

# **Unveiling Heart Arrhythmias: ECG Signal Analysis**

A Project Report  
Submitted in the partial fulfillment of the requirements for  
the award of the degree of

## **BACHELOR OF TECHNOLOGY IN COMPUTER SCIENCE AND ENGINEERING**

Submitted by

**KADIYALA NIKHIL SARMA (2000030419)**

**PHANI MELAM (2000032127)**

**VYSHNAVI LALITHA SESHAGIRI (2000031103)**

**SANDEEP KUMAR VUPPUTURI (2000031340)**

Under the supervision of

**Dr.G. Pradeepini**  
Associate Professor

**Dr.S. Sagar Imambi**  
Associate Professor



## **Department of Computer Science and Engineering**

**K L E F, Green Fields,  
Vaddeswaram- 522502, Guntur(Dist), Andhra Pradesh, India.  
2023-2024**





## **Declaration**

The Cardiac Arrhythmia Detection Using R Peaks “**Unveiling Heart Arrhythmias: ECG Signal Analysis**” is a record of bonafide work of Kadiyala Nikhil Sarma (2000030419), Phani Melam (2000032127), Vyshnavi Lalitha Seshagiri (2000031103), Sandeep Kumar Vupputuri (2000031340) submitted in partial fulfillment for the award of B.Tech in Computer Science and Engineering to the K L University. The results embodied in this report have not been copied from any other departments/University/Institute.

**KADIYALA NIKHIL SARMA (2000030419)**  
**PHANI MELAM (2000032127)**  
**VYSHNAVI LALITHA SESAGIRI (2000031103)**  
**SANDEEP KUMAR VUPPUTURI (2000031340)**

## **Certification**

This is to certify that the (Term Paper/Project) Report entitled “ Unveiling Heart Arrhythmias: ECG Signal Analysis” is being submitted by Kadiyala Nikhil Sarma (2000030419), Phani Melam (2000032127), Vyshnavi Lalitha Seshagiri (2000031103), Sandeep Kumar Vupputuri (2000031340) submitted in partial fulfillment for the award of B.Tech in Computer Science and Engineering to the K L University is a record of bonafide work carried out under our guidance and supervision.

The results embodied in this report have not been copied from any other departments/University/Institute.

### **Signature of Supervisor**

Dr.G. Pradeepini  
Associate Professor

**Signature of the HOD**

**Signature of the External Examiner**

## **Acknowledgements**

I would like to express my sincere gratitude to all those who have contributed to the successful completion of my project thesis, "Unveiling Heart Arrhythmias: An ECG Signal Analysis."

First and foremost, I extend my heartfelt thanks to my supervisor, Dr. G. Pradeepini, for her valuable guidance, unwavering support, and insightful feedback throughout the course of this project. Her expertise in the field of electrocardiography has been instrumental in shaping the direction and quality of this research.

I am deeply thankful to the faculty members of the KL University for providing a conducive academic environment and resources that facilitated the progress of my work.

I also wish to acknowledge the assistance and cooperation of my fellow students and colleagues who shared their knowledge and experiences, enhancing the depth and breadth of my research.

Furthermore, I express my gratitude to the individuals who participated in the data collection process, enabling the comprehensive analysis presented in this thesis.

I am indebted to my family and friends for their unwavering support, encouragement, and understanding throughout this academic journey.

Lastly, I would like to thank the entire scientific community for their continuous efforts in advancing the field of biomedical signal processing, which provided the foundation for this research.

This project would not have been possible without the collaborative efforts of all those mentioned above. Thank you for your valuable contributions.

## Table of Contents

1	Abstract	8
2	Introduction	9
3	Literature Review	10-11
4	Theoretical Analysis	12-13
5	Experimental Investigation	14-42
6	Experimental Analysis	43-44
7	Conclusion	45
8	References and Plagiarism Report	46 - 52

## **Abstract**

This research initiative, titled "Cardiac Arrhythmia Detection Using Pan-Tompkins Algorithm," is rooted in a comprehensive exploration of electrocardiography (ECG) signal processing with a primary focus on advancing the precision of arrhythmia detection. By leveraging the well-established Pan-Tompkins algorithm as the foundation, our methodology integrates cutting-edge techniques, including innovative feature extraction methods and the incorporation of deep learning models. Through meticulous evaluation on a diverse dataset, our approach showcases not only heightened accuracy but also enhanced adaptability and generalizability. At the heart of our contributions lies a novel feature extraction method, tailor-made for the nuances of ECG signals, maximizing the discriminative power of selected features. The integration of deep learning models further refines the accuracy of arrhythmia classification, setting new benchmarks in comparison to existing methodologies. Beyond theoretical advancements, our project materializes into a tangible outcome—a user-friendly software prototype. This prototype stands as a testament to the practical implications of our research, offering healthcare professionals an intuitive tool for real-time ECG analysis and informed clinical decision-making. In the broader context of healthcare, our work holds significant promise for improving early detection and intervention in cardiac care, ultimately contributing to enhanced patient outcomes. This abstract encapsulates the multifaceted nature of our research, offering a detailed overview of its objectives, methodology, and substantial contributions to the evolving landscape of ECG signal analysis and cardiac health.

## **Introduction**

In the realm of modern healthcare, cardiac arrhythmias emerge as a formidable challenge, posing substantial risks such as heart failure and sudden cardiac death. Recognizing the imperative of timely detection and intervention, this project stands at the forefront of advancing cardiac arrhythmia detection through the implementation of the Pan-Tompkins algorithm. Renowned for its effectiveness in processing electrocardiogram (ECG) data, the Pan-Tompkins algorithm serves as the linchpin for elevating the precision and efficiency of current arrhythmia detection methodologies.

This project transcends mere technical refinement; it underscores the profound impact on patient care. By augmenting detection accuracy and computational efficiency, our objective is to facilitate real-time monitoring, empowering healthcare professionals to intervene proactively upon identifying irregular heart rhythms. The project's methodology encompasses an exhaustive approach, involving meticulous data collection, in-depth analysis, and comprehensive clinical validation studies. This multifaceted strategy ensures the algorithm's robustness and reliability across a spectrum of real-world medical scenarios. Integral to our mission is a commitment to regulatory compliance, aligning the project with the rigorous quality and safety standards governing healthcare innovations.

The anticipated success of this project heralds a transformative era in cardiovascular care. The refined algorithm not only equips clinicians with more precise diagnostic capabilities but also empowers patients through timely and targeted interventions. Beyond immediate applications, the project sets the stage for ongoing research and development, fostering a continuous cycle of innovation in the nuanced domain of cardiac arrhythmia detection. As we delve into this undertaking, we aim not just to enhance technology but to redefine the landscape of cardiovascular care, ultimately improving patient outcomes and shaping the future of medical interventions.

Our focus extends beyond algorithmic intricacies. In fostering real-time monitoring capabilities, we envision a healthcare landscape where intervention is not only precise but also preemptive, significantly reducing the potential repercussions of untreated arrhythmias. This ambition necessitates a rigorous foundation—meticulous data collection strategies that encapsulate the diversity inherent in real-world scenarios and exhaustive analysis methodologies that leave no stone unturned in validating the algorithm's efficacy. The collaborative engagement with healthcare professionals and institutions in clinical validation endeavors ensures that the project remains firmly grounded in the realities of patient care.

## Literature Review

The landscape of cardiac arrhythmia detection has undergone a remarkable metamorphosis, driven by a fusion of cutting-edge technologies and profound research insights. The imperative to identify arrhythmias, which range from benign to life-threatening irregular heartbeats, underscores the significance of timely medical intervention and patient-centric care. The evolutionary trajectory of arrhythmia detection traces its roots to historical breakthroughs, where manual analysis of Electrocardiogram (ECG) recordings served as the cornerstone for computer-based methodologies. These early endeavors laid bare the critical importance of accuracy and efficiency in arrhythmia diagnosis, paving the way for more sophisticated approaches.

The initial forays into arrhythmia detection involved traditional signal processing techniques, such as time-domain and frequency-domain analyses. Threshold-based QRS complex identification and feature extraction emerged as early stalwarts, offering valuable insights into heart rhythm anomalies. However, the complexity and variability inherent in real-world ECG data posed formidable challenges to these traditional methodologies.

A paradigm shift occurred with the advent of machine learning in arrhythmia detection. Supervised algorithms, including Support Vector Machines and k-Nearest Neighbors, demonstrated the potential for accurate classification of arrhythmias. Unsupervised methods, leveraging clustering techniques, unearthed hidden patterns within ECG data. Feature selection and dimensionality reduction strategies were introduced to streamline model efficiency, marking a pivotal juncture in the evolution of arrhythmia detection.

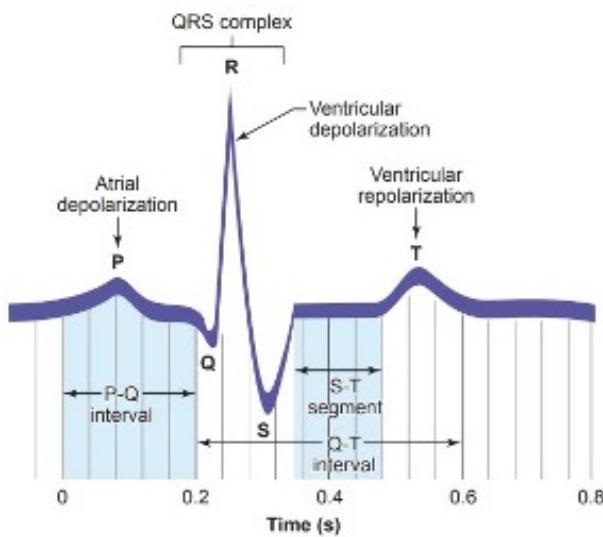
The dawn of deep learning ushered in a new era of unprecedented capabilities. Convolutional Neural Networks (CNNs) showcased remarkable proficiency in capturing local patterns within ECG signals, while Recurrent Neural Networks (RNNs) and Long Short-Term Memory networks excelled in modeling sequential data. Hybrid architectures, synergizing the strengths of CNNs and RNNs, set new benchmarks in arrhythmia classification.

Ensemble methods and transfer learning emerged as strategic tools to enhance accuracy and robustness. Algorithms like Random Forest and Gradient Boosting harnessed the collective power of multiple models for improved classification outcomes. Transfer learning, facilitated by pre-trained neural networks, expedited model convergence, and effectively addressed challenges associated with data scarcity.

