ECG Data Compression and Reconstruction Project Plan

Francesco Rosnati
Numerical Analysis for Machine Learning
Professor Edie Miglio
PoliMi

August 7, 2024

Contents

1 Project Overview

The project involves developing a Python software for ECG data processing, compression, and reconstruction using both non-CS-based and CS-based methods. The software will include modules for data reading, visualization, structuring, compression, reconstruction, and evaluation.

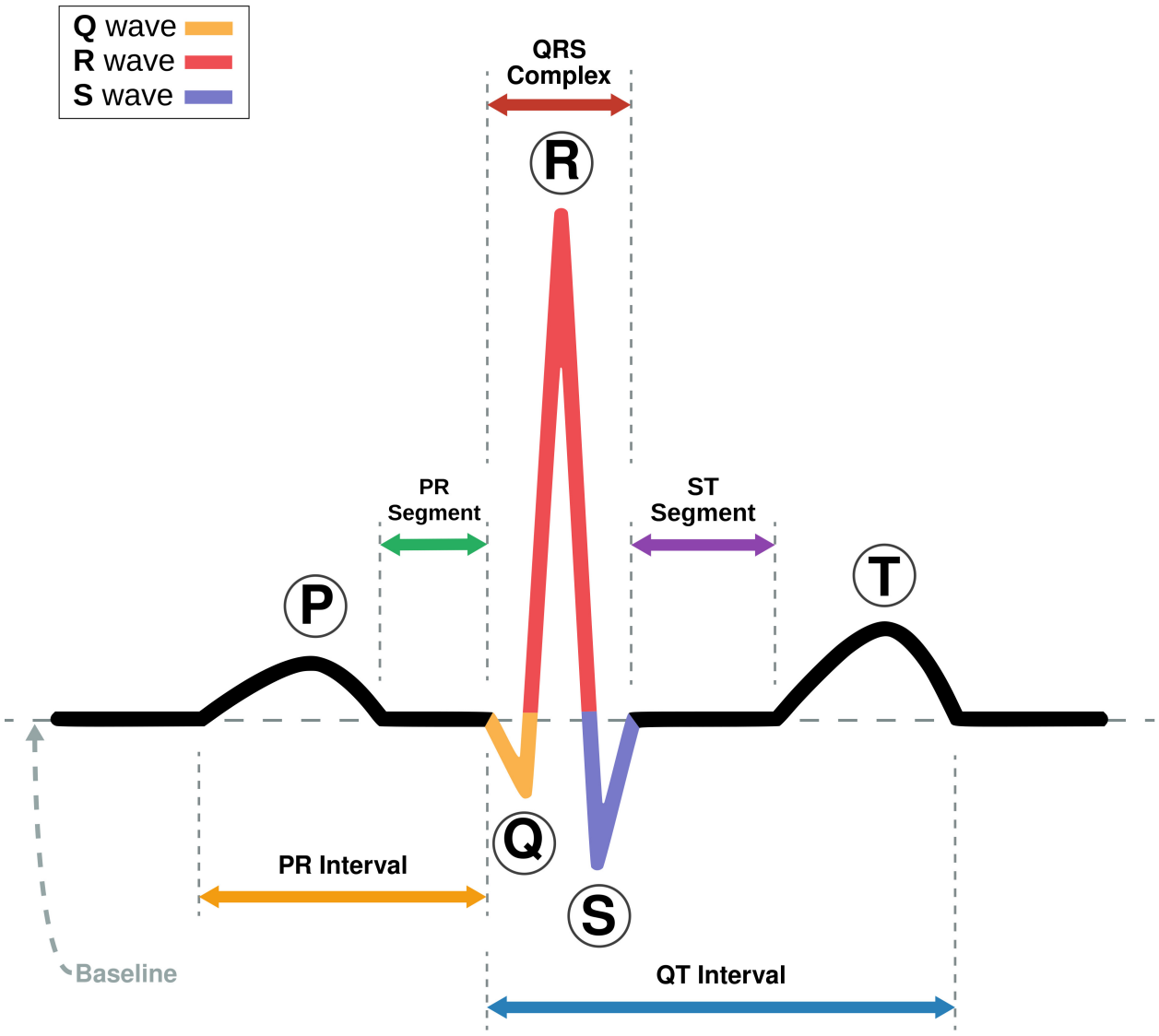

Figure 1: ECG of a heart in normal sinus rhythm

2 First Macro Step: Preprocessing

2.1 Data Reader

Purpose: Read ECG data from various formats.

