains. The hyperplane is a function which is used to differentiate between features. In 2-D, the function used to classify between features is a line whereas, the function used to classify the features in a 3-D is called as a plane, in higher dimension is called as a hyperplane. Gaussian RBF (Radial Basis Function) is a popular Kernel method used in SVM models for more. Applying kernel trick means just to the replace dot product of two vectors by the kernel function.

- Random Forest and Ensemble Tree

Random forest, like its name implies, consists of many individual decision trees that operate as an ensemble. Each individual tree in the random forest spits out a class prediction and the class with the most votes becomes our model's prediction. The fundamental concept behind random forest is a simple but powerful one: the wisdom of crowds. Ensemble tree classifiers can in themselves be considered random forests, I also wanted to use this model because it allows me to set other options and therefore to compare these two different algorithms based on decision trees. The selection of the information functions was made by means of the first Gini index, while the second uses the information Gain ratio.

- PNN

Probabilistic neural network (PNN) offer an alternative to the conventional back-propagation neural networks in classification problems without the need for massive forward and backward calculations, that is associated with the ordinary neural networks. In the PNN algorithm, the probability distribution function (PDF) of each class is approximated by a Parzen window and a non-parametric function. Then, using PDF of each class, the class probability of a new input data is estimated and Bayes' rule is then employed to allocate the class with highest posterior probability to new input data. This choice was made both because some of these algorithms represent the state of the art for a similar problem, and then because I consider them suitable for this data set and have no limitations of use. Through a statistical test I checked if there were significant differences between the models; this phase of validation and comparison of the classifiers allowed me to choose the one I thought was the best. I carried out the paired t-tests made specifically for cross validation. Paired t-test involves calculating the distances of the accuracy for each fold of two classifiers, calculating the average of this difference vector, building a variable t for comparison with the reference Student's t and calculating the sample standard deviation with the vector differences and the average. The test chosen is bilateral, it has a significance of 95%, $\alpha = 0.05$, the Student's t with which to make the comparison is 2.776 being the degrees of freedom 5 – 1.

$$t = \frac{d_m}{s_d / \sqrt{n}} \qquad s_d = \sqrt{\frac{\sum_{i=1}^{n} (d_i - d_m)^2}{n-1}}$$

With dm = mean of the vector differences, di = i-th difference of the accuracy of the two classifiers, n = number of folds, t = the variable t to be compared.

III. RESULTS

The results of the features selection led me to choose the following subsets:

Feature selection	**N. of features**
Genetic Algorithm	26
Backward Elimination	23

I report the results of two experiment of the 5 fold cross validation carried out in the models discussed in the Methods section, first with the subset of features derived by GA, the latter with the subset derived by Backward elimination.

Fold	**PNN**	**SVM 1-against-1**	**SVM 1-against-all**	**RF**	**ET**
1	57.14	60.71	55.36	64.28	69.64
2	63.64	69.09	67.27	69.09	70.9
3	67.27	70.9	67.27	67.27	70.9
4	65.45	69.09	63.64	69.09	72.73
5	69.09	69.09	63.64	81.82	78.18
mean	64.52	67.77	63.44	70.31	72.47

Fold	**PNN**	**SVM 1-against-1**	**SVM 1-against-all**	**RF**	**ET**
1	62.5	60.71	57.14	58.93	58.93
2	67.27	65.45	67.27	70.9	72.73
3	67.27	69.09	67.27	69.09	76.36
4	67.27	67.27	67.27	74.55	72.73
5	69.09	69.09	61.82	70.9	72.73
mean	66.68	66.32	64.15	68.87	70.69

Table. 1 Accuracies obtained from 5 fold CV, above for the FS with GA and below with the Backward strategy

Between the classifiers of the first experiment and for the second, paired t-test were made which led to the following results, $2.776 < t < -2.776$ leads to say that there are statistically significant differences. I then reported the results of the statistical test in which significant differences emerged. In the first experiment:

ET - PNN	t = 5.51
ET - SVM 1 against all	t = 3.75
SVM 1 against 1 - PNN	t = 3.67
SVM 1 against 1 – SVM 1 against all	t = 6.04

In the second experiment:

RF – SVM 1 against all	t = 3.19
ET – SVM 1 against all	t = 4.11

The performances obtained and the statistical tests have allowed me to reach the conclusion that the best classifiers are RF and ET. I then continued with testing the test set with these two models which, although very similar, were set differently. The subset I have chosen is that of the first experiment, that is the 26 features obtained by GA. The final results are shown in the following tables.

