

RR Interval Variability for Detecting Irregular Heart Rhythms

Nora Dabbagh

Department of Biomedical Engineering
University of Illinois Chicago

Revised October 2026: results regenerated in MATLAB R2024b with a fixed random seed; detected R peaks added to Figs. 1–2; figure numbering and citations corrected.

Abstract—This project used MATLAB to test whether RR interval variability can distinguish a regular ECG-like rhythm from an irregular ECG-like rhythm. We simulated two ECG-like signals over 10 s at 500 Hz. The normal signal used evenly spaced beat times, and the irregular signal used uneven beat times. We detected R peaks with a threshold-based local maximum method, calculated RR intervals from consecutive R-peak times, and compared the mean RR interval, standard deviation, and coefficient of variation. The normal rhythm produced stable RR intervals (coefficient of variation 0.007). The irregular rhythm produced much higher beat-to-beat variability (coefficient of variation 0.309) at a similar mean RR interval.

Clinical relevance—Simple RR interval analysis can help flag irregular rhythm behavior and support basic ECG screening in wearable or low-cost monitoring systems.

Keywords—ECG, RR interval, R peak detection, heart rate variability, MATLAB, irregular rhythm

I. INTRODUCTION

An electrocardiogram, or ECG, records the heart’s electrical activity over time. Clinicians use ECG signals to evaluate heart rhythm because each heartbeat creates clear waveform features, including the P wave, QRS complex, and T wave [1], [2]. This project focuses on the R peak because it clearly marks each heartbeat and gives a reliable timing point for rhythm analysis.

The RR interval measures the time between consecutive R peaks. A regular rhythm keeps those intervals close together from beat to beat. An irregular rhythm changes that spacing, so the RR intervals spread out more. This timing-based measurement gives a direct way to measure rhythm consistency. Standard heart rate variability methods also use beat-to-beat timing to evaluate rhythm changes [3].

This project tests whether RR interval variability can distinguish a normal ECG-like signal from an irregular ECG-like signal. We used MATLAB to simulate controlled ECG signals, detect R peaks, calculate RR intervals, and compare variability metrics. This project asked if a basic timing method could show a measurable difference between regular and irregular rhythm behavior.

II. METHODS

The signal processing model was created in MATLAB using a 10 s time window and a sampling frequency of 500 Hz.

Two sets of beat times were defined in seconds. The normal rhythm used evenly spaced beats from 1 s to 9 s, while the irregular rhythm used uneven beat times across the same window. This kept the waveform shape controlled and made beat spacing the main variable being tested. A custom MATLAB function called *make_ecg* was used to create each ECG-like signal. For each beat, the function added a small P wave before the R peak, a tall narrow R peak at the beat time, and a wider T wave after the R peak. Gaussian-shaped curves were used because they provide a simple way to create smooth ECG-like peaks. Published synthetic ECG work also uses Gaussian-based waveform models [4]. The purpose was to create a clear signal where the R peaks could be detected and used for timing analysis.

A small amount of random noise was added to both signals using $0.03 \cdot \text{randn}$. This made the signals look less perfect and still kept the R peaks clear enough to detect. The random number generator was seeded with $\text{rng}(1)$, so the noise, and every value reported below, can be reproduced exactly. The same R-peak detection method was then applied to both signals. The detection function scanned through the signal and counted a point as an R peak when it was above 0.6, larger than the point before it, larger than the point after it, and at least 0.4 s after the previous detected peak. This helped the code ignore smaller waveform parts, random noise, and repeated points from the same peak.

After the R peaks were detected, RR intervals were calculated using the time difference between neighboring R peaks:

$$RR_i = t_{i+1} - t_i \quad (1)$$

where t_i is the time of one detected R peak and t_{i+1} is the time of the next detected R peak.

For each signal, the mean RR interval, standard deviation, and coefficient of variation were calculated. The mean RR interval shows the average spacing between beats. The standard deviation shows how much the RR intervals spread out from that average. The coefficient of variation compares that spread relative to the mean RR interval:

$$CV = \sigma_{RR} / \mu_{RR} \quad (2)$$

where σ_{RR} is the standard deviation of the RR intervals and μ_{RR} is the mean RR interval. A higher coefficient of variation

means the beat spacing changed more, so it was used as the main indicator of rhythm irregularity.

III. RESULTS

The normal ECG-like signal produced evenly spaced R peaks across the 10 s window. The irregular ECG-like signal also had clear detected R peaks, but the spacing between those peaks changed throughout the signal. Some intervals were shorter, and others were longer. Figs. 1 and 2 show this timing difference, with each detected R peak marked.

