Automated detection & classification of arrhythmias

Richard Tang and Saurabh Vyas
Dept. of Bioengineering, Stanford University, Stanford, CA 94305
{rhtang, smvyas}@stanford.edu

Abstract

There is a great clinical need for accurate detection and classification of cardiovascular arrhythmias. If arrhythmias are detected early enough, potential life-threatening conditions, such as heart failure, can be successfully avoided through therapies and interventions. An electrocardiogram (ECG) provides a surrogate representation of cardiac activity. Analysis of ECGs can allow for accurate classification of arrhythmias. We used the UCI Machine Learning Repository Arrhythmia dataset to compare the effectiveness of four learning algorithms for arrhythmia classification: multinomial logistic regression, support vector machines, linear discriminant analysis, and random forest (RF). In particular, we employed a grouping paradigm to isolate features that are physiologically interconnected in order to address the relative importance of each feature. Using 10-fold cross validation, we found that RF performed the best with a detection performance (AUC) of 0.864, a mean F_1 score of 0.892 ± 0.043, and an accuracy of 0.902 ± 0.037.

1 Introduction

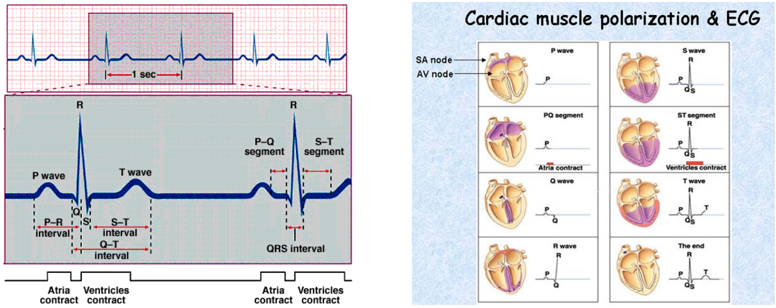
Heart failure is a leading cause of death worldwide in individuals above the age of 50. It is often deadly; mortality rates include 37% in men and 33% in women within 2 years of diagnosis [6]. If the cause of heart failure is diagnosed from arrhythmias early enough, potential therapies and interventions can often lead to good prognoses, and prevent death [5].

1.1 Prior work and current objectives

There are a number of methods to detect and classify arrhythmias from electrocardiogram (ECG) data; however, each method attempts to balance simplicity of implementation with accuracy. A good review of many current methods is provided by Acharya et al. [1]. In most machine learning implementations, the key to good performance is inherently tied to the quality of the features used. In this project, we assert that in order to successfully detect and classify the anomalous activity, we must train the classifier to learn what is important, i.e., the classifier must learn to differentiate normal physiology from abnormal physiology. Therefore, all features derived in this work are based directly on cardiovascular physiology. Furthermore, we derive five distinct groups of features from the ECG data (Sec 2.2), each with direct physiological relevance. Using our derived features, we train four different algorithms (Sec. 2.3) for the task of (a) automatic detection of arrhythmias from ECG's acquired from patients *in vivo*, and (b) classification of arrhythmias into one of four broad classes (Sec. 3.2). This is done on the University of California Irvine Machine Learning Repository Arrhythmia dataset [3].

1.2 Brief review of cardiovascular physiology

The main function of the heart is to pump blood to the rest of the body. It accomplishes this by using a complex electrical signaling cascade. When this electrical propagation cascade is modified, it leads to an arrhythmia. These can range from benign irregularities in the heart rate or rhythm, to severe disruptions that prevent the heart from contracting altogether. Arrhythmias can be detected using an electrocardiogram (ECG). An ECG provides an electrical readout of the heart's activity. This is done non-invasively by attaching a set of electrodes to the surface of the skin [8]. A notional ECG has five main features, and is termed a PQRST wave. Figure 1(a-b) show typical PQRST waves, and describe their role in the cardiac cycle [2].



(a) (b)

Figure 1: A prototypical PQRST wave of an ECG, and its relevance to the cardiac cycle and physiology. Panels (a) and (b) were taken courtesy from http://intranet.tdmu.edu.ua, and http://csmbio.csm.jmu.edu respectively.

2 Methods

2.1 Dataset and Pre-processing

The data represents 452 patients' ECG; each is represented by 279 features, including amplitudes and widths of each PQRST wave measured from 12 different channels. The data is publicly available at the University of California at Irvine Machine Learning Repository [3]. The data was preprocessed to have zero mean, and unit variance. Furthermore, the signals were de-trended (using the usual methods), and median filtered (using a 1×4 kernel). Feature extraction is discussed next.

