Instituto Superior Técnico

Intelligent Systems

Project Report

A MACHINE LEARNING APPROACH TO ARRHYTHMIA DETECTION

TÉCNICO LISBOA

Masters in Mechanical Engineering

Profª Susana Vieira
Prof. Rodrigo Ventura

Group 2

Álvaro Lopes Nº96148

André Lopes Nº96351

October 24, 2023

Index

List of Figures

1 Introduction

In the context of the Intelligent Systems course, it was proposed to elaborate a project in order to apply the acquired competences and knowledge from the classes. For this, computational intelligence methods will be used in an ECG data set, in this case, the MIT-BIH Arrhythmia Database [1].

1.1 Motivation

Atrial fibrillation is the most frequent type of cardiac arrhythmia. It affects around 2.2 million people in the United States (until 2010). It has been estimated that by 2050, 6–12 million people worldwide will suffer from this condition in the US, and by 2060, 17.9 million people in Europe will be affected [2]. Since it is a health problem of high concern, it is of extreme importance to correctly diagnose the patient, so that the correct treatment can be applied.

Machine learning techniques are increasingly being used to automate tasks, doing them much more efficiently or even doing them better than humans. If correctly applied, the use of these techniques in the field of cardiology to identify and diagnose arrhythmias could be revolutionary. The authors of this project believe that technology and science should serve the human race in a positive way, and see the potential of the application of these new techniques to the health sector.

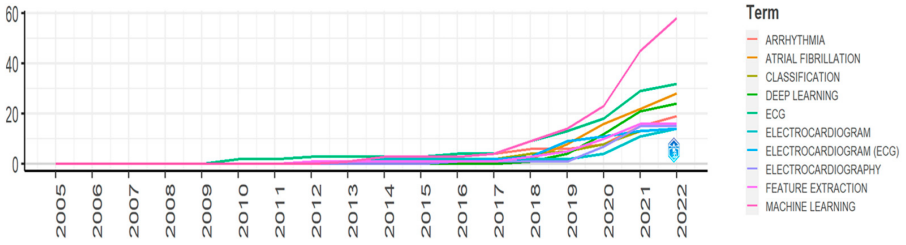

Figure 1: Growth of top 10 keywords for arrhythmia detection. From [3].

1.2 Medical overview

A heart arrhythmia is an irregular heartbeat caused by malfunctioning electrical signals that control the heart. A heart beating too fast, too slowly, or in an erratic pattern are the results of this problem. Symptoms of this condition can be a racing feeling in the chest, fast or slow heartbeat, anxiety, tiredness, chest pain, among others [4].

According to [5], there are five main types of arrhythmias: tachycardia, bradycardia, premature heartbeat, supraventricular arrhythmias and ventricular arrhythmias, among other variations. Explaining each of them is out of the scope of this project but basically they are categorized by their impact on heart rate and their point of origin in the heart. There are several options of treatment: medications, devices like pacemakers, or medical procedures, if necessary. The primary goal of the treatment is to regulate or eliminate irregular heartbeats.

The main form of diagnostic for arrhythmia is the analysis of an ECG by a medical professional. The exam involves placing sensors on the skin to detect the electrical signals generated by each heartbeat, which are recorded, and latter reviewed [6].

1.3 Data set

The data set in study is the MIT-BIH Arrhythmia Database [1]. The ECG recordings were obtained by the Arrhythmia Laboratory of Boston's Beth Israel Hospital, between 1975 and 1979. Originally, it served to stimulate competition among manufacturers of arrhythmia analyzers, since it was a way of objectively measure performance [7]. More recently, it has been used for machine learning applications.

The database consists of 48 half-hour ECG recordings obtained from 47 subjects (there are 2 recordings of the same patient). 23 of the recordings were chosen at random and 25 were purposely chosen since they incorporated uncommon but clinically important arrhythmias. The subjects included 25 men aged 32 to 89 years and 22 women aged 23 to 89 years. The original recordings were analog and were later digitized at a sample rate of 360Hz [7].

Each recording has associated a .csv file, containing the readings from the electrodes of the ECG, two for each patient. The name of the signal varies according to the place in which the electrode is placed: MLII, V1, V2, V4 and V5. Their occurrence can be seen bellow.