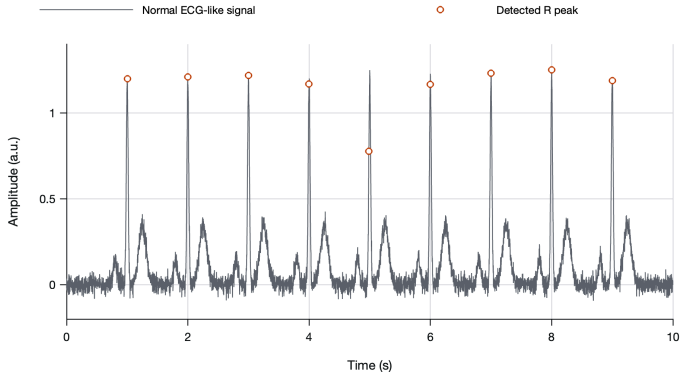


Fig. 1. Normal ECG-like signal with detected R peaks (circles). The beats are evenly spaced. Take-home message: R-peak timing makes the rhythm visible.

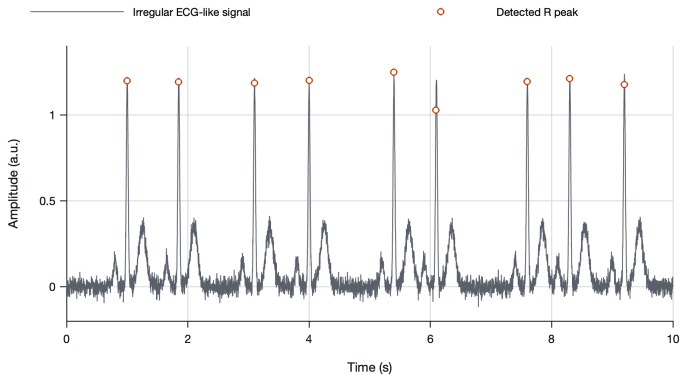


Fig. 2. Irregular ECG-like signal with detected R peaks (circles). The spacing between peaks changes from beat to beat. Take-home message: R-peak timing makes the rhythm difference visible.

The RR interval values showed the same pattern more clearly (Fig. 3). The normal rhythm had RR intervals that stayed close to 1.00 s. The irregular rhythm had RR intervals that changed more from beat to beat, from 0.69 s to 1.51 s. This showed that the two signals had similar overall pacing, but different timing consistency.

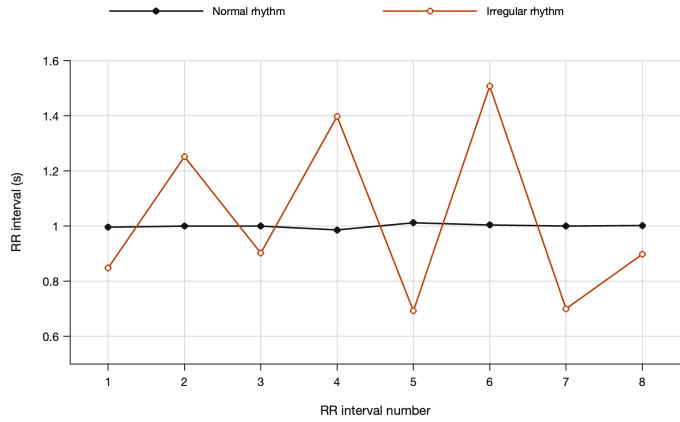


Fig. 3. RR interval comparison for normal and irregular ECG-like signals. The normal rhythm stays almost flat, while the irregular rhythm moves up and down across beats. Take-home message: the irregular signal has more beat-to-beat timing variation.

The variability metrics gave the clearest quantitative comparison between the two signals (Table I, Fig. 4). The normal rhythm had a mean RR interval of 1.000 s, a standard deviation of 0.007 s, and a coefficient of variation of 0.007. The irregular rhythm had a mean RR interval of 1.025 s, a standard deviation of 0.317 s, and a coefficient of variation of 0.309. This means the average beat timing was close between the two signals, but the consistency of that timing was very different: the irregular rhythm’s coefficient of variation was more than 40 times larger.