2.2 ECG Feature Selection

Often the key step in most learning applications is deriving a robust and concise set of features. In this project, we derived a total of 129 features from the ECG data. In particular, we extracted features into one of five broad *physiological* feature "blocks": (1) 4 features concerning biographical characteristics, i.e., age, sex, height, and weight; (2) 6 features concerning average wave durations of each interval (PR interval, QRS complex, and ST intervals); (3) 5 features concerning vector angles of each wave; (4) 33 features concerning widths of each wave, measured from 12 channels; and (5) 81 features concerning amplitudes of each wave, measured from 12 channels.

Each algorithm was trained independently on each of the five feature blocks. The output probability distribution of the model was weighted for each block, and then linearly combined to create the final output model. The weights were learned using cross validation (e.g., learned RF weights are provided in Table 1). Note that this is not a majority vote scheme; actual class probabilities are computed, weighted, and then combined. Further note that we don't mix and match algorithms; this is repeated separately for each model. Figure 2 provides a visual representation of this paradigm.

2.3 Learning algorithms employed

We start our analysis by using the *Multinomial Logistic Regression* (MNLR) model. The traditional model is implemented with standard parameters. Next, we use the LIBSVM implementation of *Support Vector Machines* (SVM) [4]. In particular, we modified the method to use a Gaussian radial basis function, with $\gamma = {}^{1}\!/_{2\sigma^2}$, where the hyperparameters were learned via cross-validation. We also optimized ϵ (the tolerance of the termination criterion), and ν (the rejection ratio). Next, we wanted to systematically do dimensionality reduction, so we implement a *Linear Discriminant Analysis* (LDA)-based downsampling scheme as a preprocessing step for SVM. The typical model is implemented, and we select the eigenvalues that capture 85% of the variance (learned threshold).

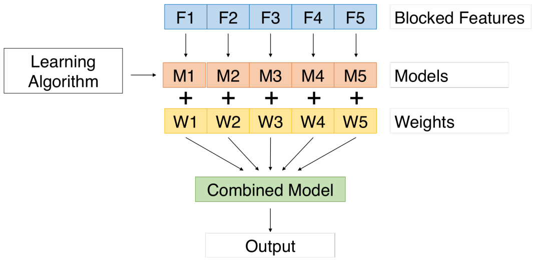


Figure 2: Features are broken into 5 blocks, independently learned, then weighted and combined.

Table 1: Optimal Weights for Features computed for the Random Forest model

Weights	**Value**
w_1 : biographical	0.05
w_2 : wave durations	0.26
w_3 : vector angles	0.03
w_4 : wave widths	0.28
w_5 : wave amplitudes	0.38

2.3.1 Random Forest

Finally, we implement a *Random Forest* ensemble classifier; the typical model is implemented as given in literature [7]. The model works by continually sampling (with replacement) a portion of the training dataset, and fitting a decision tree to it. The number of trees refer to the number of times the dataset is randomly sampled. Moreover, in each sampling iteration, a random set of features are also selected. At the end, the trees are averaged together. This is broadly known as bootstrap aggregation or bootstrapping. In decision trees, each node refers to one of the input variables, which has edges to children for all possible values that the input can take. Each leaf corresponds to a value of the class label given the values of the input variables represented by the path from the root node to the leaf node. The number of trees and the number of leaves are learned via cross-validation.

3 Results

3.1 Arrhythmia detection

The first main objective of this project was to develop a system that could robustly detect an arrhythmia. Figure 3 provides a ROC curve for the detection of arrhythmias using each of the four algorithms discussed in Section 2, and Table 2 includes the area under the curve (AUC) metric for each of the traces. Note that the RF classifier (black trace) obtains an AUC of 0.8635.

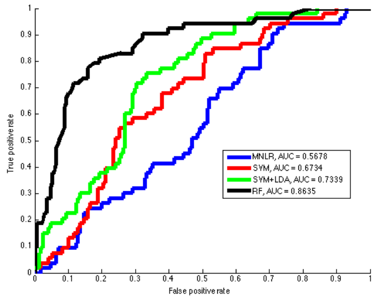


Figure 3: ROC curves for arrhythmia detection using each model. RF (black) obtains best AUC.