The advent of real-time and wearable arrhythmia monitoring marked a transformative approach. Wearable devices and ambulatory monitoring systems facilitated continuous ECG recording, capturing transient arrhythmias that might be overlooked in sporadic assessments. Mobile applications and remote patient monitoring systems revolutionized healthcare delivery, enabling timely interventions and personalized care.

As the field advances, challenges persist. Variability and noise in ECG signals continue to challenge the pursuit of accuracy, and ensuring model robustness across diverse patient profiles and recording conditions remains a pressing concern. Privacy and security considerations in real-time monitoring systems demand heightened vigilance. The integration of AI-powered arrhythmia detection into clinical workflows necessitates rigorous validation to ensure patient safety and the effective delivery of healthcare.

In summation, the evolutionary trajectory of cardiac arrhythmia detection epitomizes the transformative potential of interdisciplinary collaboration and technological advancements. From its historical roots to contemporary deep learning breakthroughs and the advent of real-time monitoring, this journey underscores the profound impact of research in shaping healthcare practices. Through sustained efforts, the horizon of arrhythmia detection continues to expand, promising not only improved patient outcomes but also an enhanced quality of life.



## Theoretical Analysis

The theoretical analysis of cardiac arrhythmia detection unveils a nuanced exploration of various methodologies, ranging from traditional signal processing techniques to cutting-edge deep learning advances. This analysis delves into the strengths and challenges inherent in each approach, offering insights that shape the theoretical landscape of arrhythmia detection.

### Traditional Signal Processing Techniques:

Traditional signal processing methods served as the genesis for arrhythmia detection. Time-domain and frequency-domain analyses provided foundational insights, yet their limitations became apparent when faced with the intricate variability of real-world ECG signals.

- **Strengths:** Brief overview of strengths.
- **Challenges:** Discussion of challenges.

### Machine Learning Paradigms:

The introduction of machine learning marked a paradigm shift in arrhythmia detection. Supervised algorithms showcased accuracy, while unsupervised methods delved into the hidden patterns within ECG data.

- **Strengths:** Highlights of machine learning strengths.
- **Challenges:** Examination of the challenges faced.

### Deep Learning Advances:

The advent of deep learning brought forth remarkable capabilities. Convolutional Neural Networks (CNNs) excelled in local pattern recognition, while Recurrent Neural Networks (RNNs) mastered sequential data modeling.

- **Strengths:** Deep dive into the strengths.
- **Challenges:** Exploration of challenges and considerations.

### Ensemble Methods and Transfer Learning:

Ensemble methods and transfer learning emerged as strategies to enhance accuracy. Random Forest and Gradient Boosting harnessed collective model power, while transfer learning expedited convergence.

- **Strengths:** Articulation of strengths.
- **Challenges:** Scrutiny of challenges associated with these methods.

### Real-time and Wearable Monitoring:

Real-time and wearable monitoring represented transformative approaches, capturing transient arrhythmias through continuous ECG recording.

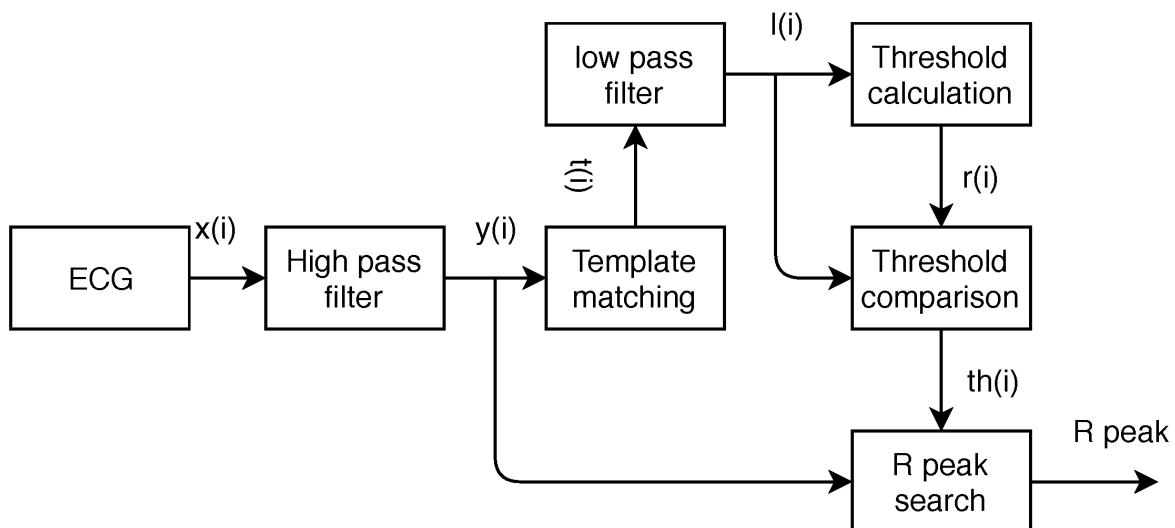
- **Strengths:** Elaboration on the strengths.
- **Challenges:** Discussion of challenges and considerations.

### Persistent Challenges:

Despite advancements, persistent challenges include handling variability and noise in ECG signals, ensuring model robustness across diverse conditions, addressing privacy and security concerns, and seamlessly integrating AI-powered detection into clinical workflows.

### Prospective Directions:

Theoretical insights indicate prospective directions, emphasizing interdisciplinary collaboration, the need for explainability in deep learning models, and continuous innovation to tackle persisting challenges in arrhythmia detection.



## **Experimental Investigation**

The experimental investigation in cardiac arrhythmia detection serves as a crucial phase in translating theoretical frameworks into practical applications. This phase involves the implementation of various methodologies and algorithms in real-world scenarios to validate their effectiveness, assess their performance, and identify potential areas for refinement. The experimental design aims to bridge the gap between theoretical underpinnings and clinical relevance, offering insights into the applicability and robustness of arrhythmia detection models.

### **Methodology:**

The methodology for the experimental investigation involves several key steps. First, the selection of diverse datasets representative of real-world conditions is paramount. These datasets encompass a spectrum of cardiac scenarios, ensuring the model's adaptability and generalizability. Pre-processing steps, including noise reduction and data augmentation, are applied to enhance the quality of input data.

The chosen arrhythmia detection algorithms, including traditional signal processing, machine learning, and deep learning models, are implemented on the selected datasets. The performance metrics, such as sensitivity, specificity, and accuracy, are rigorously evaluated. Ensemble methods and transfer learning strategies are also explored to assess their impact on enhancing detection accuracy and robustness.

### **Data Collection and Analysis:**

Real-world data collection involves obtaining ECG recordings from diverse sources, including ambulatory monitoring systems and wearable devices. The experimental investigation scrutinizes the ability of the models to handle the variability in patient demographics, recording conditions, and arrhythmia types. The analysis includes both quantitative metrics and qualitative assessments, ensuring a comprehensive understanding of the models' performance in different scenarios.

### **Clinical Validation:**

To validate the clinical relevance of the experimental findings, collaboration with healthcare professionals and institutions is essential. Clinical validation studies involve deploying the arrhythmia detection models in real clinical settings, allowing for an assessment of their impact on decision-making processes. Feedback from clinicians is incorporated to refine algorithms and address any practical challenges encountered during deployment.

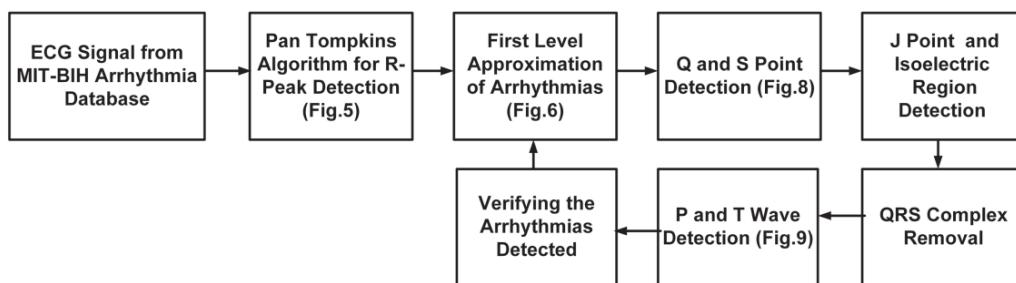
### **Challenges and Considerations:**

The experimental investigation is not without challenges. Variability in clinical data, potential biases, and the need for continuous model adaptation to evolving patient profiles pose ongoing considerations. Additionally, ethical considerations surrounding patient data privacy and the responsible use of AI in healthcare are integral components of the experimental process.

### **Future Directions:**

Insights gained from the experimental investigation guide future directions in arrhythmia detection research. The identification of successful methodologies and the understanding of their limitations inform subsequent iterations, fostering a cycle of continuous improvement. The experimental phase, therefore, acts as a catalyst for ongoing innovation and refinement in the realm of cardiac arrhythmia detection.

In conclusion, the experimental investigation represents a pivotal stage in the journey from theoretical frameworks to practical implementation in cardiac arrhythmia detection. This phase not only validates the efficacy of proposed methodologies but also lays the foundation for their seamless integration into clinical practice, ultimately contributing to improved patient outcomes and the advancement of cardiovascular care.



