Supervised Learning Techniques for a electrocardiogram database

1st Alejandra Ossa Yepes
dept. Mathematical Science
Universidad EAFIT
Medellin, Colombia
aossay@eafit.edu.co

***Abstract*—supervised learning is a technique to deduce a function from training data, so three types of algorithms were used for data learning, such as decision trees, Support Vector Machine, Random Forests. In the search for the best learning algorithm in a database of electrocardiograms which are classified with a variable "ecgdept" which uses the classification of the patient to whom the electrocardiogram was performed.**

***Index Terms*—Electrocardiograms, Decision Trees, Support Vector Machine, Random Forests**

I. Introduction

In this project we want to perform an supervised analysis on the data obtained in [3] which are Electrocardiograms (ECGs) provide valuable clinical information about a patient's cardiac status. Since the widespread implementation of electronic health records (EHRs), ECG records and patient data–including laboratory test results and diagnosis of disease and prescribed drug histories–have accumulated in daily clinical practice. These records are an excellent source of practice-based evidence for evaluating electrophysiological changes on ECGs under many clinical circumstances.

In previous cases there have already been studies on the non-supervision of learning machines for electrocardiograms such as Rodriguez, Gallego, Mora, Orozco and Bustamante (Rodriguez. C. A, et al., 2014), obtained results with k-means of 97,41% and 92,94% for specificity and sensibility, respectively, although sensibility is higher than the obtained with the methodology used in this work , it is pertinent to explain that the mentioned work takes into account only the ventricular contraction heartbeats. Juie D. Peshave and Rajveer Shastri (Peshave. J. D and Shastri.R, 2014), obtain similar results with the 85 % for sensibility when clustering 3 different types of arrhythmias using Thresholding's method.N.Jannah and S. Hadjiloucas (Jannah. N and S. Hadjiloucas. S, 2014) use supervised classifiers as the Support Vector Machine (MSVM) and Complex Support Vector Machine (CSVM) and obtain results in terms of 94% for accuracy, thus the supervised methods result be also useful in the process of arrhythmias identification [5].

For several years, the automatic classification of electrocardiogram (ECG) signals has received great attention from the biomedical engineering community. This is mainly due to the fact that ECG provides cardiologists with useful information about the rhythm and functioning of the heart. Therefore, its analysis represents an efficient way to detect and treat different kinds of cardiac diseases. In the literature, several methods have been proposed for the automatic classification of ECG signals. In [2], the authors implemented two classification systems based on the support vector machine (SVM) approach. The first exploits features based on high-order statistics, while the second uses the coefficients of Hermite polynomials.

For improved performance, the authors propose to combine the two classifiers by means of a weighting mechanism, whose weights are determined according to a least square estimation method. Detection of premature ventricular contractions (PVCs) by means of a fuzzy-neural network classifier with features derived from a quadratic spline wavelet transform is proposed, different classification systems based on linear discriminant classifiers are explored, together with different morphological and timing features obtained from single and multiple ECG leads.

II. Methodology

A. Dataset

ECG data of one South Korean tertiary teaching hospital with 1,103 beds. The study protocol was approved by the Ajou University Hospital Institutional Review Board. All ECGs performed from 1 June 1994 to 31 July 2013 were included in the database. There were no restrictions with respect to comorbidities or prescribed drugs. The database contained 979,273 ECGs from 461,178 patients.

The aim of this study was to establish a real-world ECG database that can be used to evaluate the effects of drugs and diseases on ECG changes, by updating and upgrading our previous ECG-ViEW database. This new database will provide an opportunity to evaluate the effects of a drug or combination of drugs, on electrophysiological changes in patients with many diseases and drug treatments.Figure 1 shows a wave segment of an electrocardiogram in which the study points of the electrocardiogram are marked [3].

