EXPERIMENT: 04

AIM: Heart rate variability analysis

THEORY:

The **Electrocardiogram (ECG)** records the electrical activity of the heart and is widely used for **heart rate (HR) and heart rate variability (HRV) analysis**. Here's how:

1. Extracting Heart Rate from ECG

   - The **ECG signal** consists of P, Q, R, S, and T waves.
   - The **R-wave (R-peak)** is the most prominent peak and represents **ventricular depolarization**.
   - **Heart rate (HR)** is calculated from the R-R interval (RR Interval):

     HR = 60 / Mean RR Interval(s)

   - Example: If **RR Interval = 0.8s**, then **HR = 75 beats per minute (bpm)**.

2. Heart Rate Variability (HRV) Analysis

   HRV refers to the **variation in time between successive R-peaks (RR intervals)** and provides insights into **autonomic nervous system (ANS) regulation**.

   A. Time-Domain Analysis

   - **Mean RR Interval:** Average time between R-peaks.
   - **SDNN (Standard Deviation of RR Intervals):** Measures overall HRV.
   - **RMSSD (Root Mean Square of Successive Differences):** Reflects short-term HRV.

- **pNN50:** Percentage of RR intervals differing by >50 ms.

B. Frequency-Domain Analysis (FFT)

  - **Low-Frequency (LF) Power (0.04 - 0.15 Hz):** Represents sympathetic and parasympathetic activity.
  - **High-Frequency (HF) Power (0.15 - 0.4 Hz):** Represents parasympathetic activity.
  - **LF/HF Ratio:** Used to assess autonomic balance.

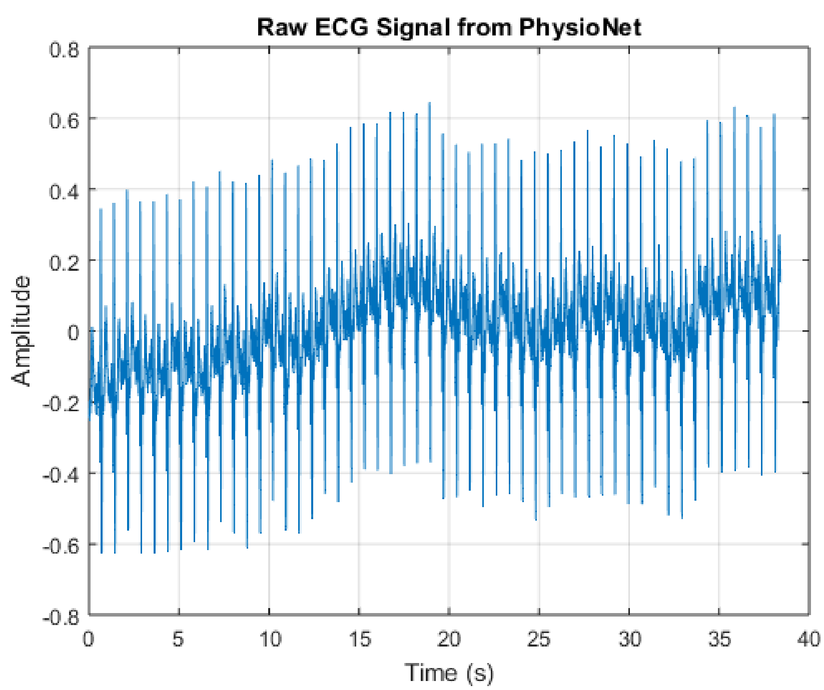
3. Clinical and Research Applications

- **Cardiac Health Monitoring:** Detects arrhythmias, bradycardia, tachycardia, and heart disease.
- **Stress and Mental Health Assessment:** HRV analysis helps evaluate stress levels and autonomic function.
- **Athlete Performance Monitoring:** HR and HRV provide insights into fitness and recovery.
- **Wearable ECG Devices:** Smartwatches and fitness trackers use ECG-based HR monitoring.

CODE:

```
clc; clear all; close all;


% Perform QRS detection using Pan-Tompkins algorithm
% Load ECG signal (select only one channel)
[ecg_signal, fs] = rdsamp('s0010_re');

% Check if ecg_signal has multiple columns
if size(ecg_signal, 2) > 1
    ecg_signal = ecg_signal(:, 1); % Select the first channel
end

% Display basic info
disp(['Sampling Frequency: ', num2str(fs), ' Hz']);
disp(['Signal Length: ', num2str(length(ecg_signal)), ' samples']);

% Plot ECG signal
t = (0:length(ecg_signal)-1) / fs; % Time axis
figure;
plot(t, ecg_signal);
xlabel('Time (s)');
ylabel('Amplitude');
title('Raw ECG Signal from PhysioNet');
grid on;
```

```
Sampling Frequency: 1000 Hz
Signal Length: 38400 samples
```



Step 1: Bandpass Filtering (0.5 - 50 Hz)

```
low_freq = 0.5;
high_freq = 50;
[b, a] = butter(3, [low_freq high_freq] / (fs/2), 'bandpass');
filtered_ecg = filtfilt(b, a, ecg_signal);
```

Step 2: Differentiation

```
diff_ecg = diff(filtered_ecg);
```

Step 3: Squaring

```
squared_ecg = diff_ecg .^ 2;
```

Step 4: Moving Window Integration

```
window_size = round(0.150 * fs); % 150ms window
mwi_ecg = filter(ones(1, window_size)/window_size, 1, squared_ecg);

% Ensure mwi_ecg is a vector
mwi_ecg = mwi_ecg(:);  % Convert to column vector if needed
```