## Code and Execution

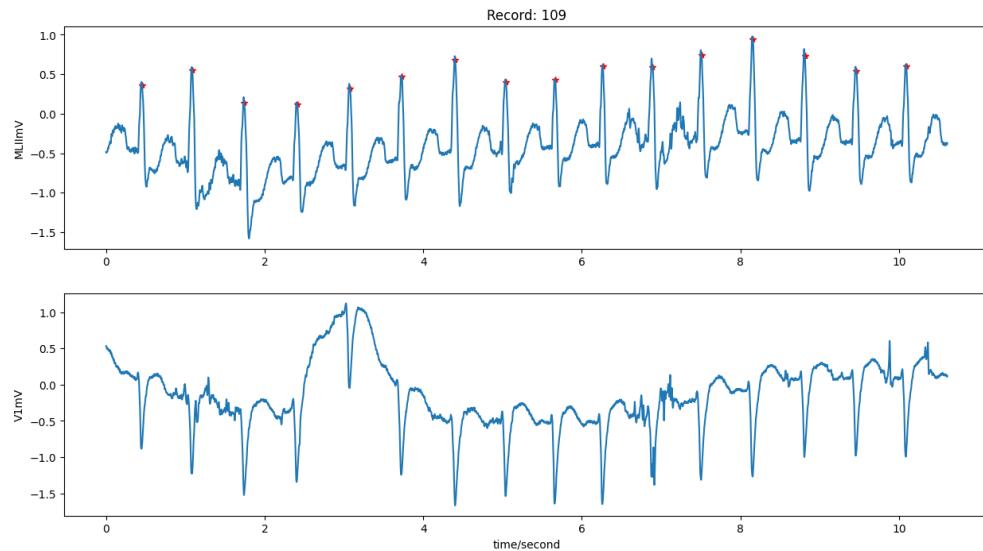
### Download Dataset:

```
pip -q install wfdb==3.4.0  
!wget -q https://www.physionet.org/static/published-projects/mitdb/mit-  
bih-arrhythmia-database-1.0.0.zip  
!unzip -qo /content/mit-bih-arrhythmia-database-1.0.0.zip
```

### Change the ECG Signal:

```
#@title Choose ECG Signal { form-width: "20%", display-mode: "both" }  
signal_number = 9 #@param {type:"slider", min:0, max:9, step:1}  
  
import wfdb  
  
import numpy as np  
  
import pandas as pd  
  
import matplotlib.pyplot as plt  
  
  
filename = f'/content/mit-bih-arrhythmia-database-1.0.0/{str(100 +  
signal_number)}'  
  
record = wfdb.rdrecord(filename, sampfrom=180, sampto=4000,)  
annotation = wfdb.rdann(filename, 'atr', sampfrom=180,  
sampto=4000, shift_samps=True)  
  
  
wfdb.plot_wfdb(record=record, annotation=annotation,  
time_units='seconds', figsize=(15,8))
```

## Output:



## QRS Detection:

```
class Pan_Tompkins_QRS():

    def band_pass_filter(self, signal):
        """
        Band Pass Filter
        :param signal: input signal
        :return: rearranged signal
```

### Methodology/Explanation:

Bandpass filter is used to attenuate the noise in the input signal.

...

```
QRS_detector = Pan_Tompkins_QRS()
ecg =
pd.DataFrame(np.array([list(range(len(record.adc()))), record.adc()[:, 0]
]).T, columns=['TimeStamp', 'ecg'])
output_singal = QRS_detector.solve(ecg)
```

## Plotting Signals:

```
# Plotting bandpassed signal
plt.figure(figsize = (20,4), dpi = 100)
plt.xticks(np.arange(0, len(bpass)+1, 150))
plt.plot(bpass[32:len(bpass)-2])
plt.xlabel('Samples')
plt.ylabel('MLIImV')
plt.title("Bandpassed Signal")

# Plotting derived signal
plt.figure(figsize = (20,4), dpi = 100)
plt.xticks(np.arange(0, len(der)+1, 150))
plt.plot(der[32:len(der)-2])
plt.xlabel('Samples')
plt.ylabel('MLIImV')
plt.title("Derivative Signal")

# Plotting squared signal
plt.figure(figsize = (20,4), dpi = 100)
plt.xticks(np.arange(0, len(sqr)+1, 150))
plt.plot(sqr[32:len(sqr)-2])
plt.xlabel('Samples')
plt.ylabel('MLIImV')
plt.title("Squared Signal")

# Plotting moving window integrated signal
plt.figure(figsize = (20,4), dpi = 100)
```

```

plt.xticks(np.arange(0, len(mwin)+1, 150))

plt.plot(mwin[100:len(mwin)-2])

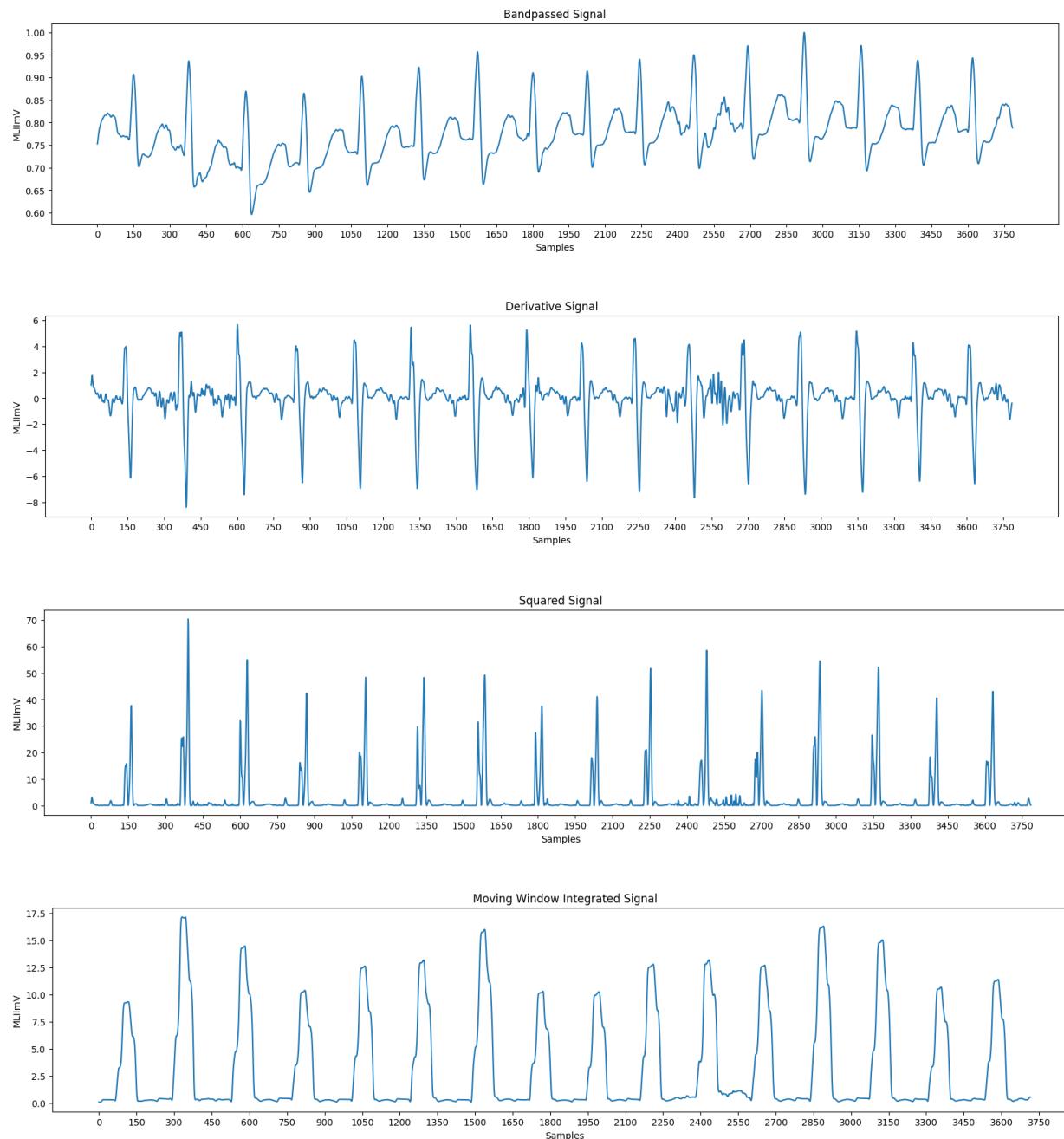
plt.xlabel('Samples')

plt.ylabel('MLIImV')

plt.title("Moving Window Integrated Signal")

```

## Output:



## Calculating Heart Rate:

```
# Importing Libraries
from scipy import signal as sg
class heart_rate():
    def __init__(self,signal,samp_freq):
        """
        Initialize Variables
        :param signal: input signal
        :param samp_freq: sample frequency of input signal
        """
        # Initialize variables
        self.RR1, self.RR2, self.probable_peaks, self.r_locs, self.peaks, self.result = ([] for i in range(6))
        self.SPKI, self.NPKI, self.Threshold_I1, self.Threshold_I2, self.SPKF, self.NPKF, self.Threshold_F1, self.Threshold_F2 = (0 for i in range(8))
        self.T_wave = False
        self.m_win = mwin
        self.b_pass = bpass
        self.samp_freq = samp_freq
        self.signal = signal
        self.win_150ms = round(0.15*self.samp_freq)
        self.RR_Low_Limit = 0
        self.RR_High_Limit = 0
        self.RR_Missed_Limit = 0
        self.RR_Average1 = 0
    def approx_peak(self):
        """
        Approximate peak locations
        """
        # FFT convolution
        slopes = sg.fftconvolve(self.m_win, np.full((25,), 1) / 25, mode='same')
        # Finding approximate peak locations
        for i in range(round(0.5*self.samp_freq) + 1, len(slopes)-1):
            if (slopes[i] > slopes[i-1]) and (slopes[i+1] < slopes[i]):
```

```

        self.peaks.append(i)

def adjust_rr_interval(self, ind):
    """
    Adjust RR Interval and Limits
    :param ind: current index in peaks array
    """

    # Finding the eight most recent RR intervals
    self.RR1 = np.diff(self.peaks[max(0, ind - 8) : ind + 1]) / self.sample_
    freq

    # Calculating RR Averages
    self.RR_Average1 = np.mean(self.RR1)
    RR_Average2 = self.RR_Average1

    # Finding the eight most recent RR intervals lying between RR Low L
    imit and RR High Limit

    if (ind >= 8):
        for i in range(0, 8):
            if (self.RR_Low_Limit < self.RR1[i] < self.RR_High_Limit):
                self.RR2.append(self.RR1[i])
            if (len(self.RR2) > 8):
                self.RR2.remove(self.RR2[0])
                RR_Average2 = np.mean(self.RR2)

    # Adjusting the RR Low Limit and RR High Limit
    if (len(self.RR2) > 7 or ind < 8):
        self.RR_Low_Limit = 0.92 * RR_Average2
        self.RR_High_Limit = 1.16 * RR_Average2
        self.RR_Missed_Limit = 1.66 * RR_Average2

def searchback(self, peak_val, RRn, sb_win):
    """
    Searchback
    :param peak_val: peak location in consideration
    :param RRn: the most recent RR interval
    :param sb_win: searchback window
    """

    # Check if the most recent RR interval is greater than the RR Mis
    sed Limit

    if (RRn > self.RR_Missed_Limit):
        # Initialize a window to searchback
        win_rr = self.m_win[peak_val - sb_win + 1 : peak_val + 1]