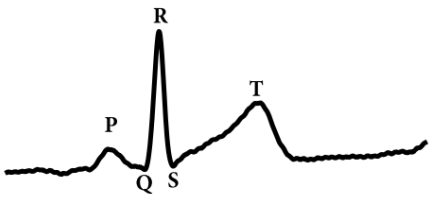

Fig. 1. Electrocardiogram segment [1]

Characteristics		Value
Patients (n)		461,178
Healthy individuals		94,326
Electrocardiogram, n		979,273
Age, years		42.6 ± 19.2
Male sex, n (%)		231,058 (50.1)
ECG parameters	*RR interval, ms*	851.7 ± 197.0
	PR interval, ms	157.1 ± 26.7
	QRS duration, ms	91.1 ± 15.2
	QT interval, ms	390.0 ± 43.5
	QTc interval, ms	425.4 ± 31.5
	P axis (degrees)	48.4 ± 24.8
	QRS axis (degrees)	45.2 ± 38.1
	T axis (degrees)	46.2 ± 38.4
Source, n (%)	*Electronic health records*	48,083 (4.9)
	ECG management system	865,590 (88.2)
	Printouts	67,384 (6.9)
Department, n (%)	*Emergency*	173,356 (17.7)
	Health examination	177,972 (18.1)
	Inpatient	193,851 (19.8)
	Outpatients	435,878 (44.4)

For computational reasons, only data from 2010 onwards will be considered in order to reduce the size of the data and speed up the process of obtaining results, leaving a total of 3,263 processed data so that the missing data are the averages of the variable to which they belong, and the variable categories were renamed as follows:

Emergency	**E**	1	ECG management system	**M**	1
Health examination	**H**	2	Scanned paper ECG	**P**	2
Outpatient	**O**	3	Electronic health records	**E**	3
Inpatient	**I**	4			

B. Decision Trees

Decision Trees are considered to be one of the most popular approaches for representing classifiers. Researchers from various disciplines such as statistics, machine learning, pattern recognition, and Data Mining have dealt with the issue of growing a decision tree from available data. This paper presents an updated survey of current methods for constructing decision tree classifiers in a top-down manner [4].

1) Information Gain: Shannon invented the concept of entropy, which measures the impurity of the input set. In physics and mathematics, entropy referred as the randomness or the impurity in the system. In information theory, it refers to the impurity in a group of examples. Information gain is the decrease in entropy. Information gain computes the difference between entropy before split and average entropy after split of the dataset based on given attribute values [6].

$$\mathrm{Info(D)} = -\sum_{i=1}^{m} pi \log_2 pi$$

Where, Pi is the probability that an arbitrary tuple in D belongs to class Ci

$$\mathrm{Info_A(D)} = \sum_{j=1}^{V} \frac{|Dj|}{|D|} \quad \mathrm{X\,Info\,(D_j)}$$

$$\mathrm{Gain(A)} = \mathrm{Info(D)} - \mathrm{Info_A(D)}$$

Where,

• $Info(D)$ is the average amount of information needed to identify the class label of a tuple in D.

•$|Dj|/|D|$ acts as the weight of the jth partition.

•$Info_A(D)$ is the expected information required to classify a tuple from D based on the partitioning by A.

The attribute A with the highest information gain, $Gain(A)$, is chosen as the splitting attribute at node $N()$.

Information gain is an impurity-based criterion that uses the entropy measure (origin from information theory) as the impurity measure (Quinlan, 1987)

2) Gain Ratio: Information gain is biased for the attribute with many outcomes. It means it prefers the attribute with a large number of distinct values. For instance, consider an attribute with a unique identifier such as customer has zero info(D) because of pure partition. This maximizes the information gain and creates useless partitioning. >> Info(D) is the average amount of information needed to identify the class label of a tuple in D

$$\mathrm{SplitInfo}_A(D) = -\sum_{j=1}^{v} \frac{|D_j|}{|D|} \times \log_2\left(\frac{|D_j|}{|D|}\right)$$

Where,

• $|Dj|/|D|$ acts as the weight of the jth partition.