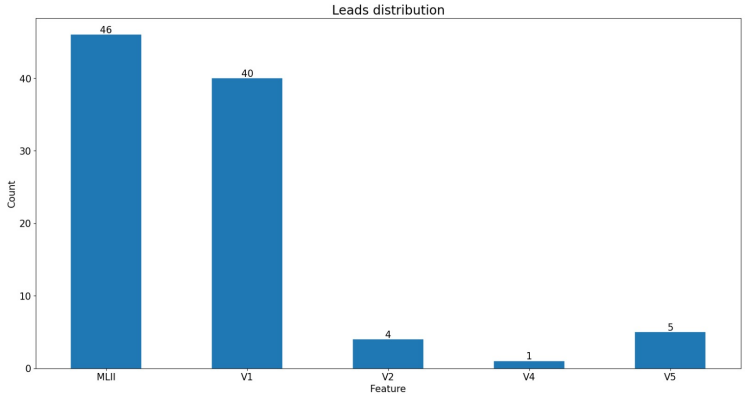

Figure 2: Signal types distribution

Each of the recordings also has a .txt file associated with it. It is in this file that are the annotations for each beat, which identify if the beat is normal of if it represents some kind of arrhythmia. The annotations and their respective meaning can be consulted in [8]. The beats were labeled one by one, by cardiologists.

1.4 Objectives

The main objective of this project will be to apply machine learning techniques to the data set, so as to able to identify if a beat is normal, or if it is some kind of arrhythmia. The type of arrhythmia is also important since different types will have different treatments. Being so, this is a multi-class classification problem.

Preprocessing the data will be of high importance, in order to train the models with the cleanest data possible. Also, defining the features to be extracted will be a key part to achieve good results. Several models should be trained and various methods used, so that the best one for this application can be identified. Their respective accuracy should be presented.

2 Methods

In this section, the practical implementation that goes from the raw data to the final models will be presented. Starting with preprocessing, passing through feature extraction and dataset balancing, and finally reaching the various final models.

2.1 Preprossessing

The first thing to do before training a model, regardless of each type it is, is to preprocess the data. This measure ensures that the models are being fed with the best data possible, hopefully leading to better results.

2.1.1 Heartbeat segmentation

The first thing to do was to see how the beats were segmented in the database. By opening the annotation files, it was possible to see that each annotation is associated with a number of samples. Being so, the interval between the sample referenced with the current annotation and the sample after the referenced sample in the previous annotation corresponds to the classification made by the cardiologists. For example, looking at the highlighted line in figure 3, it is possible to infer that samples 78-370 are of type "N", which stands for normal.

Time	Sample #	Type
0:00.050	18	+
0:00.214	77	N
0:01.028	370	N
0:01.839	662	N
0:02.628	946	N

Figure 3: Excerpt of the annotations file of patient 100

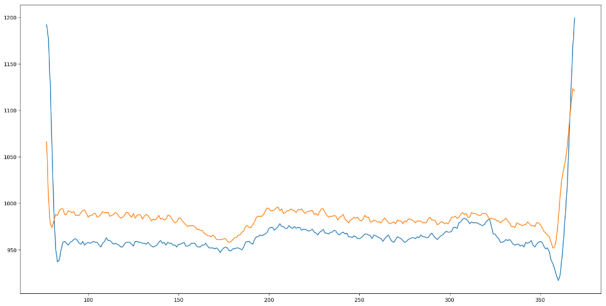

Figure 4: Samples 77-370 of MLII and V5 signals of patient 100

Analysing figure 4, we can observe that the segmentation method used by the doctors was the R-R peak interval. Basically, an ECG is a graphical representation of the heart's electrical activity, and results in identifiable wave forms. The QRS complex comprises three distinct components: the Q wave, R wave, and S wave. They represent the electrical impulse as it courses through the ventricles, signaling ventricular depolarization [9]. The R-R interval corresponds to the time elapsed between two successive R-waves.

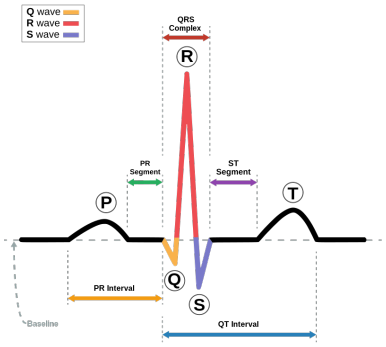

Figure 5: Schematic diagram of normal sinus rhythm for a human heart as seen on ECG. From [10].

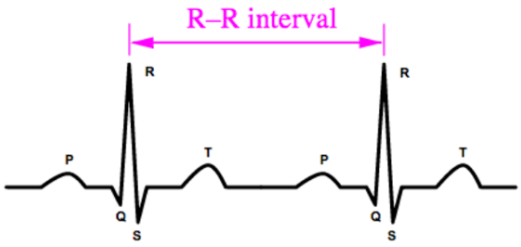

Figure 6: R-R interval. From [11]

The beginning and end of the recordings was truncated because it didn't correspond to a complete R-R peak interval. It consists of a negligible amount of samples but if maintained, unnecessary data would be used to train the model, and error would thus be induced.

2.1.2 Noise filtering

In order to further clean the data, a moving average was applied to the ECG signals. This was done because one of the methods that cardiologists use to classify the types of arrhythmia is simply by the shape of the wave. Being so, high frequencies oscillations are not useful to classify the samples and will be considered as noise.

