Flipr Hackathon XIII

Machine Learning

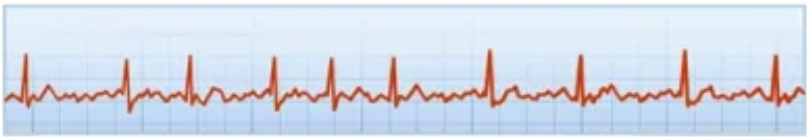
Theme : Health

Measuring Heart Rhythm Disorders

The most common way of analysing the functioning of a heart is to do an Electrocardiogram (ECG) measurement. This ECG reading is used by cardiologists to identify heart rhythm disorders and other abnormalities. For example, ECGs are commonly used to detect the presence of arrhythmias in patients. If present and not treated in time, these arrhythmias can lead to stroke and possibly heart failure.

In this task, we focus on a very common arrhythmia called Atrial Fibrillation (AF) that occurs in 1-2% of the population (5-10% of the population older than 60 years). It's a type of irregular heartbeat that can lead to blood clots, stroke, heart failure and other heart-related complications. About 15–20% of people who have strokes have AF. An example ECG reading with AF is shown below.

ECG tracing of a normal heart rhythm.



In atrial fibrillation, the tracing shows tiny, irregular "fibrillation" waves between heartbeats. The rhythm is irregular and erratic.

Problem Statement

In this test, you are provided with ECG data that is classified into two parts: Normal and AF. Your objective is to learn a model that classifies the ECG recording correctly into one of the two categories. Smartwatches such as an Apple watch also use a similar algorithm to detect AF in ECG data measured through the watch.

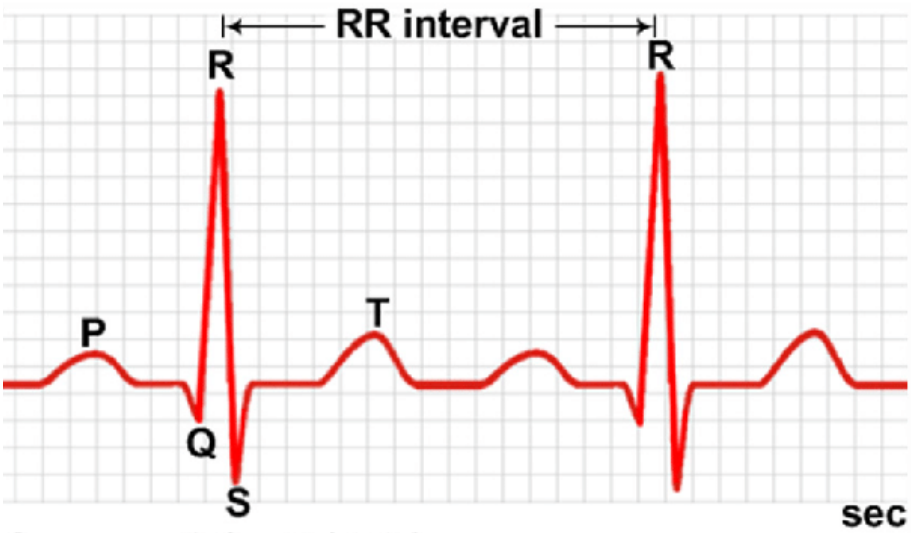
The training data is provided in the file ECG_training.csv. The ECG data is under the column 'ECG' and the corresponding label is under the column 'Classification'.

ECG recordings were sampled at 300 Hz and there are a total of 3500 readings. A classification of 'N' implies the reading is normal, and a classification of 'A' implies that the reading contains an AF episode. The testing dataset is not available to you and will be used to evaluate the model you have trained.

The problem statement is divided into two parts. In the first part, you will extract relevant features from the ECG dataset (such as the heart rate). In the second part, you will use extracted features from Part 1 to train a model that classifies the ECG reading as Normal or AF.

Part 1: Identify the relevant ECG features

An ECG reading has many features. For example, the RR interval as shown below is the interval between two successive peaks in the ECG recording (aka the time interval between two heartbeats). The statistics of the RR intervals such as its average, standard deviation, RMSSD, etc. are important parameters in determining the fitness of the heart.



Similar to this, there are many other features of the ECG data both in the time and frequency domain. Reference [1] provides a list of existing papers that identify features from the heart rhythm data and apply ML to detect AFib (feel free to implement any such existing idea).

Further, there are existing software packages that can calculate most of the required features from an ECG reading, e.g., Heartpy, which calculates features of a PPG reading through the following API:

```
import heartpy as hp
working_data, features = hp.process(ecg_reading, sample_rate=300.0, calc_freq=True)
```

The variable 'features' is a python dictionary that contains >20 ECG features which can be later used to train a model that classifies an AF episode. However, it is important to note that out-of-the-box libraries might not be the best-performing but can be a good starting point. You are free to use any method to calculate the features as long as it results in a good quality model.

Part 2: Train a model to classify an ECG with AF

It's time to wear your Machine Learning hat because, in this part, you will use the features calculated from part 1 along with the ECG label ('N' or 'A') provided in the training data to train an ML model to classify AF. Note that not all features are important, and if the dataset is small (such as in this problem), feature engineering becomes an important part of solving an ML problem where we find and use only the relevant features required to train a good model.

Feel free to implement any of the ML algorithms from [1], or any other ML method of your choice.

Evaluation: Your evaluation script should take a CSV file named ECG_testing.csv (that contains the test data with ECG and its classification as two columns) as an input and uses the model that you have trained to calculate and output its F1-score. Your solution will be evaluated based on the F1-score and the methodology used. Partial marks will be awarded for good solutions to Part 1 (feature extraction).

References

[1] Photoplethysmography-based Atrial Fibrillation Detection: A Review

Competition Rules

- There should only be **one submission per participant**
- Privately sharing of code is not permitted. In case of plagiarism, the participant shall be disqualified
- Those attempting both the parts should send 2 separate files as submissions.
- The **solution_sheet** should also be attached along with the results
- Share all your files in this Google form link: https://forms.gle/j2CEmLTNvBQntGqF6