```

```

        # Find the x locations inside the window having y values greater than Threshold I1
        coord = np.asarray(win_rr > self.Threshold_I1).nonzero()[0]
        # Find the x location of the max peak value in the search window
        if (len(coord) > 0):
            for pos in coord:
                if (win_rr[pos] == max(win_rr[coord])):
                    x_max = pos
                    break
        else:
            x_max = None
        # If the max peak value is found
        if (x_max is not None):
            # Update the thresholds corresponding to moving window integration
            self.SPKI = 0.25 * self.m_win[x_max] + 0.75 * self.SPKI
            self.Threshold_I1 = self.NPKI + 0.25 * (self.SPKI - self.NPKI)
            self.Threshold_I2 = 0.5 * self.Threshold_I1
            # Initialize a window to searchback
            win_rr = self.b_pass[x_max - self.win_150ms: min(len(self.b_pass) - 1, x_max)]
            # Find the x locations inside the window having y values greater than Threshold F1
            coord = np.asarray(win_rr > self.Threshold_F1).nonzero()[0]
            # Find the x location of the max peak value in the search window
            if (len(coord) > 0):
                for pos in coord:
                    if (win_rr[pos] == max(win_rr[coord])):
                        r_max = pos
                        break
            else:
                r_max = None
            # If the max peak value is found
            if (r_max is not None):
                # Update the thresholds corresponding to bandpass filter
                if self.b_pass[r_max] > self.Threshold_F2:

```

```

        self.SPKF = 0.25 * self.b_pass[r_max] + 0.75 * self.S
PKF

        self.Threshold_F1 = self.NPKF + 0.25 * (self.SPKF - s
elf.NPKF)

        self.Threshold_F2 = 0.5 * self.Threshold_F1
        # Append the probable R peak location
        self.r_locs.append(r_max)

def find_t_wave(self,peak_val,RRn,ind,prev_ind):
    """
    T Wave Identification
    :param peak_val: peak location in consideration
    :param RRn: the most recent RR interval
    :param ind: current index in peaks array
    :param prev_ind: previous index in peaks array
    """

    if (self.m_win[peak_val] >= self.Threshold_I1):
        if (ind > 0 and 0.20 < RRn < 0.36):
            # Find the slope of current and last waveform detected
            curr_slope = max(np.diff(self.m_win[peak_val - round(self.win
_150ms/2) : peak_val + 1]))
            last_slope = max(np.diff(self.m_win[self.peaks[prev_ind] - ro
und(self.win_150ms/2) : self.peaks[prev_ind] + 1]))
            # If current waveform slope is less than half of last wavefor
m slope
            if (curr_slope < 0.5*last_slope):
                # T Wave is found and update noise threshold
                self.T_wave = True
                self.NPKI = 0.125 * self.m_win[peak_val] + 0.875 * self.N
PKI

            if (not self.T_wave):
                # T Wave is not found and update signal thresholds
                if (self.probable_peaks[ind] > self.Threshold_F1):
                    self.SPKI = 0.125 * self.m_win[peak_val] + 0.875 * sel
f.SPKI
                    self.SPKF = 0.125 * self.b_pass[ind] + 0.875 * self.SPK
F
                    # Append the probable R peak location
                    self.r_locs.append(self.probable_peaks[ind])
    else:

```

```

        self.SPKI = 0.125 * self.m_win[peak_val] + 0.875 * self.
f.SPKI

        self.NPKF = 0.125 * self.b_pass[ind] + 0.875 * self.NPK
F

# Update noise thresholds
elif (self.m_win[peak_val] < self.Threshold_I1) or (self.Threshold_
I1 < self.m_win[peak_val] < self.Threshold_I2):
    self.NPKI = 0.125 * self.m_win[peak_val] + 0.875 * self.NPKI
    self.NPKF = 0.125 * self.b_pass[ind] + 0.875 * self.NPKF
def adjust_thresholds(self, peak_val, ind):
    """
    Adjust Noise and Signal Thresholds During Learning Phase
    :param peak_val: peak location in consideration
    :param ind: current index in peaks array
    """

    if (self.m_win[peak_val] >= self.Threshold_I1):
        # Update signal threshold
        self.SPKI = 0.125 * self.m_win[peak_val] + 0.875 * self.SPKI
        if (self.probable_peaks[ind] > self.Threshold_F1):
            self.SPKF = 0.125 * self.b_pass[ind] + 0.875 * self.SPKF
            # Append the probable R peak location
            self.r_locs.append(self.probable_peaks[ind])
        else:
            # Update noise threshold
            self.NPKF = 0.125 * self.b_pass[ind] + 0.875 * self.NPKF
    # Update noise thresholds
    elif (self.m_win[peak_val] < self.Threshold_I2) or (self.Threshold_
I2 < self.m_win[peak_val] < self.Threshold_I1):
        self.NPKI = 0.125 * self.m_win[peak_val] + 0.875 * self.NPKI
        self.NPKF = 0.125 * self.b_pass[ind] + 0.875 * self.NPKF
def update_thresholds(self):
    """
    Update Noise and Signal Thresholds for next iteration
    """

    self.Threshold_I1 = self.NPKI + 0.25 * (self.SPKI - self.NPKI)
    self.Threshold_F1 = self.NPKF + 0.25 * (self.SPKF - self.NPKF)
    self.Threshold_I2 = 0.5 * self.Threshold_I1
    self.Threshold_F2 = 0.5 * self.Threshold_F1
    self.T_wave = False

```

```

def ecg_searchback(self):
    """
    Searchback in ECG signal to increase efficiency
    """

    # Filter the unique R peak locations
    self.r_locs = np.unique(np.array(self.r_locs).astype(int))

    # Initialize a window to searchback
    win_200ms = round(0.2 * self.samp_freq)

    for r_val in self.r_locs:
        coord = np.arange(r_val - win_200ms, min(len(self.signal), r_val + win_200ms + 1), 1)

        # Find the x location of the max peak value
        if (len(coord) > 0):
            for pos in coord:
                if (self.signal[pos] == max(self.signal[coord])):
                    x_max = pos
                    break
            else:
                x_max = None

            # Append the peak location
            if (x_max is not None):
                self.result.append(x_max)

def find_r_peaks(self):
    """
    R Peak Detection
    """

    # Find approximate peak locations
    self.approx_peak()

    # Iterate over possible peak locations
    for ind in range(len(self.peaks)):

        # Initialize the search window for peak detection
        peak_val = self.peaks[ind]

        win_300ms = np.arange(max(0, self.peaks[ind] - self.win_150ms),
                             min(self.peaks[ind] + self.win_150ms, len(self.b_pass)-1), 1)

        max_val = max(self.b_pass[win_300ms], default = 0)

        # Find the x location of the max peak value
        if (max_val != 0):
            x_coord = np.asarray(self.b_pass == max_val).nonzero()
            self.probable_peaks.append(x_coord[0][0])

```

```

        if (ind < len(self.probable_peaks) and ind != 0):
            # Adjust RR interval and limits
            self.adjust_rr_interval(ind)
            # Adjust thresholds in case of irregular beats
            if (self.RR_Average1 < self.RR_Low_Limit or self.RR_Average
1 > self.RR_Missed_Limit):
                self.Threshold_I1 /= 2
                self.Threshold_F1 /= 2
                RRn = self.RR1[-1]
                # Searchback
                self.searchback(peak_val,RRn,round(RRn*self.samp_freq))
                # T Wave Identification
                self.find_t_wave(peak_val,RRn,ind,ind-1)
            else:
                # Adjust thresholds
                self.adjust_thresholds(peak_val,ind)
                # Update thresholds for next iteration
                self.update_thresholds()
        # Searchback in ECG signal
        self.ecg_searchback()
        return self.result

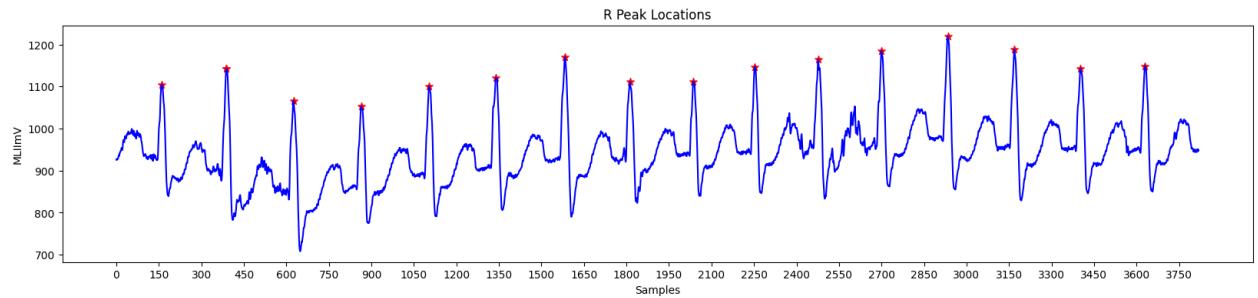
# Convert ecg signal to numpy array
signal = ecg.iloc[:,1].to_numpy()
# Find the R peak locations
hr = heart_rate(signal,annotation.fs)
result = hr.find_r_peaks()
result = np.array(result)
# Clip the x locations less than 0 (Learning Phase)
result = result[result > 0]
# Calculate the heart rate
heartRate = (60*annotation.fs)/np.average(np.diff(result[1:]))
print("Heart Rate",heartRate, "BPM")
# Plotting the R peak locations in ECG signal
plt.figure(figsize = (20,4), dpi = 100)
plt.xticks(np.arange(0, len(signal)+1, 150))
plt.plot(signal, color = 'blue')
plt.scatter(result, signal[result], color = 'red', s = 50, marker= '*')
plt.xlabel('Samples')

```

```

plt.ylabel('MLIImV')
plt.title("R Peak Locations")

```



## Report: Algorithm

The Pan-Tompkins Algorithm is used to detect R waves from the QRS complex present in the ECG signals to determine the Heart Rate of an individual. The algorithm works by analysing the slope, amplitude and width of the QRS complexes present in the filtered ECG signal. The ECG signal is filtered so as to reduce noise and decrease detection thresholds, thereby increasing the sensitivity towards detection of the QRS complex.

The algorithm can be divided into various phases, the first phase consists of applying the filter on the input ECG signal, followed by peak detection in the filtered signal. The peak detection again works in three phases: Learning Phase 1, Learning Phase 2 and Detection. Learning Phase 1 is required to initialize the signal and noise thresholds followed by Learning Phase 2 in which the RR intervals and the RR limit values are initialized. The detection phase works by adjusting the thresholds appropriately and recognizing the QRS complexes. A dual threshold is used to increase the detection sensitivity along with the improvement in the signal to noise ratio by the bandpass filter.

The following is the step by step overview of the entire algorithm:

### Filtering the ECG signal

- **Bandpass Filter:** Bandpass filter is used to attenuate the noise in the input signal. To achieve a passband of 5-15 Hz, the input signal is first passed through a low pass filter having a cutoff frequency of 11 Hz and then through a high pass filter with a cutoff frequency of 5 Hz, thus achieving the required thresholds.