Possible sources of noise are for example electrical interference, faulty equipment, bad electrode-skin contact, among others. Also, it is important to remember that the recordings were originally analog and were then digitized, so that's another possible source of noise. Bellow, one can see the difference between the signals before and after the noise filtering. The smoothness induced by the introduction of the moving average is clearly visible.

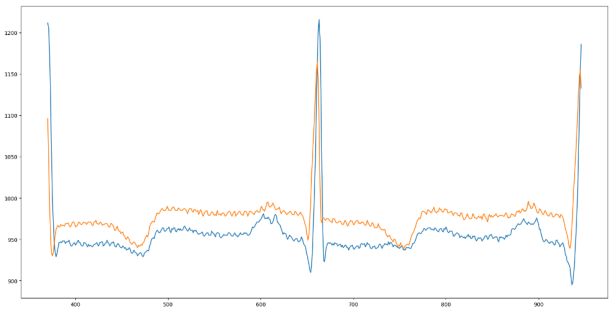

Figure 7: Example of signals before noise filtering

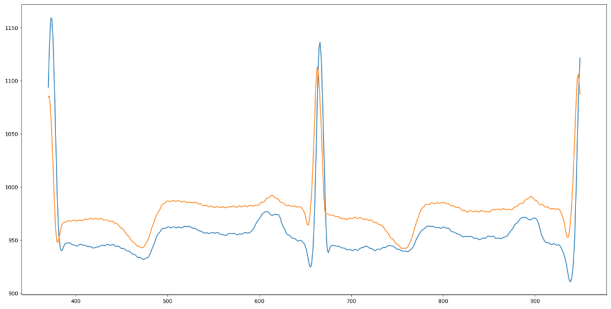

Figure 8: Example of signals after noise filtering

2.2 Feature extraction

The dataset in question is a time-series. Being so, the feature extraction requires that one obtains usefully information from the 2 signals in each R-R peak interval. Since those were classified by medical professionals, when training a model with these features, classification of each interval should be possible. The first extracted feature was the R-R peak interval, i.e. the time between consecutive R wave peaks.

The first type of features that the authors of this project decided to extract were statistical: mean standard deviation, median, mean absolute deviation, 25th percentile, 75th percentile, interquartile range, skewness and kurtosis. As for time-domain features, entropy was used, as well as the minimum and maximum value that the signal assumed in each interval in study. Those values were called valleys and peaks, respectively.

Power spectral density was also used, which is a frequency domain feature that represents the distribution of the power content of the signal frequency components, and it was calculated for each of the intervals. Since PSD itself is a signal, its entropy, the dominant frequency and the dominant magnitude were the extracted features from it. An example can be seen bellow.

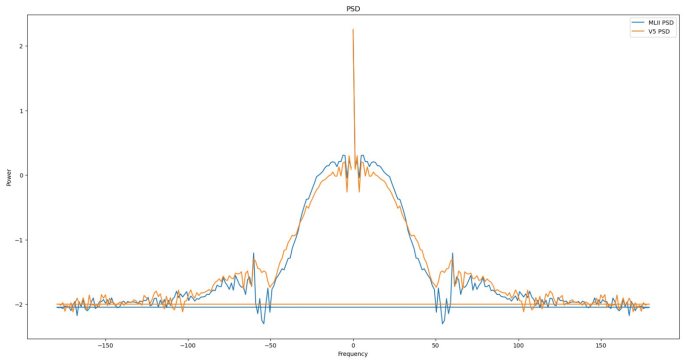

Figure 9: PSD of patient 100, samples 59-352

It is important to note that the mentioned features were extracted for each of the 2 signals present in a interval, 15 for each. With the R-R peak interval, 31 features in total were extracted from each of the intervals.

2.3 Target variable and classes aggregation

The target variable is "Type", i.e. the annotations made by the doctors in each interval. Due to the high number of different annotations, it was decided to group them, in such a way as to reduce the number of classes to predict. The AAMI convention was used to combine the beats into five classes of interest: normal beat, supraventricular ectopic beat, ventricular ectopic beat, fusion beat and unknown beat [12]. In the next page a table that summarizes this process can be consulted. Any annotation not mentioned in the table was considered an unknown beat.

AAMI class	MIT-BIH heart beat types				
Normal beat (N)	Normal beat (N)	Left bundle branch block beat (L)	Right bundle branch block beat (R)	Atrial escape beat (e)	Nodal (junctional) escape beat (j)
Supraventricular ectopic beat (S)	Atrial premature beat (A)	Aberrated atrial premature beat (a)	Nodal (junctional) premature beat (J)	Supraventricular premature beat (S)	
Ventricular ectopic beat (V)	Premature ventricular contraction (V)	Ventricular escape beat (E)			
Fusion beat (F)	Fusion of ventricular and normal beat (F)				
Unknown beat (Q)	Paced beat (/)	Fusion of paced and normal beat (f)	Unclassified beat (Q)		

Figure 10: ECG class description using AAMI standard. From [13].