Table 2: Area Under Curve (AUC)

Model	**AUC**
MNLR	0.5678
SVM	0.6734
SVM+LDA	0.7339
RF	0.8635

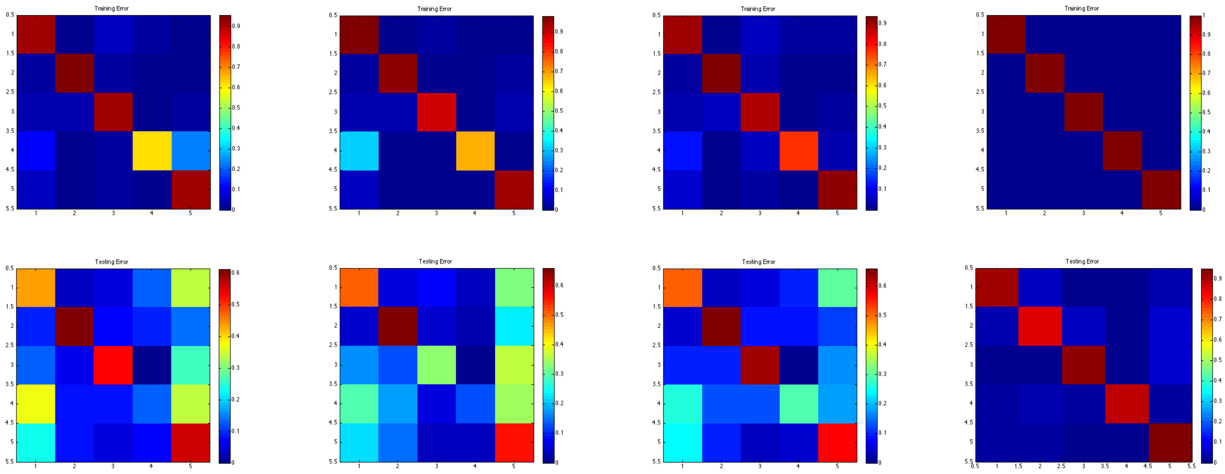
3.2 Arrhythmia classification

The second objective of this project was to develop a method to robustly classify an ECG trace into one of four broad arrhythmia class. We report our performance for each of the four methods using two different methodologies: we show confusion matrices for each algorithm, as well as compute accuracies and F_1 scores for each class. The confusion matrices for each algorithm are provided in Figure 4. Note that the axes contain labels: (1) normals, (2) coronary artery disease, (3) myocardial infarctions, (4) sinus arrhythmias, and (5) bundle branch blocks. Each column of Figure 4 corresponds to a different algorithm; from left to right these include: MNLR, SVM, SVM+LDA, RF. The first row shows confusion matrices for the training data (which will allow us to analyze bias), and the second row includes the confusion matrices for the testing data (where we analyze variance). The train/test datasets were determined using k-fold cross validation ($k = 10$).

The F_1 score and the mean accuracy for each of the four algorithms is provided in Table 3 for each of the classes. The average F_1 score is simply the mean of the F_1 scores for each class per algorithm. These results point to the RF classifier significantly outperforming the others in terms of classification. To that end, we can compute ROC curves for each of the four arrhythmia classes using the RF classier. Figure 5 contains the ROC curves for each of the four arrhythmia class. The title of each subfigure, as well as the caption of the figure, contains the AUC value.

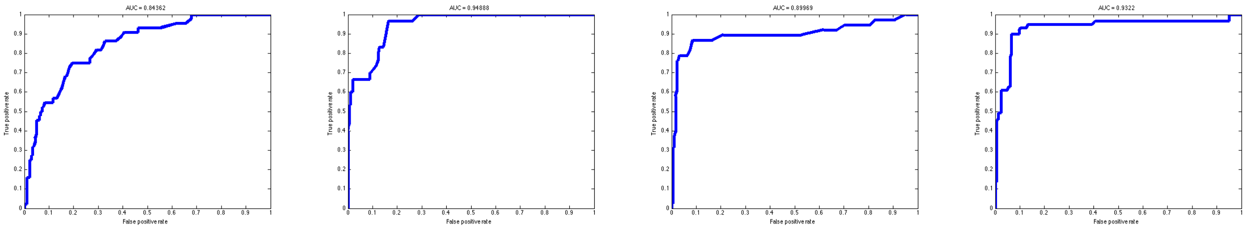
	MNLR		**SVM**		**SVM+LDA**		**RF**	
Arrhythmia Class	F_1	**Accuracy**	F_1	**Accuracy**	F_1	**Accuracy**	F_1	**Accuracy**
Normal	0.43	0.58	0.47	0.55	0.51	0.55	0.86	0.87
Coronary Artery Disease	0.61	0.68	0.64	0.76	0.66	0.69	0.84	0.86
Myocardial Infarction	0.53	0.57	0.53	0.59	0.33	0.62	0.94	0.94
Sinus Arrhythmia	0.13	0.15	0.26	0.32	0.13	0.37	0.89	0.90
Bundle Branch Blocks	0.56	0.62	0.54	0.59	0.56	0.67	0.93	0.94

Table 3: F_1 score and accuracy for each learning algorithm; best performance is colored in green.