Class	**F-measure**	**Accuracy**	**Cohen's Kappa**
Atrial	0.62		
Ventricular	0.77		
Normal	0.85		
Ischemia	0.57		
Overall		0.791	0.62

Table. 2 Accuracy statistics of Random Forest model

Class	**F-measure**	**Accuracy**	**Cohen's Kappa**
Atrial	0.70		
Ventricular	0.74		
Normal	0.83		
Ischemia	0.50		
Overall		0.779	0.60

Table. 3 Accuracy statistics of Ensemble Tree model

	Atrial	Ventricular	Normal	Ischemia
Atrial	4	1	4	0
Ventricular	0	14	4	1
Normal	0	3	46	0
Ischemia	0	0	5	4

Table. 4 Matrix of confusion for Random Forest model

	Atrial	Ventricular	Normal	Ischemia
Atrial	6	0	3	0
Ventricular	1	13	5	0
Normal	1	3	45	0
Ischemia	0	0	6	3

Table. 5 Matrix of confusion for Ensemble Tree model

	Sensitivity	**Specifity**
Atrial	0.44	1.0
Ventricular	0.74	0.94
Normal	0.94	0.65
Ischemia	0.44	0.98

Table. 6 Statistics of Random Forest model

	Sensitivity	**Specifity**
Atrial	0.66	0.97
Ventricular	0.68	0.95
Normal	0.92	0.62
Ischemia	0.33	1.0

Table. 7 Statistics of Ensemble Tree model

IV. DISCUSSION

The dataset initially presented sixteen classes, some of which were poorly or not represented. I therefore decided to combine the arrhythmia classes based on the anatomical region of the onset, so for example in the "Atrial arrhythmia" class we find sinus bradycardia and tachycardia which, although both atrial, are two opposite pathological conditions. All this heterogeneity within the same class has made learning more difficult. Despite these groupings, the classes are still unbalanced as you can see in the figure 1, but I thought it was inappropriate to proceed with a synthetic oversampling of the data because the classes with which I work are not homogeneous within them, therefore interpolation practices (similar to what SMOTE does) and given augmentation would have represented data manipulation, and being clinical data I preferred not to operate in this way. Furthermore, the increase in data proves effective only for models of trees and neural networks, while SVM tend to work well with small datasets.

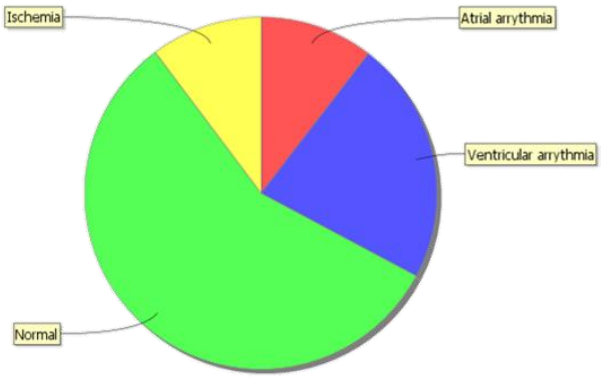

Figure. 1 Pie chart of distribution of classes

To limit the problem, I therefore used the stratified option as far as possible, therefore both in the 80% 20% partition and during the cross validation. The other critical point of the dataset was represented by the high dimensionality, which is why I proceeded first with a preprocessing and then with a feature selection; I have not opted for features extraction methods (as PCA) because being a problem in the medical-diagnostic field, it implies that there should be some dialogue with the clinical staff and I have found it inappropriate to work with new attributes with loss of their original information. Instead, I thought of taking advantage of two mechanisms and comparing them GA and Backward elimination. The former proved to be faster, and above all the best subset of 26 features includes varied attributes. The Backward option has proven to be much slower, even if it has the advantage of not having a parameterization behind it, which instead requires GA; lastly the subset from 23 that emerged from the Backward elimination is characterized by very similar attributes to each other and some clinical information as heart rate have not been included. Considering all this, my choice would fall on GA as an FS technique. Just regarding GA, another limitation emerges regarding the choice of parameters: the modification range mainly concerned the selection and crossover strategy, while all the other parameters were set by default. As for the best classifier from the CV performance analysis and from the statistics, I chose to continue with RF and ET, therefore testing the remaining 20% with these models. The results of tables 2-7 show how RF has slightly outperformed ET; the accuracy, however, is not very reliable in the presence of unbalanced classes, which is why I have also reported other measures.