The annotations were converted according to this standard and were combined in a dataframe with the respective extracted features, in order to allow model training. Also, a Min-Max scaler was used to ensure that all the values from all the extracted features are on a similar scale, in this case, -1 to 1. The values of each feature had different orders of magnitude, so there was the need to normalize the data.

2.4 Dataset balancing

After the classes aggregation, it was concluded that the dataset in question was highly imbalanced, as it is possible to ascertain by observing the table bellow.

N	Q	V	S	F
90589	8039	7236	2779	803

Table 1: Number of elements of each class in the dataset

Class imbalance can affect the accuracy of the models that will be used since it introduces bias, so it is definitely a problem that needs addressing. To solve this issue, undersampling was used: random samples of each class were used in such a way that the training set had equal number of instances of each class. SMOTE could have been used, but because the amount of artificially generated samples would have been too large, it was not considered.

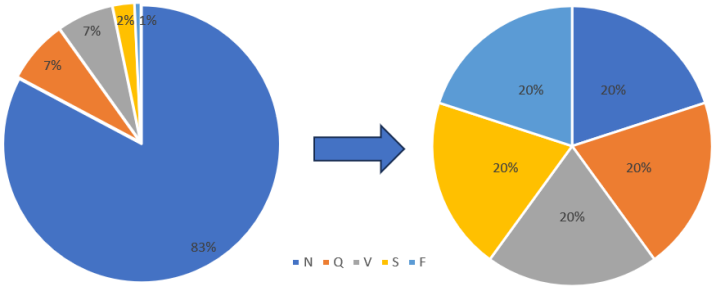

Figure 11: Classes distribution before and after undersampling

2.5 Feature selection

Feature selection was also performed before training the models. This is to ensure that no unnecessary features are used. In this case, as the signals from the dataset are somewhat similar, some features could be irrelevant and give no additional information.

Recursive feature elimination was the chosen method. The results were measured for Decision Tree, XGBoost and Random Forest models, with the latter getting the best result with 19 of the 31 features 13.

2.6 Models

2.6.1 Support Vector Machine

In simple terms, a Support Vector Machine is a model used for both classification and regression problems. It is based on creating hyperplanes that separate data into different groups. If the data has 3 inputs and one output, it can be represented in a 3D plot and the hyperplane will be a regular 2D plane on a 3D space. In the present case, a lot more than 3 inputs will be used, therefore it isn't possible to represent the groups created by the SVM in a plot. The functions that classify the data points into one group or another are called kernels and changing the used kernel can have significant implications on the obtained results.

This model was introduced to become a simple benchmark to compare the more complex models with. It could also indicate that complex models are not needed and that other simpler models could also perform well (as it will be seen in the section 3, this is not the case)

To achieve the best possible results for the SVM model, a grid search cross validation with 5 folds was used. The penalty parameter C could take one of the multiple values: 0.1, 1, 10 and 100. The analysed kernels were the linear, polynomial, radial basis function and sigmoid. As for the decision function shape, one-vs-rest and one-vs-one were used. The first treats each class as a binary classification problem against all the others, while the other trains binary classifiers for each pair of classes.

2.6.2 Neural network

Neural networks are machine learning algorithms that are based on the learning mechanisms of human brains, mimicking the way that biological neurons connect and communicate. The knowledge of the network is acquired from the input, which would be the surrounding environment in the case of humans, and the connection intensities between the neurons are called weights, which are saved, representing the acquired knowledge of the network.

In the beginning, the weights are small and random. The input is feed to the network, in the input layer, which passes the data to the rest of the network. The data is passed through the hidden layers until it reaches the output layer, which holds the calculated result or the output of the problem. An error is calculated and the weights are adjusted until reaching the desired number of iterations, which are called epochs.

The input layer was defined with 19 neurons, the number of selected features from the data. 4 fully connected hidden layers with ReLU activation were used with 128 neurons each. Along with the optimizer choice, the hyperparameters were chosen based on a grid search. The output layer has 5 neurons (since the present problem has 5 possible classes) and it uses the softmax activation function. The number of training epochs was 150, with a batch size of 32. Cross validation was once again used, using 5 folds.

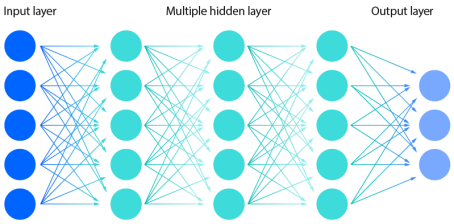

Figure 12: Deep neural network schematic. From [14].

2.6.3 Convolutional Neural Network

Although CNNs are usually used in image related tasks, they are also used for time series problems. Two different models using convolutional layers were trained on different data. The first one was trained on the dataset after the feature selection, where each interval is characterized by 19 features. The model consists of 3 pairs of 1D convolutional and pool layers, followed by a flattening layer and 2 dense layers. The structure was not iterated as this model is not usually suited for the way in which data was formatted.

