EECS4643 Project: Machine learning to detect the QRS complex

Dan Diza (216270522)

Kajeney Jeyakumar (215573470)

EECS4643 - Biomedical Signal Analysis

April 1, 2024

Academic Honesty Statement

For this project we used the resources listed in the references section of this paper and discussed the project with our classmate Tony Tran.

Introduction

Signals and Systems can be used in a variety of applications to solve real-world problems. In the course Biomedical Signal Analysis, we learned about how we can use signal processing in the medical field. Specifically in lab 3 we applied the knowledge learned in lecture to identify the QRS complex using ECG data recorded in the lab. For the final project we chose to adapt machine learning to this lab and identify the QRS complex using machine learning rather than identifying the points ourselves.

Model, Methods and Library descriptions

The model chosen for this project was the Random Forest Classifier. The Random Forest Classifier allows us to work with a large dataset such as the ECG measurements and is able to classify the data between 1s and 0s using binary classification. This makes it easier for the algorithm that is written to detect the QRS complex. For the QRS detection, a method was created that takes the ecg signal and sampling frequency, which then outputs a 1D array that contains ones and zeros called labels. These labels are then plotted where the QRS complex is found in the signal. Since we needed to normalise the signal, we created another method that removes the DC component of a signal and normalises it. Originally, we tried adapting PyTorch to this project but decided to not continue with it as it is an image-based machine learning library and did not work well with the ECG datasets. Creating a neural network using PyTorch did not seem appropriate for the purpose of this project, so other models were looked at and the Random Forest Classifier was chosen.

The libraries used include Random Forest Classifier, train test split, and accuracy score. The random forest classifier library was used to take the dataset and create a model that would easily classify the dataset values between 1s and 0s to distinguish the QRS complex in the signal compared to the rest of the signal. The train test split library was used to split the datasets so that some of it could be used for training. In this case, we set it as 20% of the dataset is used for training and the remaining is used for testing. The accuracy score library was used to evaluate the accuracy of the testing.

The detection method was tested using various ECG datasets containing different conditioned subjects. The different types of heart rhythms include idioventricular rhythm, atrial fibrillation, ventricular tachycardia, and a normal sinus rhythm. This was to ensure that the detection method would work on datasets with different behaviours for accurate results every time.

Results

Sample result using a normal sinus rhythm:

```
------------------------------
Testing Completed
------------------------------
Number of Files Tested:  283
Highest Accuracy Score:  100.0 %
Lowest Accuracy Score:  79.0 %
Average Accuracy Score:  96.06713780918727 %
```

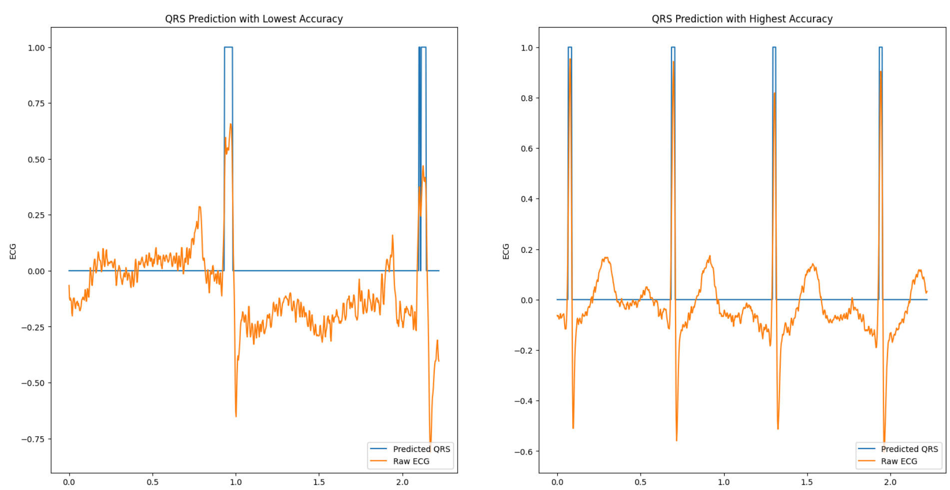

Figure 1: On the left side - Testing efficiency results; On the right side - QRS complex detection

Other tests and results are available to see in the python code file submitted along with this report.

References

1. Accuracy Score from sklearn: https://scikit-learn.org/stable/modules/generated/sklearn.metrics.accuracy_score.html
2. ECG Datasets for training and testing: https://data.mendeley.com/datasets/7dybx7wyfn/3
3. Random Forest Classifier from sklearn: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html
4. Train Test Split from sklearn: https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.train_test_split.html