• v is the number of discrete values in attribute A.

The gain ratio can be defined as

$$GainRatio(A) = \frac{\mathrm{Gain}(A)}{\mathrm{SplitInfo}_A(D)}$$

C. Support Vector Machine

Support Vector Machine is one of the classical machine learning techniques that can still help solve big data classification problems. Especially, it can help the multidomain applications in a big data environment. However, the support vector machine is mathematically complex and computationally expensive [9].

Let us first consider, for simplicity, a supervised binary classification problem. Let us assume that the training set consists of N vectors $\mathbf{x}_i \in \mathbb{R}^d (i = 1, 2, ..., N)$ from the d-dimensional feature space X. To each vector $\mathbf{x}i$, we associate a target $y_i \in \{-1, +1\}$. The linear SVM classification approach consists of looking for a separation between the two classes in X by means of an optimal hyperplane that maximizes the

separating margin. In the nonlinear case, which is the most commonly used as data are often linearly nonseparable, the two classes are first mapped with a kernel method in a higher dimensional feature space, i.e., $\Phi(X) \in \mathbb{R}^d (d' > d)$. The membership decision rule is based on the function $sign[f(x)]$, where $f(x)$ represents the discriminant function associated with the hyperplane in the transformed space and is defined as [8]

$$f(\boldsymbol{x}) = \boldsymbol{w}^* \Phi(\boldsymbol{x}) + b^*$$

The optimal hyperplane defined by the weight vector $w^* \in \mathbb{R}^d$ and the bias $b^* \in \mathbb{R}$ is the one that minimizes a cost function that expresses a combination of two criteria: margin maximization and error minimization. It is expressed as

$$\boldsymbol{\Psi}(\boldsymbol{w}, \boldsymbol{\xi}) = \frac{1}{2}\|\boldsymbol{w}\|^2 + C \sum_{i=1}^{N} \xi_i$$

This cost function minimization is subject to the following constraints:

$$y_i \left(\boldsymbol{w}\Phi\left(\boldsymbol{x}_i\right) + b\right) \geq 1 - \xi_i, \quad i = 1, 2, \ldots, N$$
$$\xi_i \geq 0, i = 1, 2, ..., N$$

where the $\xi_{'is}$ are slack variables introduced to account for nonseparable data. The constant C represents a regularization parameter that allows to control the shape of the discriminant function. The aforementioned optimization problem can be reformulated through a Lagrange functional, for which the Lagrange multipliers can be found by means of a dual optimization leading to a quadratic programming (QP) solution

$$\max_{\boldsymbol{\alpha}} \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{N} \alpha_i \alpha_j y_i y_j K\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right)$$
$$\alpha_i \geq 0, \text{for } i = 1, 2, ..., N$$
$$\sum_{i=1}^{N} \alpha_i y_i = 0$$

where $\alpha = [\alpha_1, \alpha_2, ..., \alpha_N]$ is the vector of Lagrange multipliers and $K(\cdot, \cdot)$ is a kernel function. The final result is a discriminant function conveniently expressed as a function of the data in the original (lower) dimensional feature space X.

$$f(x) = \sum_{i \in S} \alpha_i^* y_i K\left(\boldsymbol{x}_i, \boldsymbol{x}\right) + b^*$$

The set S is a subset of the indexes $\{1, 2, ..., N\}$ corresponding to the nonzero Lagrange multipliers $\alpha_{i's}$, which define the so-called SVs. The kernel $K(\cdot, \cdot)$ must satisfy the condition stated in Mercer's theorem so as to correspond to some type of inner product in the transformed (higher) dimensional feature space $\Phi(X)$. A typical example of such kernels is represented by the following Gaussian function:

$$K\left(\boldsymbol{x}_i, \boldsymbol{x}\right) = \exp\left(-\gamma \left\|\boldsymbol{x}_i - \boldsymbol{x}\right\|^2\right)$$

where γ represents a parameter inversely proportional to the width of the Gaussian kernel [8].