The second model was trained on a different set of data. Instead of having the features for each beat, the signal values for each original sample were used. This is a much better way to take advantage of the spatial capabilities that CNNs offer. The amount of samples available using this set of data were also undersampled, mainly because of the clear class imbalance, but also because the original data would make it so that the training process would take too much time. Because now there are only 2 features, the kernel size was set to 1 in order to not lose information. Pooling was also not used to keep the dimensionality.

2.6.4 1 vs all with Neural Network models

In this method, this multi-class classification problem was converted in a binary classification one, using multiple binary classifiers. For this, neural network models were defined with the same specs as in 2.6.2, but this time, the final layer has only 1 neuron and uses the sigmoid activation function, more suited for producing a binary classification output. The objective is for each of the models to be specialized in recognizing if a sample is from one particular class or not.

In total, 5 models were used, equaling the number of classes. When training each one of them, the current class is considered as the positive class and all other classes are treated as negative. When making predictions with the test dataset, each one of the models will have its own results, and the final one will be the aggregation of all the results: the class with the highest predicted probability among all the binary classifiers.

2.6.5 Fuzzy Modeling

In the healthcare context, fuzzy modelling can be very advantageous due to its interpretability which is valued a lot in this industry.

Two models were created using fuzzy logic. The first one being a very simple application of Takagi-Sugeno rules with fuzzy c-means clustering for the membership functions. The approach adopted a one-vs-all strategy to accommodate the 5 classes. The number of clusters was set to 5 for all models as there was no significant improvement for a higher number of clusters. Due to its simplicity, this model could also be seen as a sort of benchmark for fuzzy models. Attempts were made to optimize this model using ANFIS, however they were unsuccessful.

The second model was an implementation of the ALMMo-0 classifier [15]. The code to implement this model was not self-developed. This approach automatically extracts class-specific data clouds and creates AnYa type based subclassifiers for each class. The final classification is achieved with a "winner takes all" strategy determined by the confidence scores from each subclassifier. Essentially, it is an ensemble fuzzy rule-based method to solve classification problems.

3 Results

In healthcare, correctly classifying a patient's condition as positive is much more valued than classifying correctly as negative. In this study, the same applies, and so, recall is always measured. The f1 score is also measured as it also takes into account the precision, which while not as important, is still relevant in this case. The recall and f1 score are calculated using a macro average instead of a weighted one. This was to prevent the huge weight that the majority class could have in the overall scores, resulting in a more pessimistic outcome.

Additionally, the Cohen's Kappa is also measured as it performs well on imbalanced datasets and also takes into account the random nature of making a correct prediction. The Matthews correlation coefficient is also useful as it considers all the confusion matrix elements.

The following results were taken by evaluating each model (trained on undersampled data) on a test dataset which is highly imbalanced.

	Recall	F1 Score	Cohen's Kappa	Matthews correlation coefficient
SVM	0.329	0.145	0.044	0.097
Neural Network	0.207	0.118	-0.024	-0.039
1 vs All Neural Network	0.349	0.131	0.049	0.094
CNN (features)	0.212	0.156	-0.044	-0.060
CNN (raw signals)	**0.387**	**0.368**	**0.233**	**0.242**
Takagi-Sugeno	0.170	0.050	0.023	0.050
ALMMo-0	0.286	0.200	0.035	0.045

From the table above there are a few things to point out. Firstly, the SVM model performed comparatively well as it established itself as the 3rd best performing model. This indicates that while the base neural network is more complex, it is not capturing the information as good as the SVM model. Additionally, just like the 1 vs All neural network model, which also performs comparatively well, the SVM model also uses that same strategy.

Looking at both CNN models, it is clear that it thrives on working with raw data instead of the features extracted from it, similarly to how it works with tasks involving images.

The simpler Takagi-Sugeno based fuzzy model performed very poorly. This is somewhat expected as it is a simple model, even though it also used the same 1 vs all strategy that improved the performance of neural network models. The ALMMo-0 classifier is composed of one submodel per class. It is hard to know if this approach could work better than a typical 1 vs all approach as each submodel is trained only on data from said class.

4 Conclusion

The results obtained in each of the models evaluated are rather weak. Several factors may have contributed towards that. Firstly, the data extraction could be made in a different way (extracting a range of samples around each R peak instead of R-R intervals, for example). This could have resulted in better features extracted, especially given that the state of the art methods for this problem revolve around the convolution operation and variations of it.

Instead of using standard well known statistical measures, applying wavelet transforms or further analyzing the behavior of the entire QRS complex could also help improve the results as well, the latter being one area of interest in the medical environment.