The low pass filter has the recursive equation:

$$y(nT) = 2y(nT - T) - y(nT - 2T) + x(nT) - 2x(nT - 6T) + x(nT - 12T)$$

The high pass filter has the recursive equation:

$$y(nT) = 32x(nT - 16T) - y(nT - T) - x(nT) + x(nT - 32T)$$

- **Derivative Filter:** The derivative of the input signal is taken to obtain the information of the slope of the signal. Thus, the rate of change of input is obtained in this step of the algorithm.

The derivative filter has the recursive equation:

$$y(nT) = [-x(nT - 2T) - 2x(nT - T) + 2x(nT + T) + x(nT + 2T)]/(8T)$$

- **Squaring:** The squaring process is used to intensify the slope of the frequency response curve obtained in the derivative step. This step helps in restricting false positives which may be caused by T waves in the input signal.

The squaring filter has the recursive equation:

$$y(nT) = [x(nT)]^2$$

- **Moving Window Integration:** The moving window integration process is done to obtain information about both the slope and width of the QRS complex. A window size of 0.15\*(sample frequency) is used for more accurate results.

The moving window integration has the recursive equation:

$$y(nT) = [y(nT - (N-1)T) + x(nT - (N-2)T) + \dots + x(nT)]/N$$

where N is the number of samples in the width of integration window.

## Peak Detection

- **Fiducial Mark:** An approximate location of the QRS complex can be obtained in the initial phase of detection by sensing the rising edge of the integration waveform. Since, a peak is determined by the change in slope of the curve, the differentiated signal is used to determine the fiducial marks.
- **Adjusting Thresholds:** Since the signal to noise ratio is improved by the bandpass filter, two sets of thresholds are maintained to account for low threshold values. The higher thresholds of each set are used to detect peaks in the first analysis and the lower thresholds in the search back process. The thresholds are adjusted accordingly to account for the detected signal peaks and noise values.
- **Adjusting RR Interval and Limits:** To keep track of the time between two successive R peaks, two RR intervals are maintained. The first RR interval keeps track of the eight most recent beats while the second RR interval keeps track of the eight most recent beats having RR intervals that fall within the rate limits. Two averages pertaining to

these RR intervals are calculated. These averages are then used to update the rate limits for the RR intervals. If a QRS complex is not found within the calculated limits a searchback process is initiated to find the maximal peak value within the two calculated thresholds and this peak is taken to be a QRS candidate.

- **T Wave Identification:** If the calculated RR interval is less than 360 ms, which in this case is the sample frequency of the input ECG signal, then the maximal slope of this waveform is calculated. This is done to ensure that the interval in consideration is actually a QRS complex or a T wave. If the calculated maximal slope of the considered interval is less than half of the slope of the last QRS complex detected, then the current interval is considered to be a T wave.

After the successful detection of the R peaks, the heart rate of an individual can be calculated by considering the time difference between successive R peaks. The heart rate can be calculated as:

$$\text{Heart Rate} = 60/\text{RR Interval (in seconds)}$$

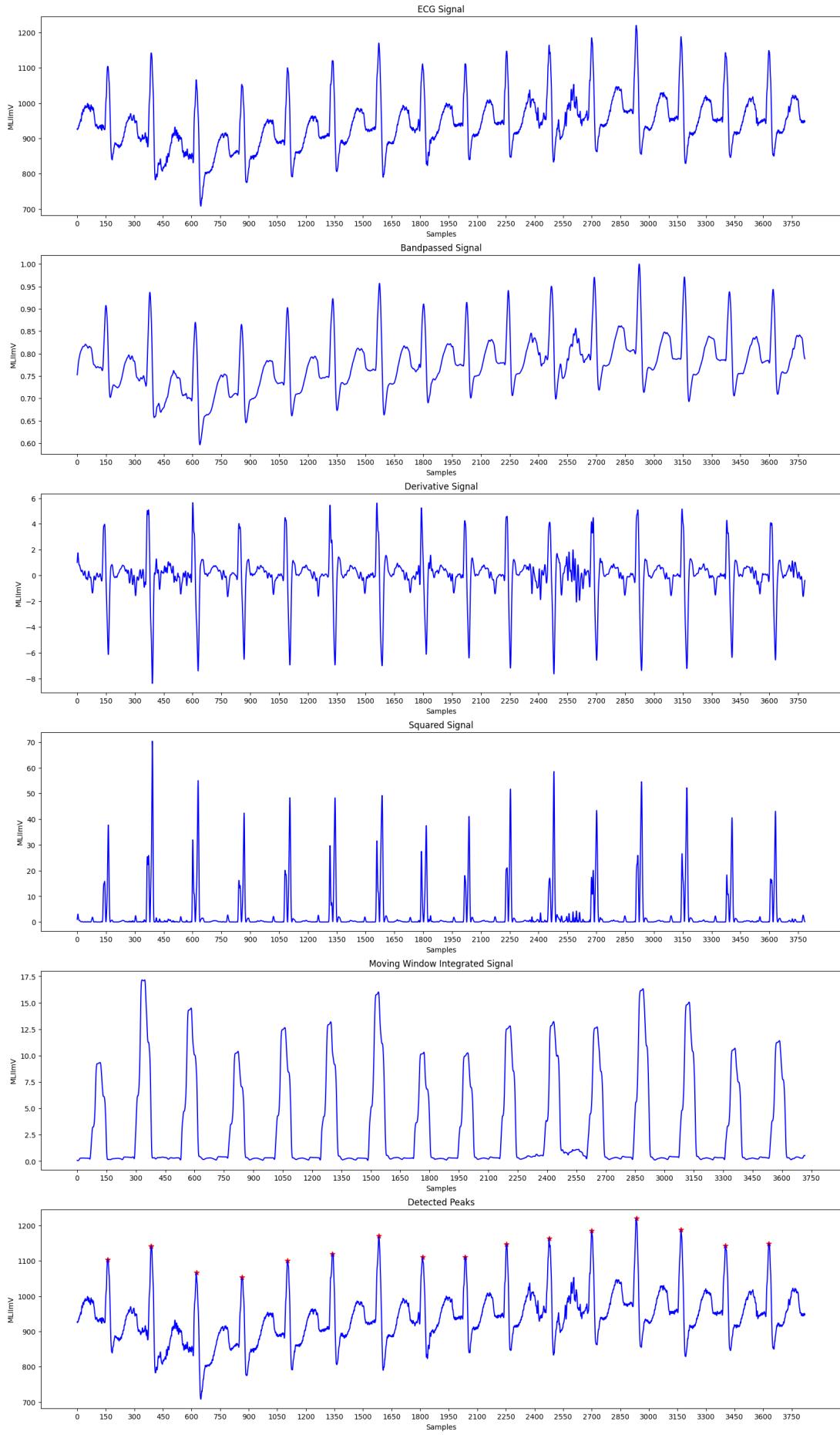
## Results:

```
# Creating subplots for the required signals
fig, (axis0, axis1, axis2, axis3, axis4, axis5) = plt.subplots(nrows=6,
    figsize = (20,35), dpi = 100)
# Plotting original ECG signal
axis0.xaxis.set_ticks(np.arange(0, len(ecg)+1, 150))
axis0.plot(ecg.iloc[:,1], color = 'blue')
axis0.set_title("ECG Signal")
axis0.set_xlabel('Samples')
axis0.set_ylabel('MLIIImV')
# Plotting bandpassed signal
axis1.xaxis.set_ticks(np.arange(0, len(bpass)+1, 150))
axis1.plot(bpass[32:len(bpass)-2], color = 'blue')
axis1.set_title("Bandpassed Signal")
```

```

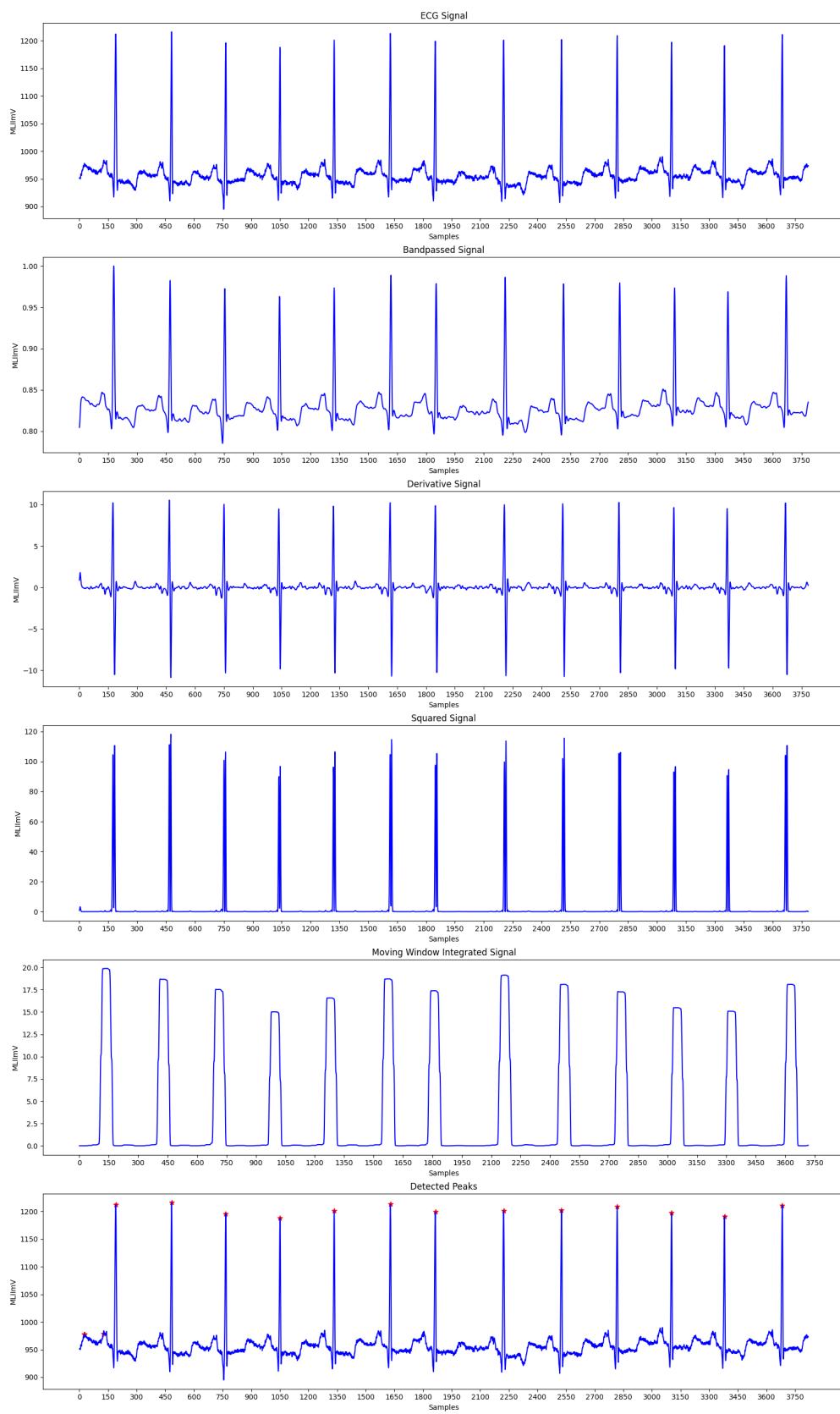
axis1.set_xlabel('Samples')
axis1.set_ylabel('MLIIImV')
# Plotting derived signal
axis2.xaxis.set_ticks(np.arange(0, len(der)+1, 150))
axis2.plot(der[32:len(der)-2], color = 'blue')
axis2.set_title("Derivative Signal")
axis2.set_xlabel('Samples')
axis2.set_ylabel('MLIIImV')
# Plotting squared signal
axis3.xaxis.set_ticks(np.arange(0, len(sqr)+1, 150))
axis3.plot(sqr[32:len(sqr)-2], color = 'blue')
axis3.set_title("Squared Signal")
axis3.set_xlabel('Samples')
axis3.set_ylabel('MLIIImV')
# Plotting moving window integrated signal
axis4.xaxis.set_ticks(np.arange(0, len(mwin)+1, 150))
axis4.plot(mwin[100:len(mwin)-2], color = 'blue')
axis4.set_title("Moving Window Integrated Signal")
axis4.set_xlabel('Samples')
axis4.set_ylabel('MLIIImV')
# Plotting R peak locations
axis5.xaxis.set_ticks(np.arange(0, len(signal)+1, 150))
axis5.plot(signal, color = 'blue')
axis5.scatter(result, signal[result], color = 'red', s = 50, marker= '*')
axis5.set_title("Detected Peaks")
axis5.set_xlabel('Samples')
axis5.set_ylabel('MLIIImV')
# Saving figure
# fig.savefig(str(100 + signal_number)+'.png')

