REPORT

Zajęcia: Analog and digital electronic circuits

Teacher: prof. dr hab. Vasyl Martsenyuk

Lab 13

24.04.2025

Topic: "ECG Signal Processing: Filtering, R-peak Detection, and Heart Rate Analysis"

Variant: 13

Rafał Żmuda
Informatyka II stopień,
stacjonarne,
1 semestr,
Gr.2b

1. Problem statement:

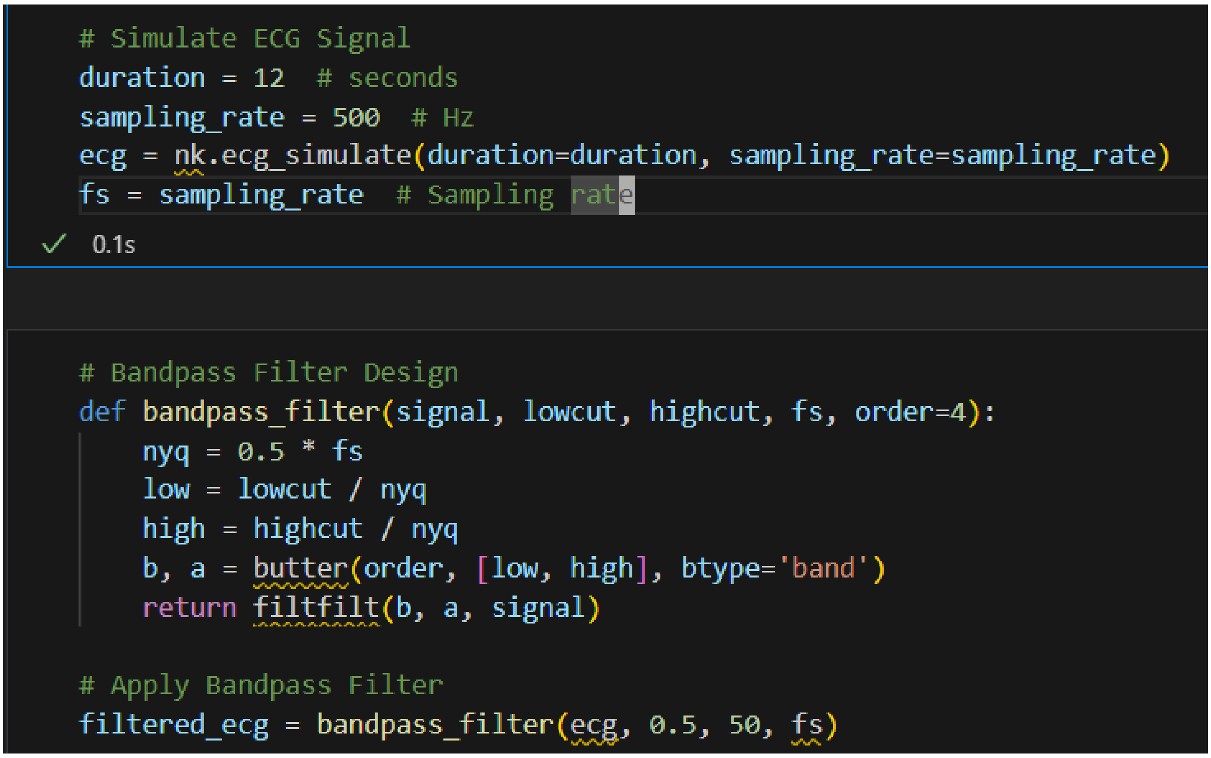
- Preprocess a simulated or real ECG signal.
- Apply bandpass filtering to remove baseline drift and high-frequency noise.
- Detect R-peaks in the filtered signal.
- Calculate and plot heart rate over time.
- Interpret the results based on signal quality and parameter choices.