Still in data preprocessing, Generative Adversarial Networks (GANs) could have solved the problem of having too much artificially generated data. It is still artificially generated, but has shown great performance in generating new samples in recent years, particularly in image tasks.

Beyond CNNs, LSTMs could also be helpful as they can capture long-term dependencies in the data. Combining CNNs (for feature extraction) and LSTMs (to capture long-term dependencies) has proved to work well in other related work [16].

The simple fuzzy model computed in Matlab lacked results but that is to be expected as the model is very simple and the dataset very complex. The ALMMo classifier performed better, but was still beaten by the more complex neural network models. Other methods include combining both fuzzy and neural networks to achieve better results [17].

The link to the GitHub project repository is here.

References

[1] MIT-BIH Arrhythmia Database Directory

[2] Lippi G, Sanchis-Gomar F, Cervellin G. Global epidemiology of atrial fibrillation: An increasing epidemic and public health challenge. International Journal of Stroke. 2021;16(2):217-221. doi:10.1177/1747493019897870

[3] Gronthy, U.U.; Biswas, U.; Tapu, S.; Samad, M.A.; Nahid, A.-A. A Bibliometric Analysis on Arrhythmia Detection and Classification from 2005 to 2022. Diagnostics 2023, 13, 1732. https://doi.org/10.3390/diagnostics13101732

[4] Heart arrhythmia

[5] Types of heart arrhythmia

[6] Electrocardiogram (ECG)

[7] G. B. Moody and R. G. Mark, "The impact of the MIT-BIH Arrhythmia Database," in IEEE Engineering in Medicine and Biology Magazine, vol. 20, no. 3, pp. 45-50, May-June 2001, doi: 10.1109/51.932724.

[8] PhysioBank Annotations

[9] The Basics of ECG

[10] QRS complex

[11] Hartati, Sri & Setianto, Budi & Darwan, Darwan. (2017). The Feature Extraction to Determine the Wave's Peaks in the Electrocardiogram Graphic Image. International Journal of Image, Graphics and Signal Processing. 9. 1-13. 10.5815/ijigsp.2017.06.01.

[12] Das, Manab Kumar, and Samit Ari. "ECG Beats Classification Using Mixture of Features." International scholarly research notices vol. 2014 178436. 17 Sep. 2014, doi:10.1155/2014/178436

[13] Das MK, Ari S. ECG Beats Classification Using Mixture of Features. Int Sch Res Notices. 2014 Sep 17;2014:178436. doi: 10.1155/2014/178436. PMID: 27350985; PMCID: PMC4897569.

[14] Neural networks - IBM

[15] Soares, Eduardo, Plamen Angelov, and Xiaowei Gu. "Autonomous learning multiple-model zero-order classifier for heart sound classification." Applied Soft Computing 94 (2020): 106449.

[16] Shoughi, Armin, and Mohammad Bagher Dowlatshahi. "A practical system based on CNN-BLSTM network for accurate classification of ECG heartbeats of MIT-BIH imbalanced dataset." 2021 26th international computer conference, Computer Society of Iran (CSICC). IEEE, 2021.

[17] Lim, Joon S. "Finding features for real-time premature ventricular contraction detection using a fuzzy neural network system." IEEE Transactions on Neural Networks 20.3 (2009): 522-527.