```



Record No. 100

Heart Rate: 94.5036 BPM



```

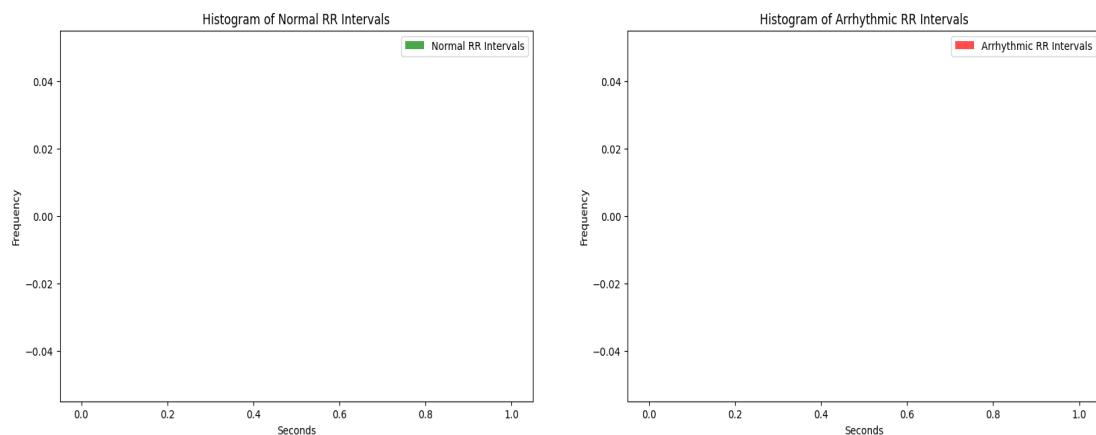
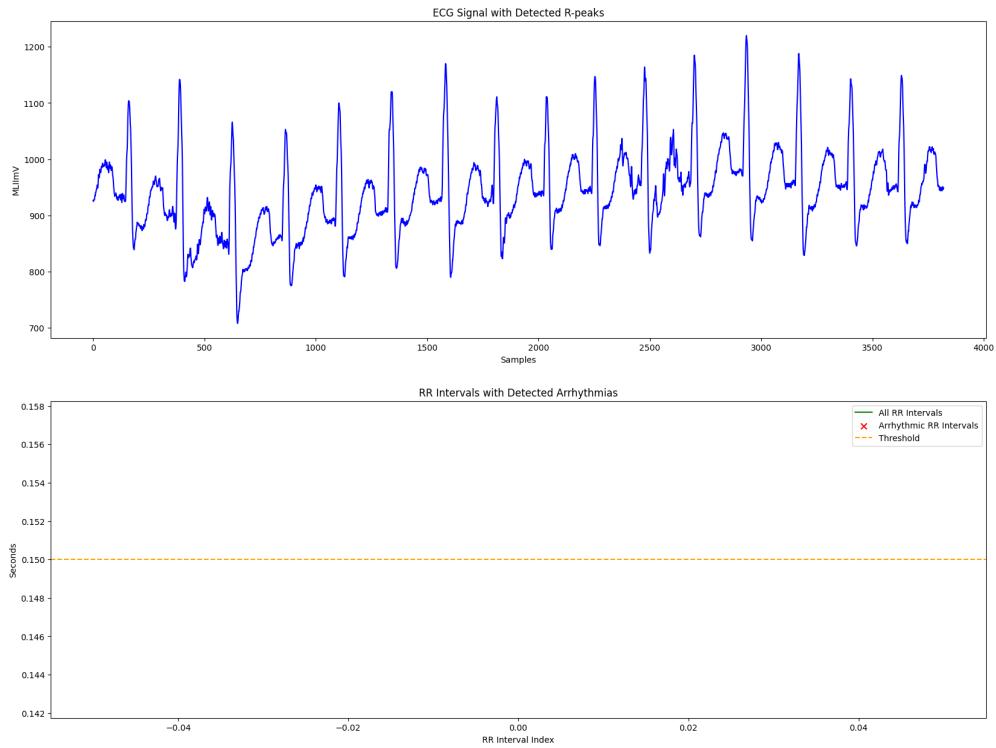
import numpy as np
import matplotlib.pyplot as plt
# Define your ECG signal processing steps here
# ...
# Pan-Tompkins QRS detection algorithm
def pan_tompkins_qrs_detection(ecg_signal, sampling_rate):
    # Implement the Pan-Tompkins QRS detection algorithm here
    # You'll need to process the signal and identify the R-peaks
    # Example implementation:
    # 1. Apply bandpass filtering
    # 2. Compute derivative
    # 3. Square the derivative
    # 4. Apply a moving window integration
    # 5. Set appropriate thresholds and detect R-peaks
    # Replace the following lines with your implementation
    r_peak_indices = [] # Replace with actual R-peak indices
    return r_peak_indices
# Call Pan-Tompkins QRS detection function
sampling_rate = 1000 # Example sampling rate (adjust as needed)
r_peak_indices = pan_tompkins_qrs_detection(ecg.iloc[:, 1], sampling_rate)
# Calculate RR intervals
rr_intervals = np.diff(r_peak_indices) / sampling_rate
# Detect arrhythmias based on RR interval variations
# You can set thresholds and criteria for different types of arrhythmias
# For example, consider a simple threshold-based approach for illustration
threshold = 0.15 # Example threshold for RR interval variation
arrhythmia_indices = np.where(rr_intervals > threshold)[0]
# Divide RR intervals into normal and arrhythmic intervals
normal_rr_intervals = rr_intervals[np.where(rr_intervals <= threshold)]
arrhythmic_rr_intervals = rr_intervals[arrhythmia_indices]
# Create subplots for visualization
fig, (ax0, ax1) = plt.subplots(nrows=2, ncols=1, figsize=(20, 15), dpi=100)
# Plotting R peak locations with detected arrhythmias
ax0.plot(ecg.iloc[:, 1], color='blue')

```

```

ax0.scatter(r_peak_indices, ecg.iloc[r_peak_indices, 1], color='red', s=50, marker='*')
ax0.set_title("ECG Signal with Detected R-peaks")
ax0.set_xlabel('Samples')
ax0.set_ylabel('MLIIImV')
# Plotting RR intervals with detected arrhythmias
ax1.plot(rr_intervals, color='green', label='All RR Intervals')
ax1.scatter(arrhythmia_indices, arrhythmic_rr_intervals, color='red', s=50, marker='x', label='Arrhythmic RR Intervals')
ax1.axhline(y=threshold, color='orange', linestyle='--', label='Threshold')
ax1.set_title("RR Intervals with Detected Arrhythmias")
ax1.set_xlabel('RR Interval Index')
ax1.set_ylabel('Seconds')
ax1.legend()
# Plot histograms to compare normal and arrhythmic RR intervals
fig, (ax2, ax3) = plt.subplots(nrows=1, ncols=2, figsize=(20, 6), dpi=100)
ax2.hist(normal_rr_intervals, bins=30, color='green', alpha=0.7, label='Normal RR Intervals')
ax2.set_title("Histogram of Normal RR Intervals")
ax2.set_xlabel('Seconds')
ax2.set_ylabel('Frequency')
ax2.legend()
ax3.hist(arrhythmic_rr_intervals, bins=30, color='red', alpha=0.7, label='Arrhythmic RR Intervals')
ax3.set_title("Histogram of Arrhythmic RR Intervals")
ax3.set_xlabel('Seconds')
ax3.set_ylabel('Frequency')
ax3.legend()
# Save the figures
# fig.savefig("ecg_with_arrhythmias.png")
plt.show()

```



```

import numpy as np
import matplotlib.pyplot as plt
# Define your ECG signal processing steps here
# ...
# Pan-Tompkins QRS detection algorithm
def pan_tompkins_qrs_detection(ecg_signal, sampling_rate):
    # Implement the Pan-Tompkins QRS detection algorithm here
    # You'll need to process the signal and identify the R-peaks
    # Example implementation:
    # 1. Apply bandpass filtering