- Implement functionality to read ECG data files.
- Ensure compatibility with different data formats (CSV, TXT, etc.).

2.2 Data Visualizer

Purpose: Visualize ECG data, specifically the PQRST curves.

- Implement plotting functionality for ECG data.
- **Note**: Ensure compatibility with data structure from Data Structurer or provide a Data Destructurer to revert to the original format before visualization.

2.3 Data Structurer

Purpose: Organize ECG data into a structured format with defined signal lengths.

- Transform the original data into a long series of contiguous memory cells (like a long array).
- Define a parameter for signal length (e.g., 16 coefficients per signal).
- Ensure data is aligned and organized for further processing.
- **Note**: Coordinate with Data Visualizer to ensure compatibility or implement a Data Destructurer.

3 Second Macro Step: Processing/Compression

3.1 Non CS-Based Compressor

Purpose: Compress ECG data using traditional methods.

- Implement transformations (DWT, DCT, etc.).
- Apply sparsification with thresholding.

3.2 CS-Based Compressor with Fixed Dictionary

Purpose: Compress ECG data using compressed sensing with a fixed dictionary.

- Implement compression using CS.
- Utilize the Kronacker technique for improved compression.

3.3 CS-Based Compressor with Adaptive Dictionary Learning

Purpose: Compress ECG data using compressed sensing with adaptive dictionary learning.

- Implement dictionary learning methods (KSVD, MOD).
- Apply CS with the Kronacker technique.
- Learn a different dictionary for each patient.

4 Third Macro Step: Reconstruction

4.1 Reconstruction for Non CS-Based Compression

Purpose: Reconstruct data from non-CS-based compressed measurements.

- Implement reconstruction methods (if different from CS-based, verify this).

4.2 Reconstruction for CS-Based Compression

Purpose: Reconstruct data from CS-based compressed measurements.

- Implement methods such as Basis Pursuit, Greedy algorithm, smooth-L0.
- Focus on smooth-L0 for implementation.
- Consider robustness to noise and implement LASSO for noisy cases.

5 Fourth Macro Step: Evaluation of Results

5.1 Qualitative Assessment

Purpose: Evaluate the quality of reconstructed ECG data.

- Use Data Visualizer to compare original and reconstructed signals.

5.2 Quantitative Assessment

Purpose: Measure the performance of compression and reconstruction.

- Metrics:
  - Actual compression rate.
  - Algorithm complexity and processing speed.
    - Record times of various steps and analyze them.
  - Accuracy:
    - Percentage Root Mean Square Difference (PRD).
    - Signal to Noise Ratio (SNR).

6 Implementation Plan

6.1 Initial Pipeline Development

1. **First Macro Step**
   - Develop Data Reader.
   - Develop Data Visualizer.
   - Develop Data Structurer.
   - **Note**: Ensure compatibility between modules.
2. **Non CS-Based Compressor**
   - Implement and integrate with the initial pipeline.
3. **Reconstruction for Non CS-Based Compression**
   - Implement reconstruction method.
   - Ensure integration with the initial pipeline.
4. **Fourth Macro Step**
   - Implement qualitative and quantitative assessment methods.
   - Ensure the entire pipeline is working and can be tested.

6.2 Future Enhancements

- Add CS-Based Compressor with Fixed Dictionary.
- Add CS-Based Compressor with Adaptive Dictionary Learning.
- Implement Reconstruction for CS-Based Compression.
- Evaluate and compare with the initial pipeline.