Artificial Intelligence 201: Starter Submission Spring 2025

David Genis, Matthew Flores, Liam Cummings
David_Genis@student.uml.edu
Matthew_Flores@student.uml.edu
Liam_Cummings@student.uml.edu

Project Title

Diagnosing Arrhythmia with ECG Data using AI Models

Completed Tasks

- Researched common arrhythmia types and ECG signal characteristics
- Downloaded Dataset and preprocessed ECG data

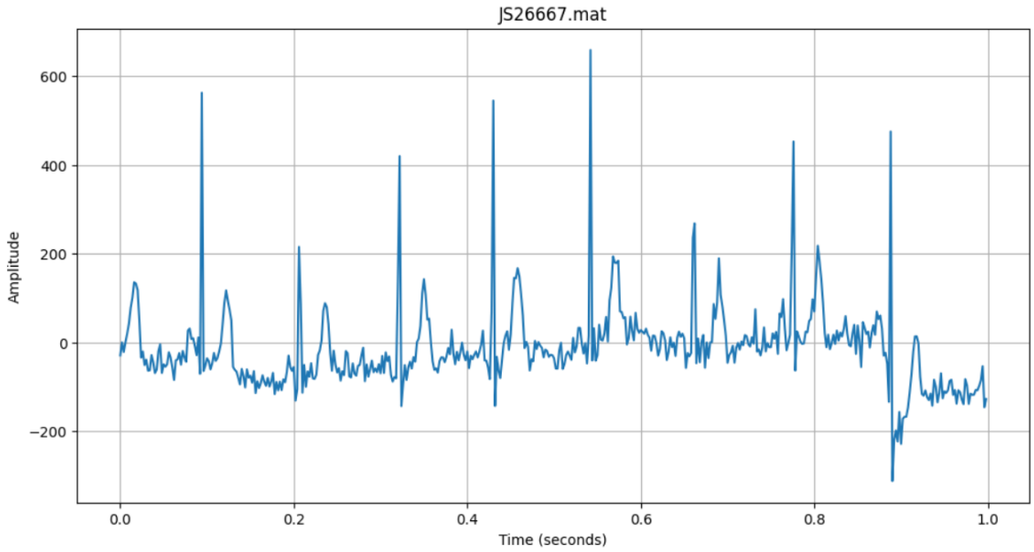

Figure 1: Sample of a Processed ECG signal

0.1 Report

The main goal of this project is to build a basic ECG arrhythmia detetection model. We are using a ECG dataset we pulled from PhysioNet. However before we can do that properly with the provided data our first task at hand was to process/filter the signals, through Identifying QRS Complexs (R-Peaks) and the corresponding features such as RR Intervals.

To begin, we installed and configured the following Python packages:

- `numpy` – Various mathematical operations and processing
- `scipy` – filtering and reading '.mat' files
- `matplotlib` – Signal Visualization
- `py-ecg-detectors` - Useful algorithims to aid in identifying QRS Complexes and RR Intervals

0.2 Milestones

We implemented code that recursively processes '.mat' ECG files. It does so by applying a filter to the signal and then downsmapling it to reduce the overall noise/complexity. Then it visualizes the processed waveform and detects R-Peaks with a Pan-Tompkins Algorithim. Also RR intervals are computed and summarized for each recording for future use.

Some major components of the implementation included:

- Preprocessing/Filtering ECG signals with low-pass Butterworth filtering
- Downsampling to reduce noise and improve performance of the signals
- Visualizing filtered ECG signals with time-domain plots
- Extracting useful information in such as RR Intervals

0.3 Important Links

- `py-ecg-detectors` **Documentation**: https://pypi.org/project/py-ecg-detectors/
- **Dataset**: https://physionet.org/content/
- **Guide we used for ECG Filtering**: https://medium.com/@shahbaz.gondal588/understanding-ecg-signal-processing-with-python-b9dd4ea68682