D. Random Forests

Random forests is a supervised learning algorithm. It can be used both for classification and regression. It is also the most flexible and easy to use algorithm. A forest is comprised of trees. It is said that the more trees it has, the more robust a forest is. Random forests creates decision trees on randomly selected data samples, gets prediction from each tree and selects the best solution by means of voting. It also provides a pretty good indicator of the feature importance [7].

Random forests has a variety of applications, such as recommendation engines, image classification and feature selection. It can be used to classify loyal loan applicants, identify fraudulent activity and predict diseases. It lies at the base of the Boruta algorithm, which selects important features in a dataset [7].

1) Advantages: • Random forests is considered as a highly accurate and robust method because of the number of decision trees participating in the process.
• It does not suffer from the overfitting problem. The main reason is that it takes the average of all the predictions, which cancels out the biases.
• The algorithm can be used in both classification and regression problems. Random forests can also handle missing values. There are two ways to handle these: using median values to replace continuous variables, and computing the proximity-weighted average of missing values.
• You can get the relative feature importance, which helps in selecting the most contributing features for the classifier.

2) Disadvantages: • Random forests is slow in generating predictions because it has multiple decision trees. Whenever it makes a prediction, all the trees in the forest have to make a prediction for the same given input and then perform voting on it. This whole process is time-consuming.
• The model is difficult to interpret compared to a decision tree, where you can easily make a decision by following the path in the tree.