(a) (b) (c) (d)

Figure 4: The figures in the top row are the confusion matrices for the training set, whereas the figures on the bottom are the testing set. The columns correspond to different algorithms; from left to right these are: (a) MNLR, (b) SVM, (c) SVM + LDA, (d) RF.



(a) (b) (c) (d)

Figure 5: The ROC curves for each of the four arrhythmia classes using the RF algorithm. The AUC values for the ROC curves from left to right are 0.8436, 0.9489, 0.8997, and 0.9322.

4 Discussion and Future Work

It is clear from Figure 3 and Table 2 that the RF algorithm is capable of automatically detecting arrhythmias with reliable accuracy (AUC = 0.8635). Furthermore, Table 3 and Figure 4 show that RF consistently performs the best (mean accuracy 90.2%) in terms of classification compared to the other models (see green cells in Table 3). Our general approach in this project was as follows. We started with MNLR and observed in 3(a) that it resulted in great bias in the training dataset, which could not be ameliorated with feature engineering or parameter optimization. So we turned to SVM, and in particular Gaussian RBFs operating in Hilbert spaces. While this improved bias as seen in 3(b), it was still below acceptable, and resulted in high variance and poor generalization error.

We observed that removing features (using a backwards elimination scheme) actually improved the bias. This is expected given the fact that we have 129 features (total) before using our feature selection paradigm. Therefore, we tried using LDA to automatically pick "optimal" features. While this reduced the bias in the train set, it still suffered from high variance as seen in 3(c). Ultimately, RF performed the best across the board. We posit that this has to do with its bootstrapping procedure, which reduced the overall variance of the model. Each iteration of the RF algorithm operates on a slightly different (random) training set, and a slightly different (random) set of features. This causes the trees to inherently become uncorrelated and while each tree individually has very high noise/variance, when they are all averaged together, it leads to an overall reduction of model variance. It would be interesting to see if bootstrapping can be incorporated within SVM+LDA.

We feel the biggest (and perhaps) novel aspects of this work are in the feature selection step. There are several papers that train independent models on pieces of the data, followed by a nearest-neighbor voting scheme to select the final output. We go a few steps further and (1) actually learn weights for each of the feature blocks, and (b) look at the raw probability outputs and combine them and use those to make the final decision. This argument is supported by the fact that if we merely use all the features, the algorithm perform significantly worse. Note, the algorithms also do worse if all weights are chosen to be the same. Therefore, it seems each block makes an unequal, yet unique contribution to the net result. Furthermore, our features in and of themselves are clustered in a physiologically desirable way, which makes computing weights for them more physiologically meaningful. For example, it is not surprising to learn that the amplitudes of the ECG components is more important (w = 0.38) than biographical information, such as age and gender (w = 0.05).

The real prize however lies in our follow-on work; we will attempt to probabilistically combine models. It is not clear how one can easily combine the probability outputs from different algorithms, as each are drawn from a different distribution. However, this might be a fruitful endeavor as SVM+LDA performs well in some cases (e.g., coronary artery disease), which could help the overall accuracy. Finally, a note on run-time. The models implemented here run in Matlab R2014a on a dual-core laptop with 8GB of RAM in less than one minute on the full dataset (Sec. 2.1).

References

[1] Acharya, U.R., Joseph, K.P., Kannathal, N., Lim, C.M., Suri, J.S.: Heart rate variability: a review. Medical and Biological Engineering and Computing 44(12), 1031–1051 (2006)

[2] Ashley, E.A., Niebauer, J.: Cardiology explained. Remedica (2004)

[3] Bache, K., Lichman, M.: UCI machine learning repository (2013), `http://archive.ics.uci.edu/ml`

[4] Chang, C.C., Lin, C.J.: LIBSVM: A library for support vector machines. ACM Transactions on Intelligent Systems and Technology 2, 27:1–27:27 (2011), software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`

[5] Dean, J., Lab, M.: Arrhythmia in heart failure: role of mechanically induced changes in electrophysiology. The Lancet 333(8650), 1309–1312 (1989)

[6] Kannel, W.B., Belanger, A.J.: Epidemiology of heart failure. American heart journal 121(3), 951–957 (1991)

[7] Liaw, A., Wiener, M.: Classification and regression by randomforest. R news 2(3), 18–22 (2002)

[8] Walraven, G., Walraven, G.: Basic arrhythmias. Brady Communications Company (1986)