2. Input data:

Duration: 12 s, Sampling Rate: 500 Hz, Filter: 0.5–50 Hz

3. Commands used (or GUI):

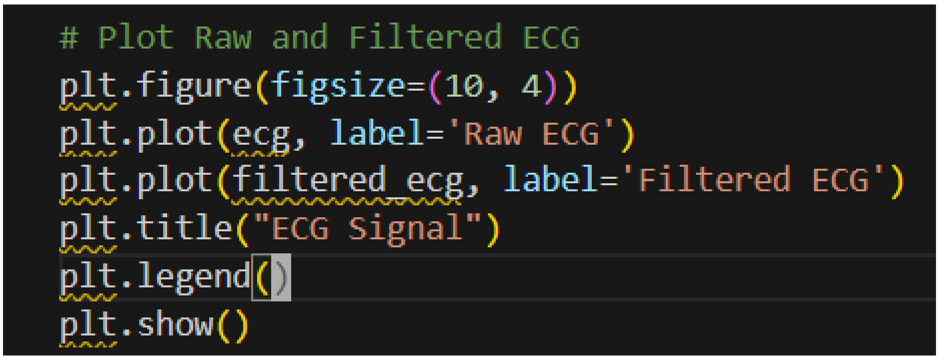
Creating input signal and applying filters



```
# Simulate ECG Signal
duration = 12  # seconds
sampling_rate = 500  # Hz
ecg = nk.ecg_simulate(duration=duration, sampling_rate=sampling_rate)
fs = sampling_rate  # Sampling rate
```

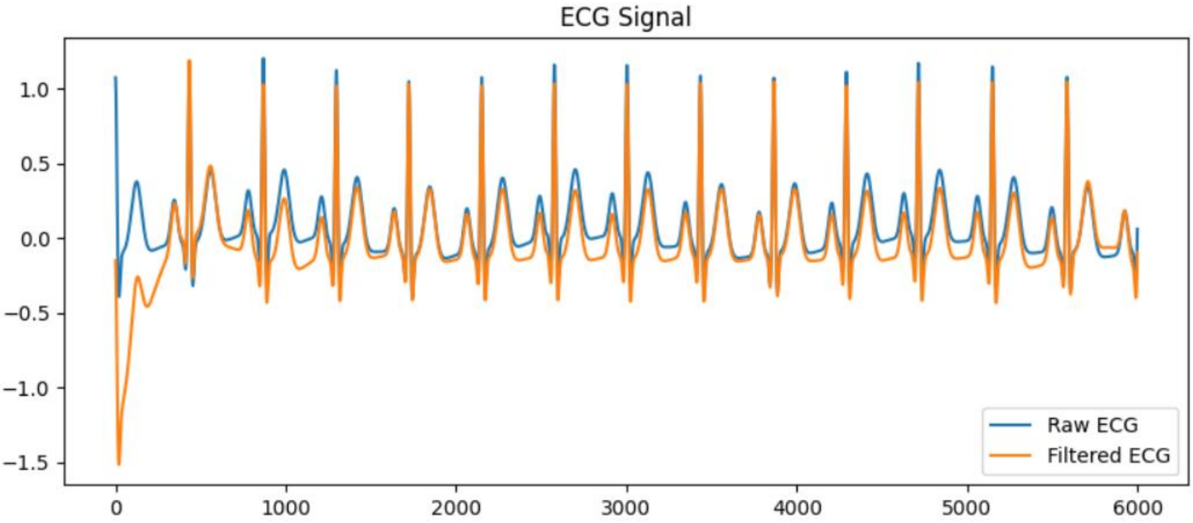
```
# Bandpass Filter Design
def bandpass_filter(signal, lowcut, highcut, fs, order=4):
    nyq = 0.5 * fs
    low = lowcut / nyq
    high = highcut / nyq
    b, a = butter(order, [low, high], btype='band')
    return filtfilt(b, a, signal)

# Apply Bandpass Filter
filtered_ecg = bandpass_filter(ecg, 0.5, 50, fs)
```

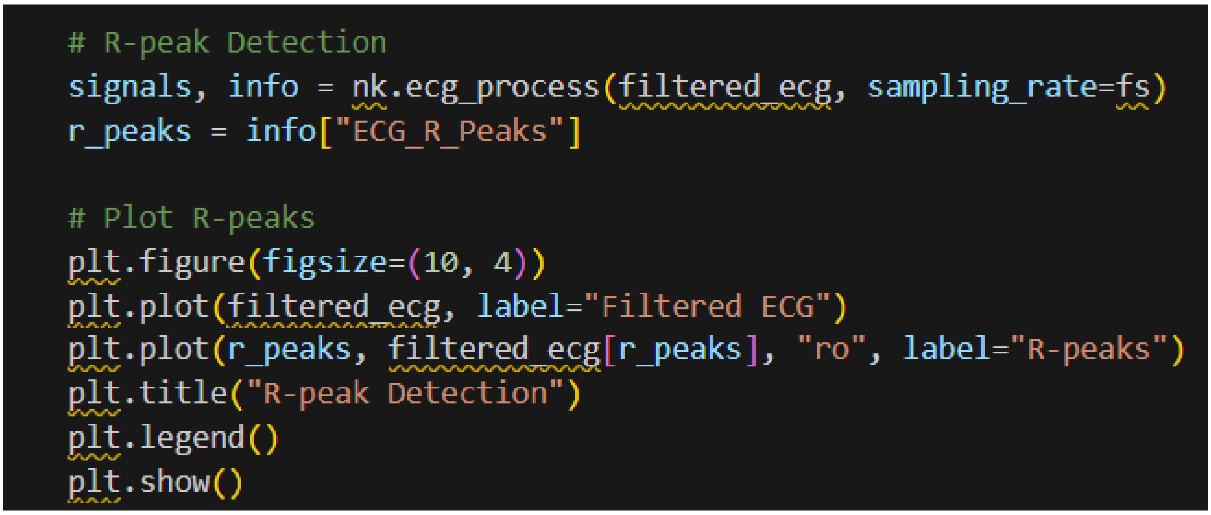


```
# Plot Raw and Filtered ECG
plt.figure(figsize=(10, 4))
plt.plot(ecg, label='Raw ECG')
plt.plot(filtered_ecg, label='Filtered ECG')
plt.title("ECG Signal")
plt.legend()
plt.show()
```