5 Appendix

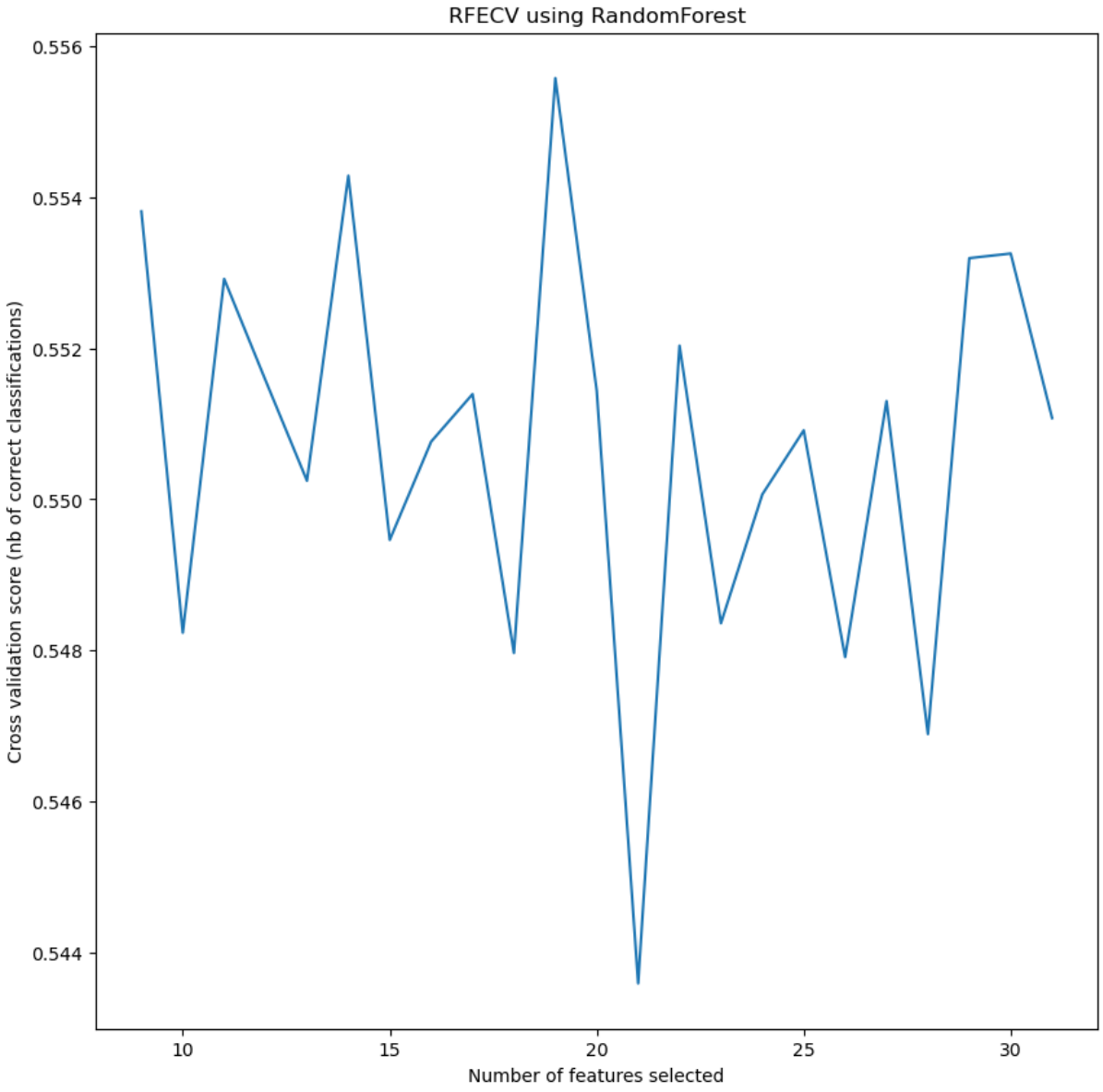

Figure 13: RFECV

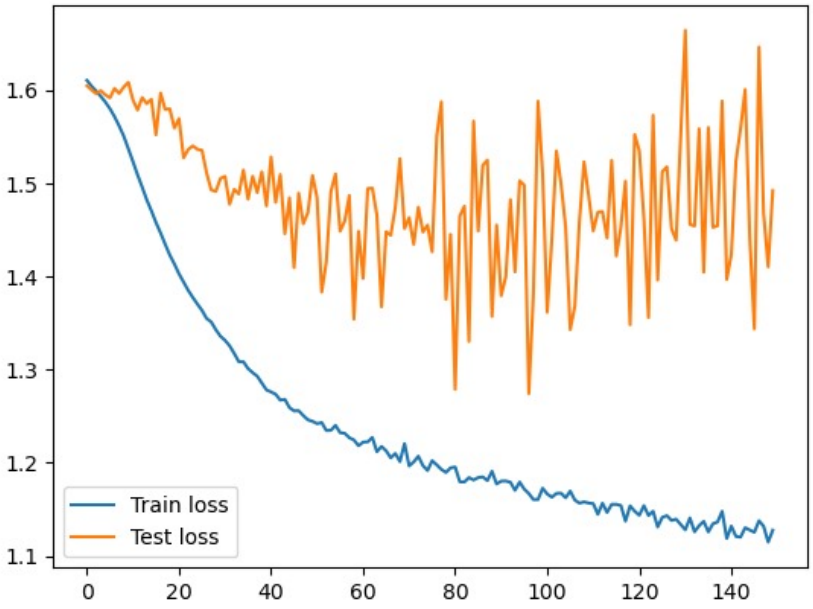

Figure 14: Training and test loss evolution for NN