III. RESULTS

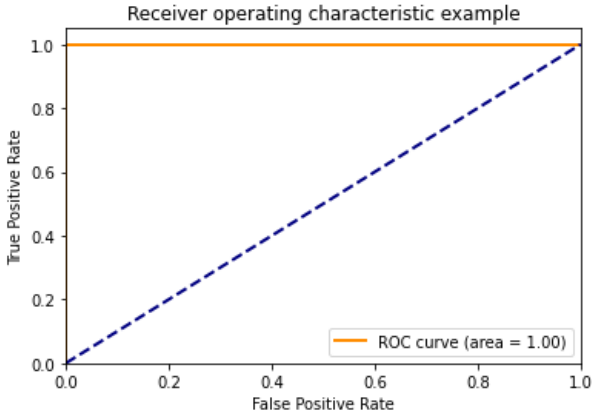

Fig. 3. Curve Roc for Decision tree

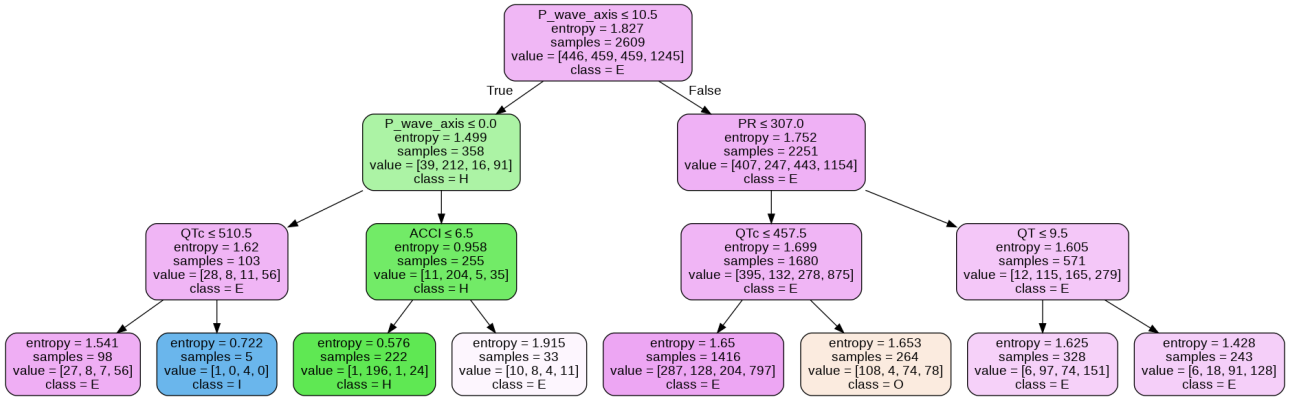

Fig. 2. Weighted decision tree

TABLE I
THE CLASSIFIER OR MODEL IN THE DECISION TREE ALGORITHM

The classifier or model can predict the type of cultivars	
Accuracy	46.554%

TABLE II
THE CLASSIFIER OR MODEL TUNING THE PARAMETERS IN THE DECISION TREE ALGORITHM

The classifier or model tuning the parameters	
Accuracy	55.283%

TABLE III
FEATURE IMPORTANCE SCORES OF RANDOM FOREST

Feature	Feature importance scores
PR	0.136626
P_wave_axis	0.135626
RR	0.134778
QRS	0.131768
QT	0.118463
QTc	0.100020
QRS_axis	0.095024
T_wave_axis	0.091498
ACCI	0.056199