Step 5: Peak Detection with Adaptive Threshold

```
threshold = 0.6 * max(mwi_ecg); % Initial threshold
[~, r_peaks] = findpeaks(mwi_ecg, 'MinPeakHeight', threshold, 'MinPeakDistance', round(0.6 *
fs));
```

Compute RR Intervals (Time Between R-peaks)

```
rr_intervals = diff(r_peaks) / fs; % Convert to seconds

% Time-domain HRV features
mean_rr = mean(rr_intervals);
sdnn = std(rr_intervals); % Standard deviation of RR intervals
rmssd = sqrt(mean(diff(rr_intervals).^2)); % Root Mean Square of Successive Differences (RMSSD)
nn50 = sum(abs(diff(rr_intervals)) > 0.05); % Number of RR intervals > 50ms
pNN50 = (nn50 / length(rr_intervals)) * 100; % Percentage of NN50

% Display HRV Metrics
disp(['Mean RR Interval: ', num2str(mean_rr), ' s']);
disp(['SDNN: ', num2str(sdnn), ' s']);
disp(['RMSSD: ', num2str(rmssd), ' s']);
disp(['pNN50: ', num2str(pNN50), ' %']);
```

```
Mean RR Interval: 0.73349 s
SDNN: 0.010047 s
```

```
RMSSD: 0.012707 s
pNN50: 0 %
```

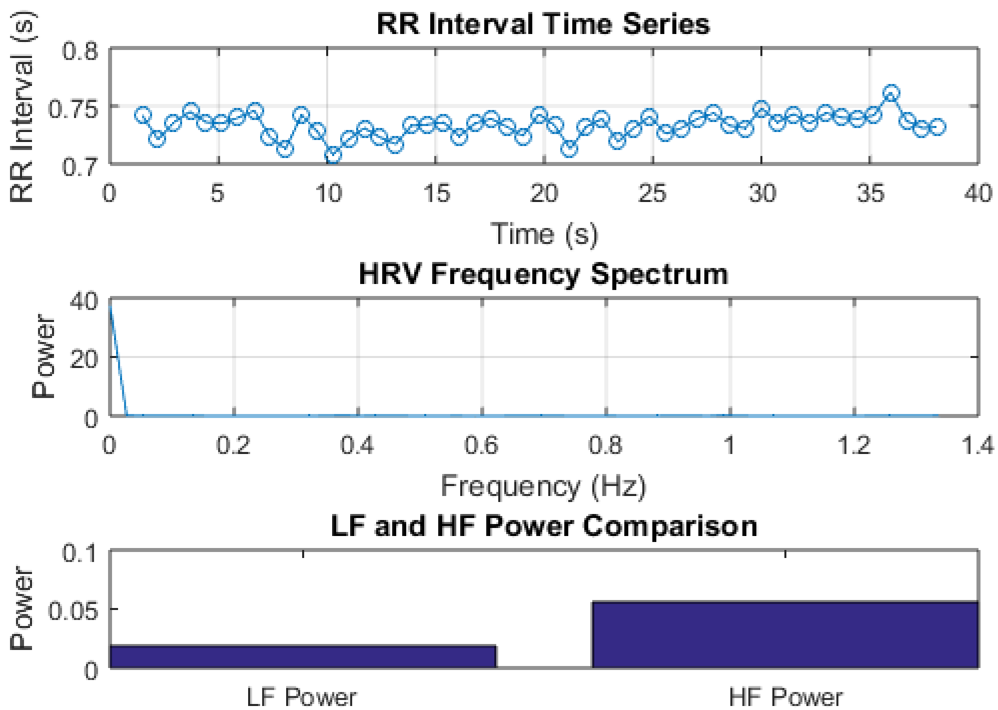
Frequency-domain HRV analysis using FFT

```
N = length(rr_intervals);
fs_rr = 1 / mean_rr; % RR sampling frequency
f = (0:N-1) * fs_rr / N;
rr_fft = abs(fft(rr_intervals));

% Extract LF and HF power
lf_band = (f >= 0.04 & f <= 0.15);
hf_band = (f > 0.15 & f <= 0.4);
lf_power = sum(rr_fft(lf_band).^2);
hf_power = sum(rr_fft(hf_band).^2);
lf_hf_ratio = lf_power / hf_power;

% Display frequency-domain metrics
disp(['LF Power: ', num2str(lf_power)]);
disp(['HF Power: ', num2str(hf_power)]);
disp(['LF/HF Ratio: ', num2str(lf_hf_ratio)]);
```

```
LF Power: 0.018751
HF Power: 0.055954
LF/HF Ratio: 0.33512
```

Plot HRV Results

```
figure;
subplot(3,1,1);
plot(r_peaks(2:end) / fs, rr_intervals, 'o-');
xlabel('Time (s)'); ylabel('RR Interval (s)');
title('RR Interval Time Series'); grid on;

subplot(3,1,2);
plot(f, rr_fft);
xlabel('Frequency (Hz)'); ylabel('Power');
title('HRV Frequency Spectrum'); grid on;

subplot(3,1,3);
bar([lf_power, hf_power]);
set(gca, 'XTickLabel', {'LF Power', 'HF Power'});
ylabel('Power'); title('LF and HF Power Comparison');
```



CONCLUSION: The code successfully detects QRS complexes using the Pan-Tompkins algorithm and analyzes heart rate variability (HRV) in both time and frequency domains. It provides key HRV metrics such as mean RR interval, SDNN, RMSSD, and LF/HF ratio, which help assess autonomic nervous system activity. The results can be used for cardiac health monitoring and stress analysis.