V. CONCLUSION

The goal of the work was to implement a machine learning model to classify patients in one of the four classes of cardiac arrhythmias, so the ultimate goal would be to help the diagnosis by the doctor and decide if the category of patients is normal, ventricular arrhythmia, atrial arrhythmia or ischemia. The pipeline followed included a phase of data preprocessing, partition of the dataset, feature selection, cross validation and final test with the best model.

My choice is the Random Forest both for the performances obtained and because, considering the nature of the problem, it would be easier to interact with the doctor by viewing trees, rather than with other algorithms discussed above. The final accuracy is 79,1% and this is attributable to

several factors also related to the limitations of this work, but having said that, better results emerged in the literature only using data augmentation techniques. One could therefore think, besides obviously expanding the dataset by collecting more data, experimenting with different sampling techniques as well as oversampling, in the latter case only on the training set. It would also be interesting to experiment with other FS approaches, for example always as a wrapper method try with GA-SVM, even test other models or try to optimize the parameters of everything I have discussed

This model, with the supervision of an expert and the necessary changes in a future study, can represent the starting point, rather than the confirmation for a more valid and faster diagnosis.

VI. REFERENCES

[1] HA. Guvenir, B. Acar G Demiroz, A cekin, “A Supervised Machine Learning Algorithm for Arrhythmia Analysis”, Bilkent University, Bagkent University, Computers in Cardiology 1997 Vol24

[2] Anam Mustaqeem, Syed Muhammad Anwar, Muahammad Majid, “Multiclass Classification of Cardiac Arrhythmia Using Improved Feature Selection and SVM Invariants”, Hindawi Computational and Mathematical Methods in Medicine Volume 2018, Article ID 7310496, 10 pages

VII. APPENDIX

The subset of 26 features obtained via GA - RF with which the final model was trained and all the parameters set in this work are listed below :

P-R interval: Average duration between onset of P and Q waves in msec.

Vector angles in degrees on front plane of QRS

Vector angles in degrees on front plane of T

Heart rate: Number of heart beats per minute

Average width in msec. of R wave, of channel DII

Number of intrinsic deflections, of channel DIII

Average width in msec. of Q wave, of channel AVR

Number of intrinsic deflections, of channel AVR

Average width in msec. of Q wave, of channel AVL

Average width in msec. of S wave, of channel V3

Number of intrinsic deflections, of channel V3

Average width in msec. of R wave, of channel V5

Average width in msec. of S wave, of channel V5

Number of intrinsic deflections, of channel V5

Average width in msec. of S wave, of channel V6

Amplitude * 0.1 millivolt of T wave of channel DI

QRSA , sum of areas of all segments divided by 10 of channel DII

QRSTA = QRSA + 0.5 * width of T wave * 0.1 * height of T wave of channel DII

Amplitude * 0.1 millivolt of JJ wave, of channel AVR

QRSA , sum of areas of all segments divided by 10 of channel VI

Amplitude * 0.1 millivolt of R wave, of channel V2

Amplitude * 0.1 millivolt of P wave, of channel V3

QRSA , sum of areas of all segments divided by 10 of channel V4

Amplitude * 0.1 millivolt of S wave, of channel V6

Amplitude * 0.1 millivolt of T wave of channel V6

QRSA , sum of areas of all segments divided by 10 of channel V6

Genetic Algorithm parameterization:
Population size: 20
Max num. of generation: 10
Selection strategy: Roulette wheel
Crossover strategy: single point
Crossover prob: 0.8
Mutation prob: 0.01

Random Forest for FS:
limited number of levels (tree depht): 10
Number of models: 50
Split criterion: Information Gain Ratio

SVM:
Multiclass strategy: 1-against-1/1-against-all
Kernel: RBF

PNN:
Theta minus: 0.2
Theta plus: 0.4

Random Forest classifier:
Illimited number of levels (tree depht)
Number of models: 100
Split criterion: Gini Index

Ensemble Tree:
Illimited number of levels (tree depht)
Data sampling mode : with replacement, stratified
Number of models 100
Attribute sampling: sample (square root)
Attribute selection: use different set of attributes for each tree node
Fraction of data to learn single model, 1
Split criterion: Information Gain Ratio