

Atrial Fibrillation Detection from ECG via Machine Learning Methods

Vardakas Giorgos 432
Spyrou Iro 440

University of Ioannina
Department of Computer Science and Engineering
Postgraduate Program: Data and Computer Systems Engineering
Course: Biomedical Data Analysis
Professor: Georgios Manis

June 1st 2020

1 Description of the problem

Given a labeled dataset, whose data is electrocardiographs with four types of labels, we are asked to create an automated method to classify each electrocardiogram. The labels are normal heart rate, atrial fibrillation, other type of heart rate and noisy electrocardiogram. The noisy electrocardiographs were not used in our project. The data used for this project is taken from [The PhysioNet Computing in Cardiology Challenge 2017](#).

2 Preprocessing

2.1 Class Imbalance

The classes that are used for the classification task are not balanced as shown in Table 1. To eliminate this problem, the samples of each class were reduced to the number of samples of the Atrial Fibrillation class.

Class	Number of recordings
Normal	5154
Atrial Fibrillation	771
Other rhythm	2557

Table 1: Recordings per class.

2.2 Electrocardiogram Preprocessing

Each electrocardiogram was preprocessed in two stages. An example of original data is shown in Figure 1. In the first stage of preprocessing, the data is scaled as shown in Figure 2. The scaling process helps in R peaks detection so is a crucial step. Its main function is to scale the electrocardiogram so that its lowest amplitude value is zero. The other stage of preprocessing enhances the scaled electrocardiogram so that only the R peaks are accentuated as shown in Figure 3.

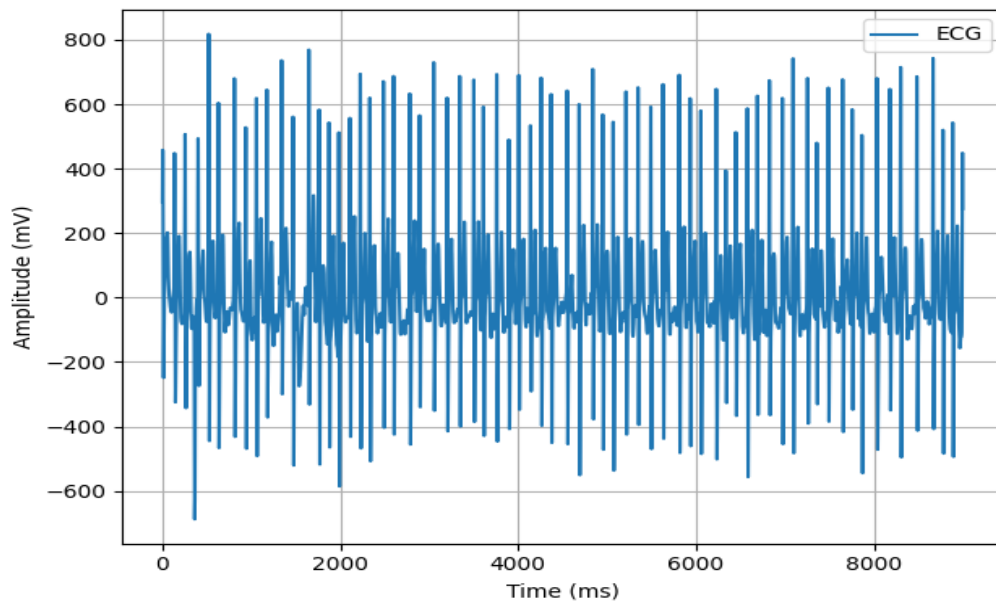


Figure 1: Original ECG before preprocessing.

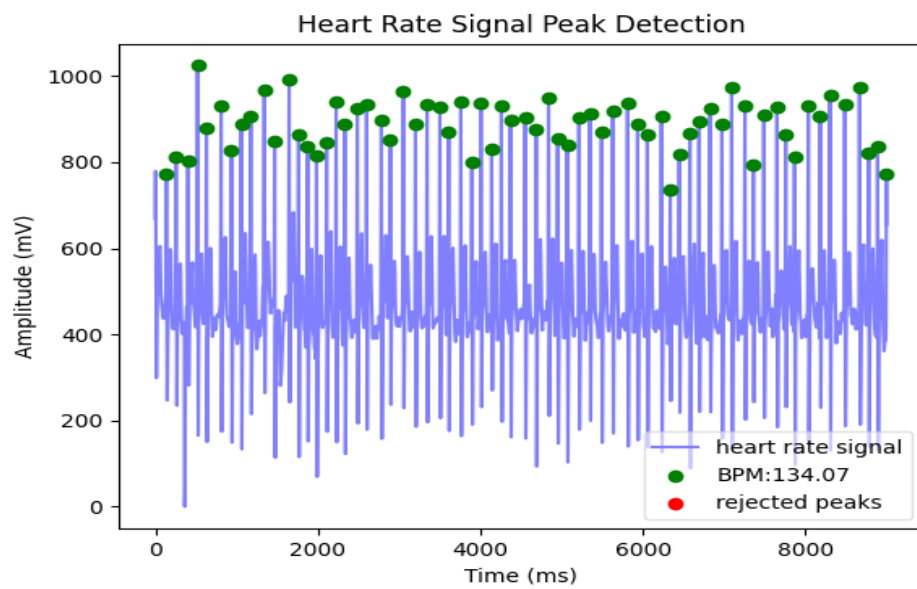


Figure 2: Scaled ECG.