Output:



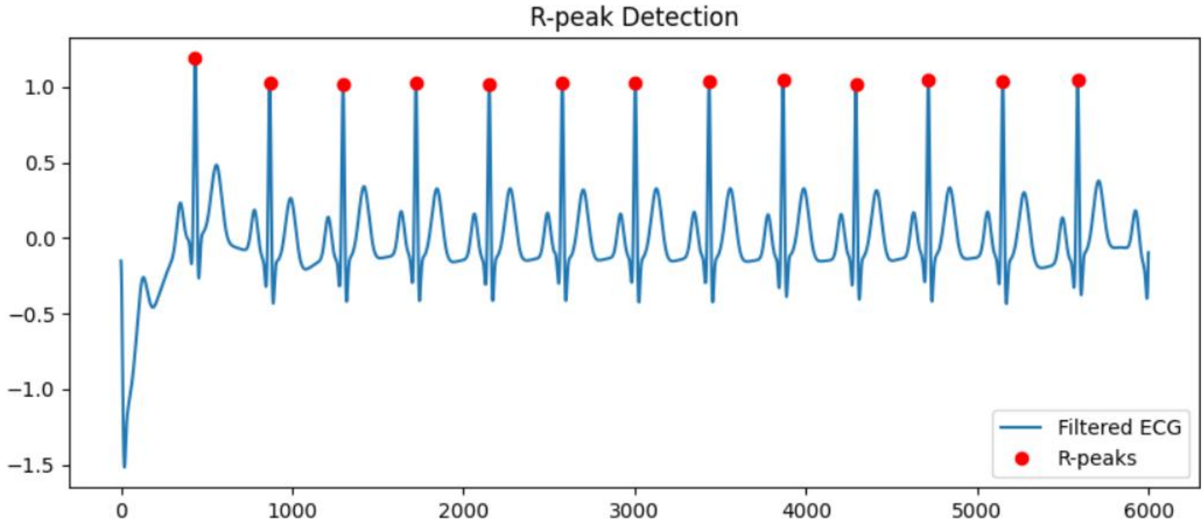
R-peak detection and ploting



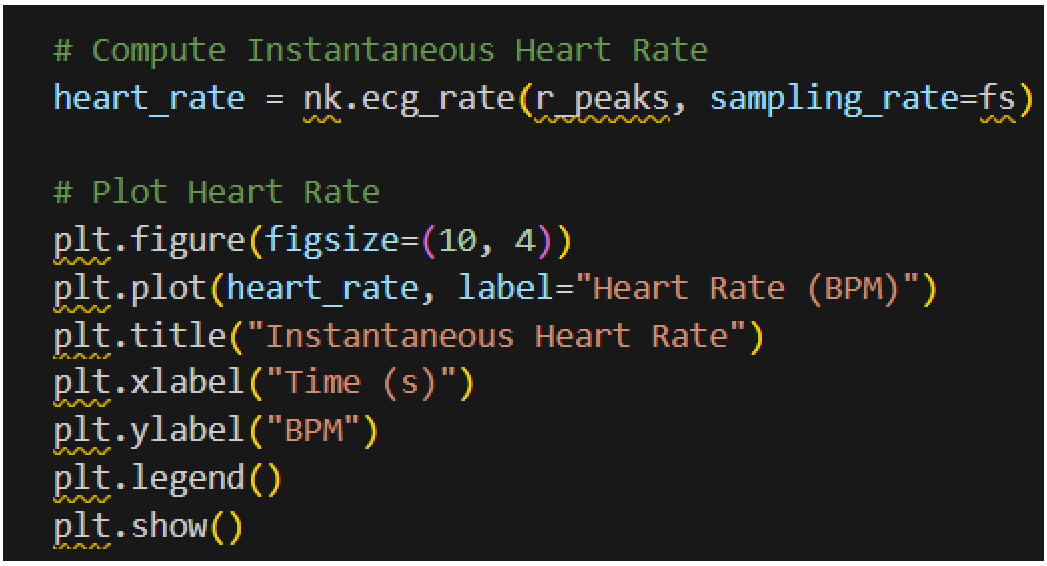
```
# R-peak Detection
signals, info = nk.ecg_process(filtered_ecg, sampling_rate=fs)
r_peaks = info["ECG_R_Peaks"]

# Plot R-peaks
plt.figure(figsize=(10, 4))
plt.plot(filtered_ecg, label="Filtered ECG")
plt.plot(r_peaks, filtered_ecg[r_peaks], "ro", label="R-peaks")
plt.title("R-peak Detection")
plt.legend()
plt.show()
```

Output:



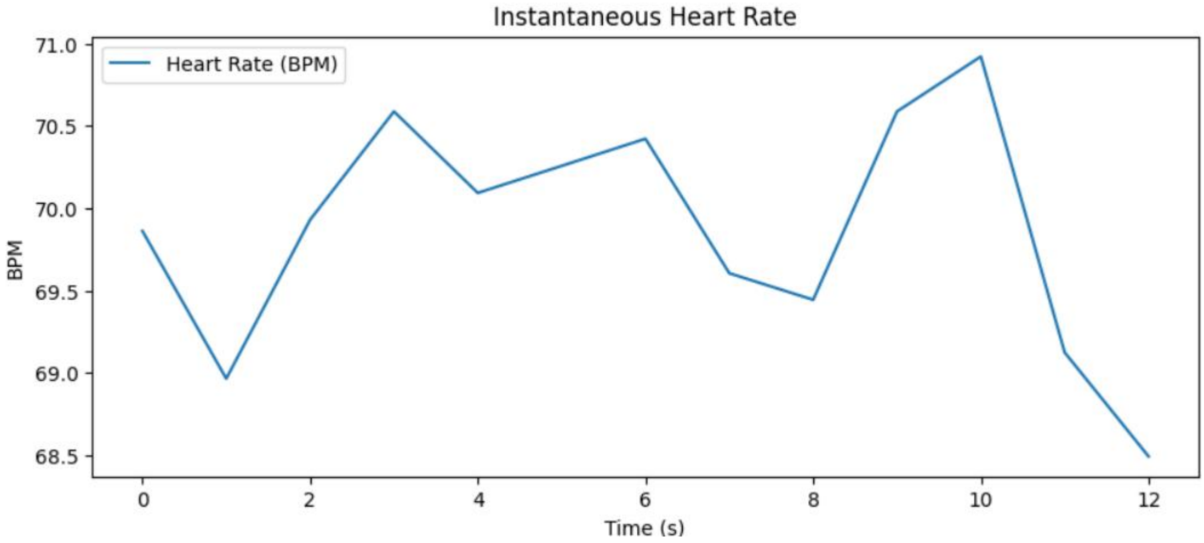
Heartrate calculation and ploting



```
# Compute Instantaneous Heart Rate
heart_rate = nk.ecg_rate(r_peaks, sampling_rate=fs)

# Plot Heart Rate
plt.figure(figsize=(10, 4))
plt.plot(heart_rate, label="Heart Rate (BPM)")
plt.title("Instantaneous Heart Rate")
plt.xlabel("Time (s)")
plt.ylabel("BPM")
plt.legend()
plt.show()
```

Output:



Link to remote repository:
https://github.com/RafalZmu/School/tree/main/Lab%2013

5. Conclusions:

This task demonstrates the application of digital signal processing techniques to analyze ECG signals. By simulating an ECG signal, applying a bandpass filter, and detecting R-peaks, we observe how filtering enhances signal clarity and improves the accuracy of R-peak detection. The computation of instantaneous heart rate further highlights the practical use of these techniques in biomedical signal analysis. Overall, this workflow showcases the importance of preprocessing and feature extraction in ECG analysis, which is critical for applications like heart rate monitoring and arrhythmia detection.