TABLE I
RR VARIABILITY METRICS

Signal type	Mean RR interval (s)	Standard deviation (s)	Coefficient of variation
Normal ECG	1.000	0.007	0.007
Irregular ECG	1.025	0.317	0.309

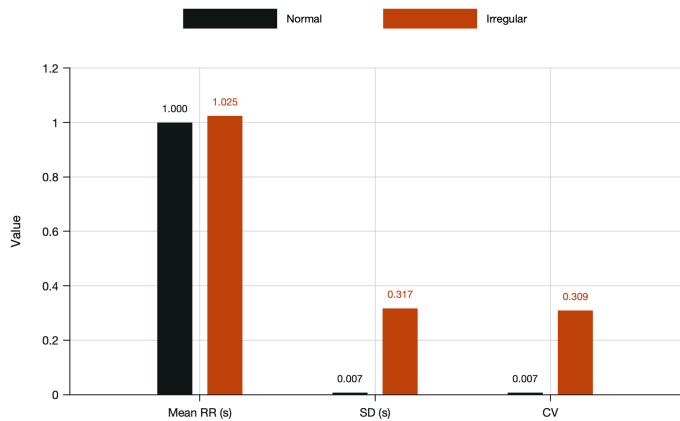


Fig. 4. RR variability metrics for normal and irregular ECG-like signals. The mean RR interval stays similar between signals, and the irregular rhythm has a higher standard deviation and coefficient of variation. Take-home message: the irregular rhythm mainly changes timing consistency.

Overall, the results showed that RR interval variability was much higher in the irregular ECG-like signal than in the nor-

mal ECG-like signal. This supports the idea that even a simple R-peak timing method can detect differences between regular and irregular rhythm patterns.

IV. DISCUSSION AND CONCLUSION

The results show that RR interval variability can separate the regular rhythm from the irregular rhythm in this controlled ECG-like model. The two rhythms had similar mean RR intervals, so their average pacing was close. The main difference was that the irregular rhythm had a much higher standard deviation and coefficient of variation. This means the irregular rhythm was defined by inconsistent beat spacing. This result fits the project goal because RR intervals measure beat-to-beat timing directly. The ECG plots show uneven R-peak spacing in the irregular signal, and the RR interval calculations turn that visual pattern into measurable values. This is useful because numbers like standard deviation and coefficient of variation make rhythm differences easier to compare in a clear and consistent way.

There are important limitations. The ECG signals were simulated, so they do not include the full complexity of real patient ECG recordings. Real ECG data can include motion artifact, baseline drift, stronger noise, and abnormal waveform shapes. The R-peak detection method was also basic because it used a fixed threshold and local maximum approach. It found all 18 beats in this model, but marking the detections exposed a weakness: near 5.0 s in Fig. 1 and 6.1 s in Fig. 2, noise on the rising edge of the R wave formed a local maximum above the threshold, and the 0.4 s window then skipped the true peak. These two detections came about 14 ms and 8 ms early, and they account for most of the normal rhythm's 0.007 s standard deviation; with those beats placed at their true peaks it would fall to about 0.002 s. Taking the largest sample within a short window after each threshold crossing would remove this error. The same fixed threshold may also fail when ECG amplitude changes or noise increases.

This method identifies timing irregularity. A separate clinical interpretation would still be needed to diagnose a specific cardiac condition. Future work should test the same RR interval method on real ECG datasets, such as the MIT-BIH Arrhythmia Database [5]. More advanced R-peak detection methods, such as the Pan-Tompkins algorithm [6], could also improve accuracy when the signal contains more noise or less consistent waveform shapes. The model could also be adjusted by changing the noise level, changing the beat times, or modifying the height and width of the P waves, R peaks, and T waves. Overall, this project shows that RR interval variability works as a simple MATLAB-based method for detecting irregular rhythm behavior in ECG-like signals.

REFERENCES

- [1] American Heart Association, "Electrocardiogram (ECG or EKG)," American Heart Association.
- [2] Mayo Clinic Staff, "Electrocardiogram (ECG or EKG)," Mayo Clinic, 2023.
- [3] Task Force of the European Society of Cardiology and the North American Society of Pacing and Electrophysiology, "Heart rate variability: Standards of measurement, physiological interpretation, and clinical use," *Circulation*, vol. 93, no. 5, pp. 1043–1065, 1996.
- [4] O. Sayadi and M. B. Shamsollahi, "Synthetic ECG generation and Bayesian filtering using a Gaussian wave-based dynamical model," *Physiological Measurement*, vol. 31, no. 10, pp. 1335–1350, 2010.
- [5] G. B. Moody and R. G. Mark, "The impact of the MIT-BIH Arrhythmia Database," *IEEE Engineering in Medicine and Biology Magazine*, vol. 20, no. 3, pp. 45–50, 2001.
- [6] J. Pan and W. J. Tompkins, "A real-time QRS detection algorithm," *IEEE Transactions on Biomedical Engineering*, vol. BME-32, no. 3, pp. 230–236, 1985.