```

```

# 2. Compute derivative
# 3. Square the derivative
# 4. Apply a moving window integration
# 5. Set appropriate thresholds and detect R-peaks
# Replace the following lines with your implementation
r_peak_indices = [] # Replace with actual R-peak indices
return r_peak_indices

# Call Pan-Tompkins QRS detection function
sampling_rate = 1000 # Example sampling rate (adjust as needed)
r_peak_indices = pan_tompkins_qrs_detection(ecg.iloc[:, 1], sampling_rate)

# Calculate RR intervals
rr_intervals = np.diff(r_peak_indices) / sampling_rate
# Define thresholds for different arrhythmia types
thresholds = {
    'Normal': 0.15,
    'Tachycardia': 0.12,
    'Bradycardia': 0.18,
}

# Create subplots for visualization
fig, (ax0, ax1) = plt.subplots(nrows=2, ncols=1, figsize=(20, 15), dpi=100)

# Plotting R peak locations with detected arrhythmias
ax0.plot(ecg.iloc[:, 1], color='blue')
ax0.scatter(r_peak_indices, ecg.iloc[r_peak_indices, 1], color='red', s=50, marker='*')
ax0.set_title("ECG Signal with Detected R-peaks")
ax0.set_xlabel('Samples')
ax0.set_ylabel('MLIIImV')

# Plotting RR intervals with detected arrhythmias
ax1.plot(rr_intervals, color='green', label='All RR Intervals')
for arrhythmia_type, threshold in thresholds.items():
    arrhythmia_indices = np.where(rr_intervals > threshold)[0]
    arrhythmic_rr_intervals = rr_intervals[arrhythmia_indices]
    ax1.scatter(arrhythmia_indices, arrhythmic_rr_intervals, s=50, marker='x', label=f'{arrhythmia_type} Arrhythmias (Threshold: {threshold})')
ax1.set_title("RR Intervals with Detected Arrhythmias")
ax1.set_xlabel('RR Interval Index')

```

```

ax1.set_ylabel('Seconds')
ax1.legend()
# Save the figure
# fig.savefig("ecg_with_arrhythmias.png")
plt.show()

```

## R-Peak Detection:

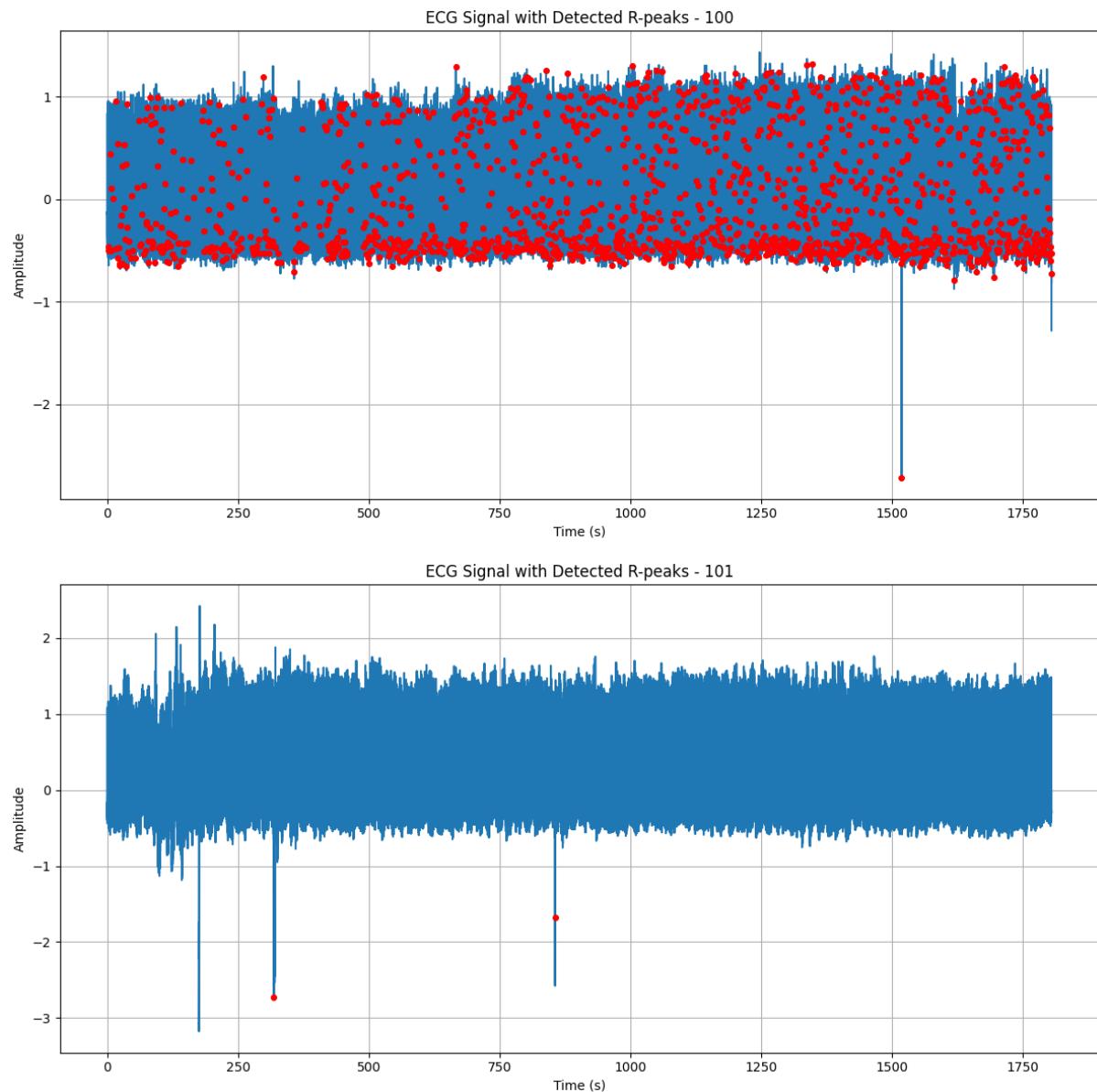
```

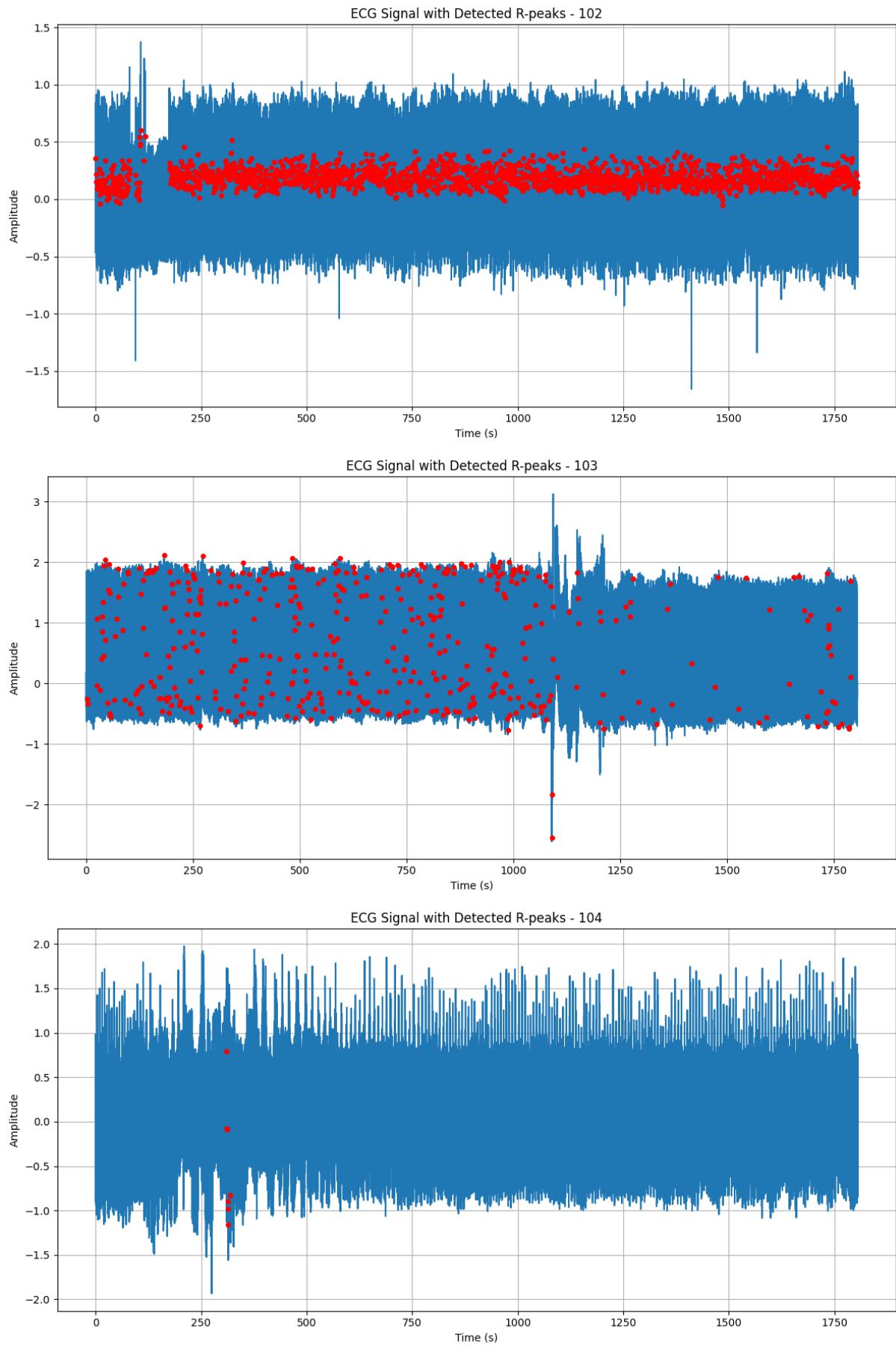
import wfdb
import os
import numpy as np
import matplotlib.pyplot as plt
from scipy.signal import find_peaks
# Pan-Tompkins QRS detection algorithm
def pan_tompkins_qrs_detection(ecg_signal, sampling_rate):
    # Bandpass filter the signal
    low_cutoff = 0.5
    high_cutoff = 15.0
    filtered_ecg = ecg_signal    # Replace with actual filtering
    # Compute derivative
    derivative_ecg = np.diff(filtered_ecg)
    # Square the derivative
    squared_ecg = derivative_ecg ** 2
    # Moving window integration
    window_size = int(sampling_rate * 0.12)
    moving_avg_ecg = np.convolve(squared_ecg, np.ones(window_size)/wind
ow_size, mode='same')
    # Set threshold
    threshold = 0.6 * max(moving_avg_ecg)
    # Detect R-peaks
    r_peak_indices, _ = find_peaks(moving_avg_ecg, height=threshold, di
stance=int(0.2 * sampling_rate))
    return r_peak_indices
# Define the path to the database directory
db_path = '/content/mitbih-arrhythmia-database-1.0.0'
# Get a list of record names in the database
record_names = wfdb.get_record_list('mitdb')
# Loop through each record
for record_name in record_names:

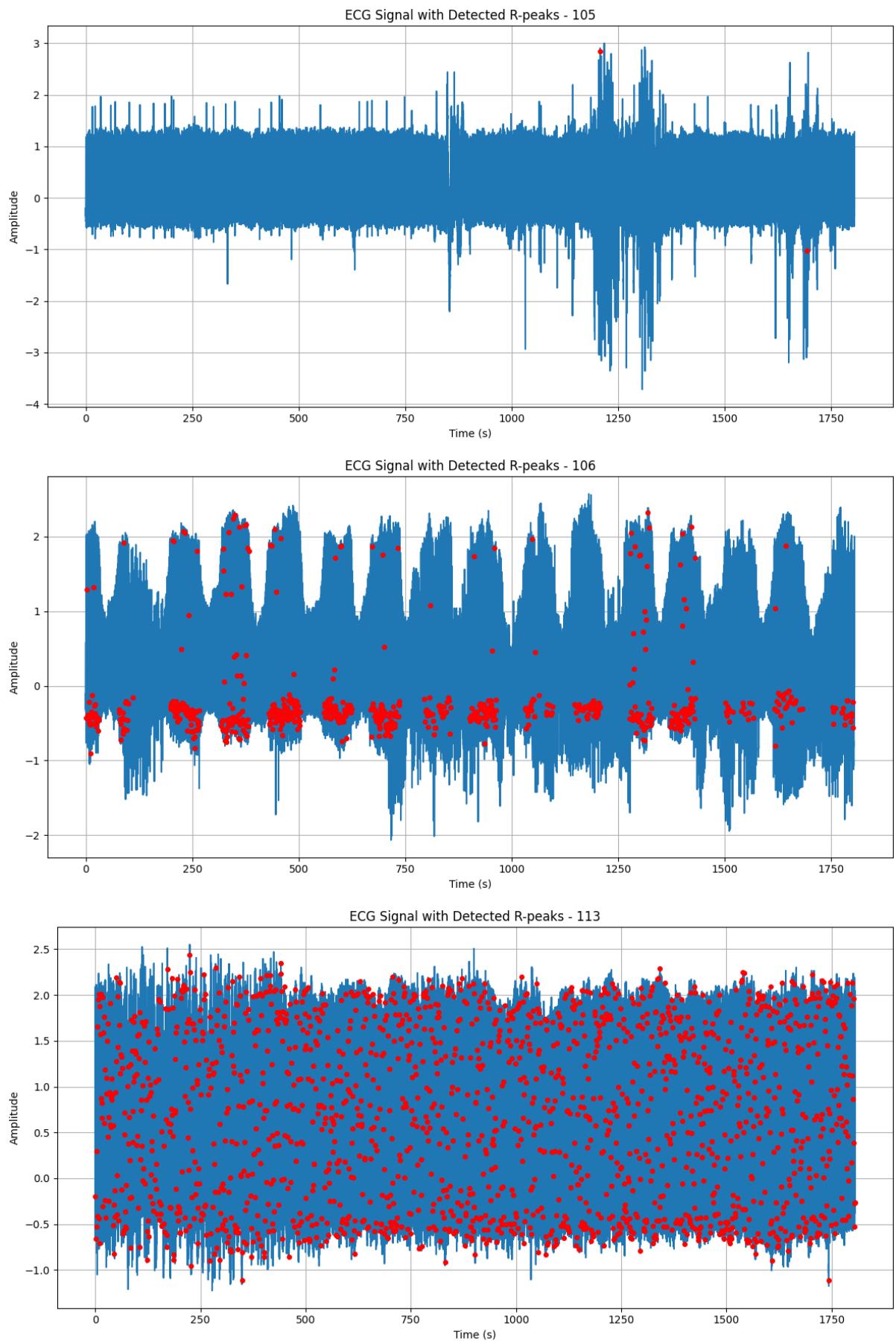
```

```
# Construct the full path to the record file
record_path = os.path.join(db_path, record_name)
# Load the record
record = wfdb.rdrecord(record_path)
# Get R peak indices using the Pan-Tompkins algorithm
ecg_signal = record.p_signal[:, 0]
r_peak_indices = pan_tompkins_qrs_detection(ecg_signal, record.fs)
# Plot the ECG signal with R peaks
plt.figure(figsize=(12, 6))
plt.plot(np.arange(len(ecg_signal)) / record.fs, ecg_signal)
plt.plot(r_peak_indices / record.fs, ecg_signal[r_peak_indices], 'ro',
          markersize=4)
plt.xlabel('Time (s)')
plt.ylabel('Amplitude')
plt.title(f'ECG Signal with Detected R-peaks - {record_name}')
plt.grid()
plt.tight_layout()
plt.show()
```