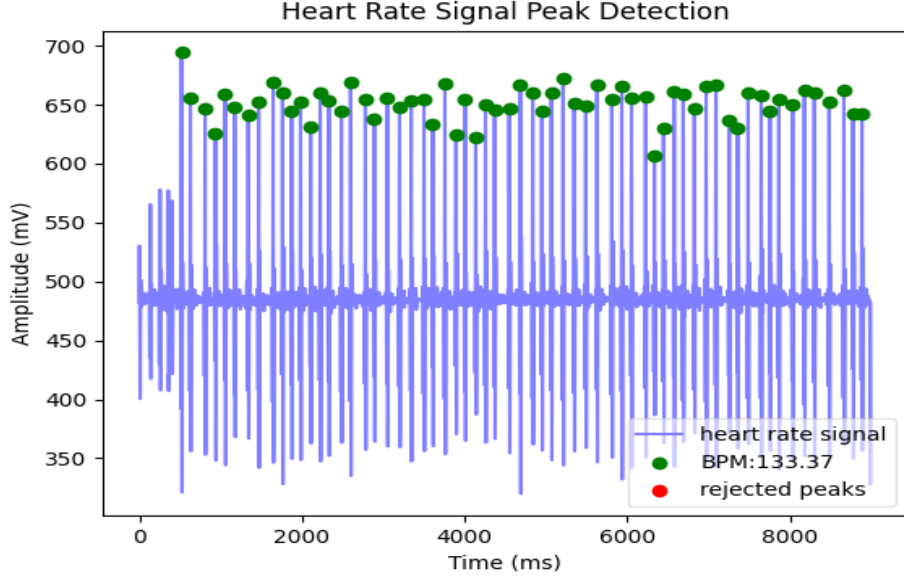


Figure 3: Enhanced ECG.

3 Processing

3.1 R Peak Detection

To detect R peaks, a moving average is calculated using a window of 1.5 seconds. Regions of interest are marked between two points of intersection (with the moving average) where the signal amplitude is larger than the moving average. R peaks are marked at the maximum of each region of interest. The R peaks are labeled with dots as shown in Figure 3.

3.2 Feature Extraction

Based on the detected R peaks the following measures are calculated:

- Beats per minute (BPM)
- Interbeat interval (IBI)
- Standard deviation of RR intervals (SDNN)
- Standard deviation of successive differences (SDSD)
- Root mean square of successive differences (RMSSD)

- Proportion of successive differences above 20ms (pNN20)
- Proportion of successive differences above 50ms (pNN50)
- Median absolute deviation of RR intervals (MAD)
- Standard deviation 1 of Poincaré plot (SD1)
- Standard deviation 2 of Poincaré plot (SD2)
- Breathing Rate (BR)

4 Classification

4.1 Classification on Original Features

For the classification task several machine learning algorithms have been used. To evaluate their performance cross validation was used on their Accuracy and F1 score. The machine learning algorithms used for the classification task are the Support Vector Machine with radial basis function kernel, the K Nearest Neighbors classifier with 3 nearest neighbors, the Naive Bayes classifier, the Random Forest classifier with gini criterion and the Multilayer Perceptron classifier with two hidden layers of sizes 100 and 50 respectively and relu activation function. The results of classification between normal heart rate and atrial fibrillation are shown in Table 2 and the results of classification between atrial fibrillation and other heart rate are shown in Table 3.

Classifier	Accuracy	F1 Score
Support Vector Machine	0.89	0.89
K Nearest Neighbors	0.91	0.91
Naive Bayes	0.93	0.92
Random Forest	0.95	0.95
Multilayer Perceptron	0.93	0.92

Table 2: Performance of Classification Between Normal or Atrial Fibrillation.

Classifier	Accuracy	F1 Score
Support Vector Machine	0.80	0.81
K Nearest Neighbors	0.81	0.82
Naive Bayes	0.81	0.80
Random Forest	0.89	0.88
Multilayer Perceptron	0.86	0.86

Table 3: Performance of Classification Between Atrial Fibrillation or Other.

4.2 Classification on Standardized Features

To improve the classification performance, the extracted features were transformed with the standardization method. The results of classification between normal heart rate and atrial fibrillation are shown in Table 4 and the results of classification between atrial fibrillation and other heart rate are shown in Table 5.

Classifier	Accuracy	F1 Score
Support Vector Machine	0.96	0.96
K Nearest Neighbors	0.96	0.96
Naive Bayes	0.94	0.94
Random Forest	0.95	0.95
Multilayer Perceptron	0.94	0.94

Table 4: Performance of Classification Between Normal or Atrial Fibrillation.

Classifier	Accuracy	F1 Score
Support Vector Machine	0.90	0.90
K Nearest Neighbors	0.88	0.88
Naive Bayes	0.83	0.83
Random Forest	0.89	0.88
Multilayer Perceptron	0.87	0.87

Table 5: Performance of Classification Between Atrial Fibrillation or Other.

5 Conclusions

Based on the experimental results the electrocardiogram contains a lot of useful information with which the electrocardiographs can be correctly classified. The extracted features seem to be powerful for this kind of task and

combined with machine learning methods the automated classification of electrocardiographs between the chosen classes has been achieved with good performance. The best results were accomplished with the Support Vector Machines in both classification tasks with Accuracy of 0.96 and F1 score of 0.96 in the first task and Accuracy of 0.90 and F1 score of 0.90 in the second task.

6 Main Libraries Used in Project

The program is written in Python 3. For the electrocardiographs preprocessing and processing the [heartpy](#) library was used. For the machine learning methods the [scikit](#) library was used. For the plots the [matplotlib](#) library was used.

7 Notes

To execute the program adjust the path to the training2017 dataset at line 178 in the project.py file. After that execute `python3 project.py` in the terminal.