APPENDIX: MATLAB CODE

The complete script used for every figure and value in this paper (rr_variability.m). The plots in Figs. 1–4 were redrawn from these variables in a consistent print style.

```
% Nora Dabbagh and Dania Sirhan
% BME 240 Term Project
% RR Interval Variability for Detecting Irregular Heart
Rhythms
clear; clc; close all
% fixed random seed so the noise, and every reported number,
can be reproduced
rng(1);
% samples per second
fs = 500;
% 10 seconds of time
t = 0:1/fs:10;
% beat times in seconds
normal_beats = [1 2 3 4 5 6 7 8 9];
irregular_beats = [1 1.85 3.1 4.0 5.4 6.1 7.6 8.3 9.2];
% make the ECG signals
normal_ecg = make_ecg(t, normal_beats);
irregular_ecg = make_ecg(t, irregular_beats);
% add a little noise so it looks more realistic
normal_ecg = normal_ecg + 0.03*randn(size(normal_ecg));
irregular_ecg = irregular_ecg + 0.03*randn(size(irregular_ecg));
% detect R peaks
[normal_peak_values, normal_peak_times] =
detect_r_peaks(normal_ecg, t);
[irregular_peak_values, irregular_peak_times] =
detect_r_peaks(irregular_ecg, t);
% calculate RR intervals
normal_rr = diff(normal_peak_times);
irregular_rr = diff(irregular_peak_times);
% calculate numbers for results table
normal_mean = mean(normal_rr);
normal_std = std(normal_rr);
normal_cv = normal_std / normal_mean;
irregular_mean = mean(irregular_rr);
irregular_std = std(irregular_rr);
irregular_cv = irregular_std / irregular_mean;
% print results
fprintf('NORMAL ECG\n')
fprintf('Mean RR interval: %.3f seconds\n', normal_mean)
fprintf('Standard deviation: %.3f seconds\n', normal_std)
fprintf('Coefficient of variation: %.3f\n\n', normal_cv)
fprintf('IRREGULAR ECG\n')
fprintf('Mean RR interval: %.3f seconds\n', irregular_mean)
fprintf('Standard deviation: %.3f seconds\n', irregular_std)
fprintf('Coefficient of variation: %.3f\n', irregular_cv)
% Figure 1: normal ECG
figure
plot(t, normal_ecg)
hold on
plot(normal_peak_times, normal_peak_values, 'ro')
title('Normal ECG with Detected R Peaks')
xlabel('Time (s)')
ylabel('Amplitude')
grid on
% Figure 2: irregular ECG
figure
plot(t, irregular_ecg)
hold on
plot(irregular_peak_times, irregular_peak_values, 'ro')
title('Irregular ECG with Detected R Peaks')
xlabel('Time (s)')
ylabel('Amplitude')
grid on
% Figure 3: RR interval comparison
figure
plot(1:length(normal_rr), normal_rr, 'o-')
hold on
plot(1:length(irregular_rr), irregular_rr, 'o-')
title('RR Interval Comparison')
xlabel('Beat Number')
ylabel('RR Interval (s)')
legend('Normal Rhythm', 'Irregular Rhythm')
grid on
% Table figure for results
figure
metrics = categorical({'Mean RR', 'Std Dev', 'CV'});
normal_vals = [normal_mean normal_std normal_cv];
irregular_vals = [irregular_mean irregular_std irregular_cv];
bar(metrics, [normal_vals; irregular_vals])
title('RR Variability Metrics')
ylabel('Value')
legend('Normal', 'Irregular')
grid on
function ecg = make_ecg(t, beat_times)
    ecg = zeros(size(t));
    for i = 1:length(beat_times)
        beat = beat_times(i);
        % P wave
        ecg = ecg + 0.15*exp(-((t-(beat-0.2)).^2)/(2*0.03^2));
        % R peak
        ecg = ecg + 1.2*exp(-((t-beat).^2)/(2*0.015^2));
        % T wave
        ecg = ecg + 0.35*exp(-((t-
(beat+0.25)).^2)/(2*0.06^2));
    end
end
function [peak_values, peak_times] = detect_r_peaks(signal, t)
    threshold = 0.6;
    peak_values = [];
    peak_times = [];
    for i = 2:length(signal)-1
        if signal(i) > threshold && signal(i) > signal(i-1) &&
signal(i) > signal(i+1)
            if isempty(peak_times)
                peak_values(end+1) = signal(i);
                peak_times(end+1) = t(i);
            elseif t(i) - peak_times(end) > 0.4
                peak_values(end+1) = signal(i);
                peak_times(end+1) = t(i);
            end
        end
    end
end
end
```