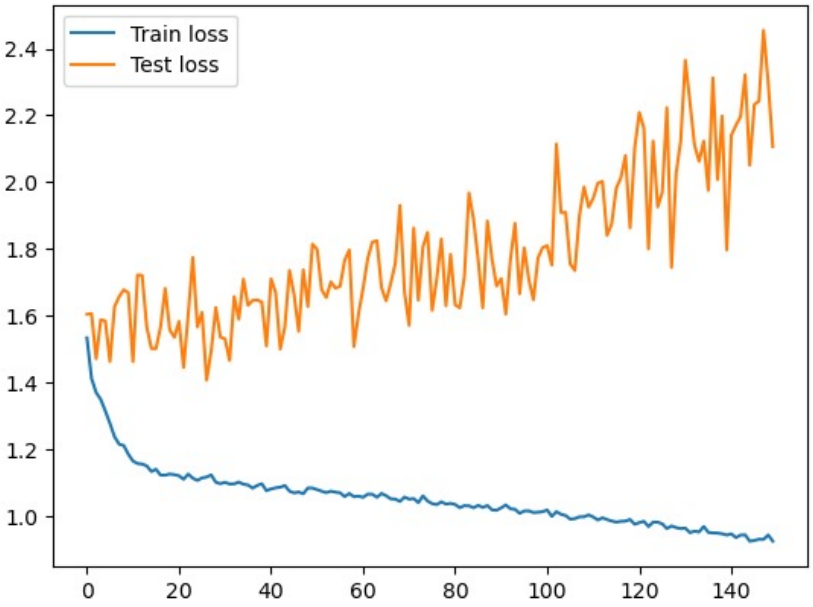

Figure 15: Training and test loss evolution for CNN

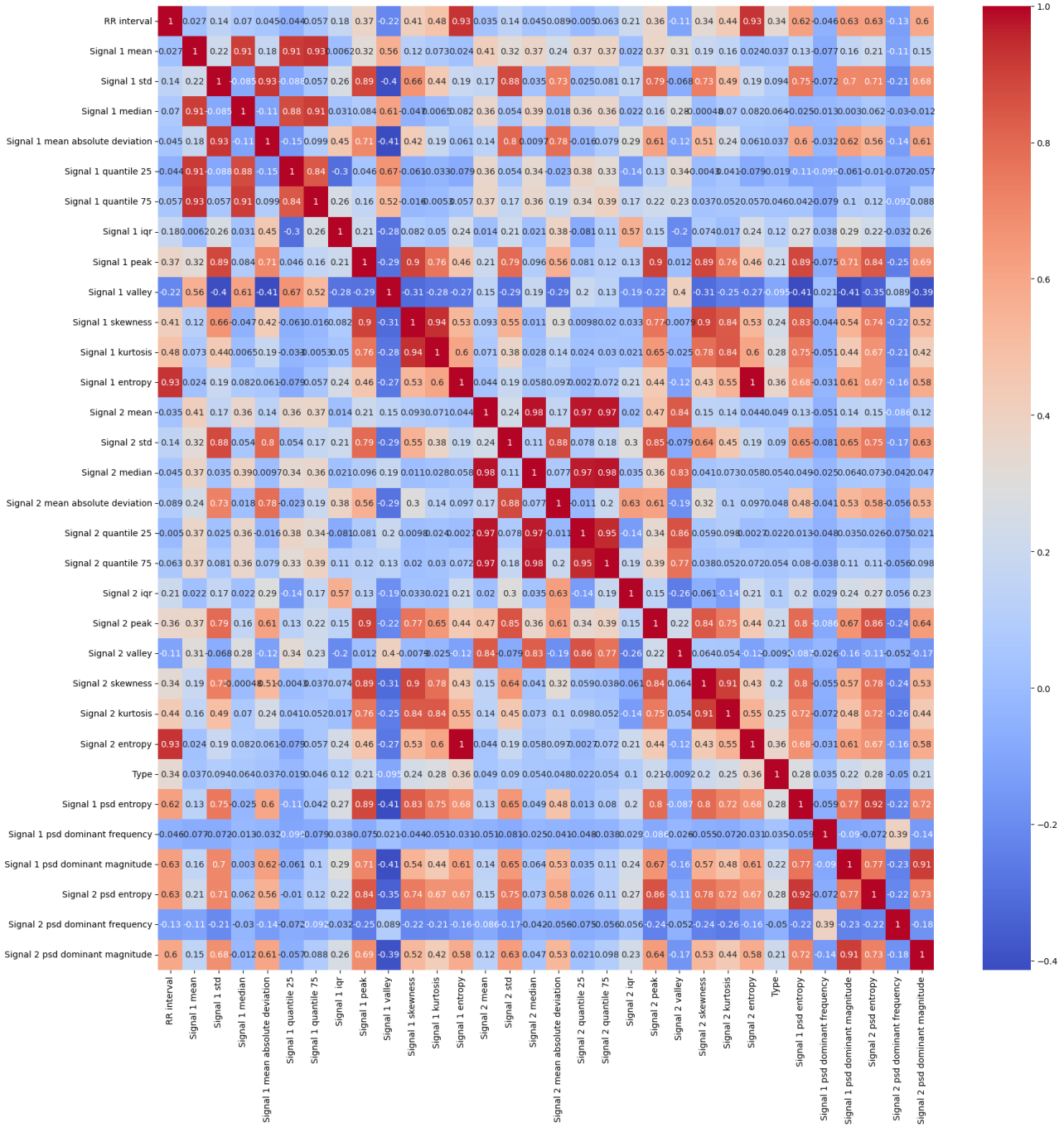

Figure 16: Correlation matrix