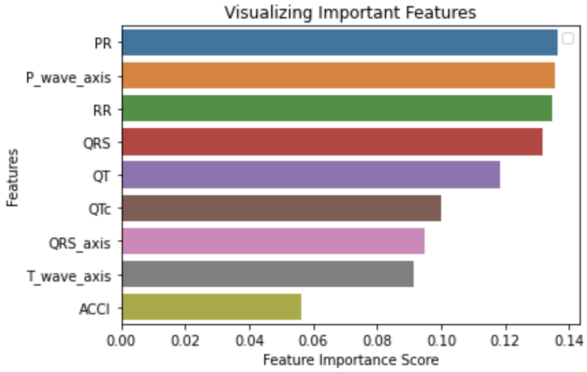

Fig. 5. Feature importance scores of Random forest

TABLE IV
ACCURACY OF MODELS FOR SUPPORT VECTOR MACHINE

SVM classification	
Linear	52.98%
Rbf	50.23%
polynomial	50.22%

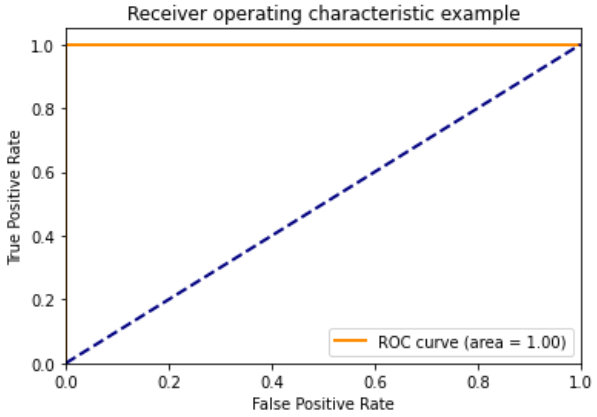

Fig. 4. Curve Roc for Random forest

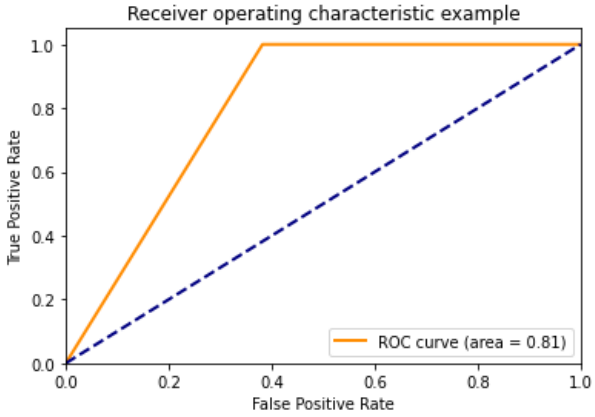

Fig. 6. Curve Roc for SVM linear

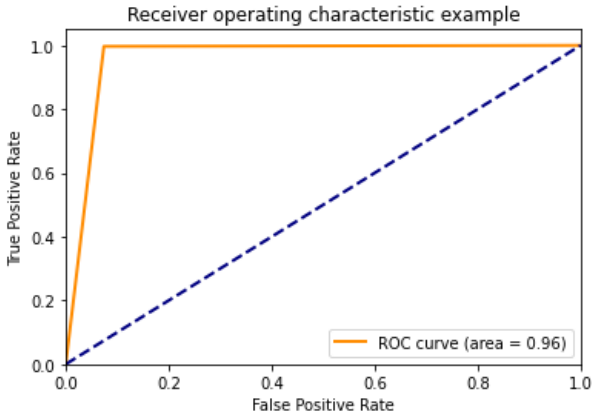

Fig. 7. Curve Roc for SVM polynomial

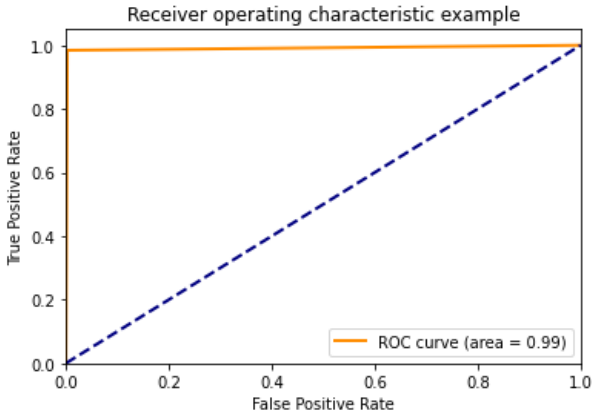

Fig. 8. Curve Roc for SVM radial

IV. Discussion

This study set out to evaluate the performance of five learning machines in classifying patients using electrocardiograms. The study was carried out to evaluate the performance of five learning machines in classifying patients using electrocardiograms. Undoubtedly, the dataset possesses a rich set of features to build suitable classifiers. This study has identified that it is possible to achieve a high-performance learner. Specifically, the results collected reveal that obtaining classifiers with accuracy rates higher than 0.80 are quite significant, as is the case of the decision trees that presents an accuracy of 0.99 as observed in Fig.3 being this a very good data learning, but still having a high computational cost, so very good values can be considered as those obtained by the random forest algorithm and the linear support vector machine, being these two algorithms more advantageous than the decision tree.

It should be noted that many of the results obtained are variable making a variation of parameters which are quite influential in the construction of the algorithm so it is suggested that performing a more thorough analysis of the parameters of each of the algorithms could generate better results as in the case of the linear support vector machines and the radial as seen in Fig.6 and Fig.7 which have the lowest classification index.

References

[1] Anup Das, Paruthi Pradhapan, W. G. P. A. R. T. R.-F. C. S. S. J. L. K. N. D.-C. V. H. Unsupervised heart-rate estimation in wearables with liquid states and a probabilistic readout. *Elsevier* (2018).
[2] Farid Melgani, Y. B. Classification of electrocardiogram signals with support vector machines and particle swarm optimization.
[3] Kim YG, Shin D, P. M. Ecg-view ii, a freely accessible electrocardiogram database.
[4] Lior Rokach, O. M. Decision tree.
[5] Monica Moreno Revelo, Sandra Patascoy Botina, A. P. B. J. R.-F. J. R.-S. S. M.-R. D. P.-O. Unsupervised analysis applied to the detection cardiac arrhythmias. *Universidad Tecnológica Equinoccial* (2017).
[6] Navlani, A. Decision tree classification in python.
[7] Navlani, A. Understanding random forests classifiers in python.
[8] Rodrigo, J. A. Support vector machines (svm) with python.
[9] Suthaharan, S. Machine learning models and algorithms for big data classification, integrated series in information systems.