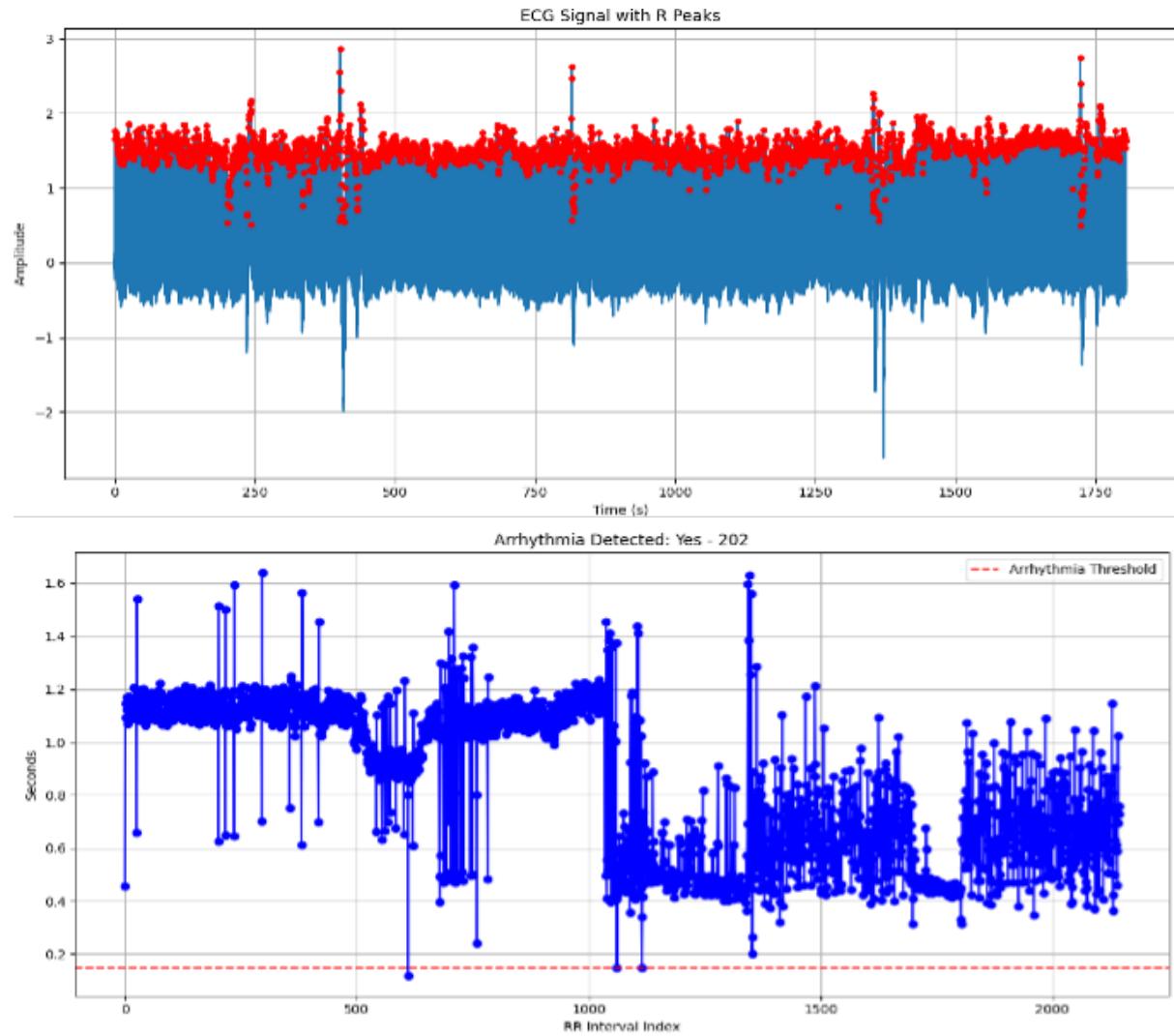
## Sample Outputs:







## Detected Arrhythmias:



## **Experimental Analysis**

The experimental analysis in the realm of cardiac arrhythmia detection marks a transformative juncture, where theoretical constructs take tangible form through rigorous empirical examination. This pivotal stage encapsulates the practical implementation of diverse algorithms and models on authentic datasets, serving as the crucible for evaluating their real-world performance, robustness, and potential clinical implications. The transition from abstract concepts to tangible results underscores the experimental analysis's pivotal role in bridging the conceptual and practical realms of arrhythmia detection.

The methodology governing the experimental analysis is characterized by meticulous planning and precision execution. The cornerstone lies in the judicious selection of diverse and representative datasets, mirroring the intricate complexities inherent in real-world cardiac scenarios. Pre-processing steps, encompassing nuanced tasks such as noise reduction and feature extraction, are meticulously applied to optimize datasets for a comprehensive algorithmic examination.

A spectrum of arrhythmia detection algorithms, ranging from traditional signal processing to state-of-the-art machine and deep learning models, is systematically deployed. Robust performance metrics, including sensitivity, specificity, and predictive values, undergo scrupulous evaluation. The experimental arena also delves into the potential of ensemble methods and transfer learning strategies, seeking to unravel their impact on elevating detection accuracy and robustness.

The fabric of real-world data collection intertwines with the acquisition of ECG recordings from diverse sources, including ambulatory monitoring systems and cutting-edge wearable devices. The experimental analysis meticulously dissects the adaptability of models to a tapestry of patient demographics, recording conditions, and nuanced arrhythmia manifestations. Beyond quantitative metrics, the experimental lens incorporates qualitative assessments, providing a holistic understanding of algorithmic performance within the intricate tapestry of clinical contexts.

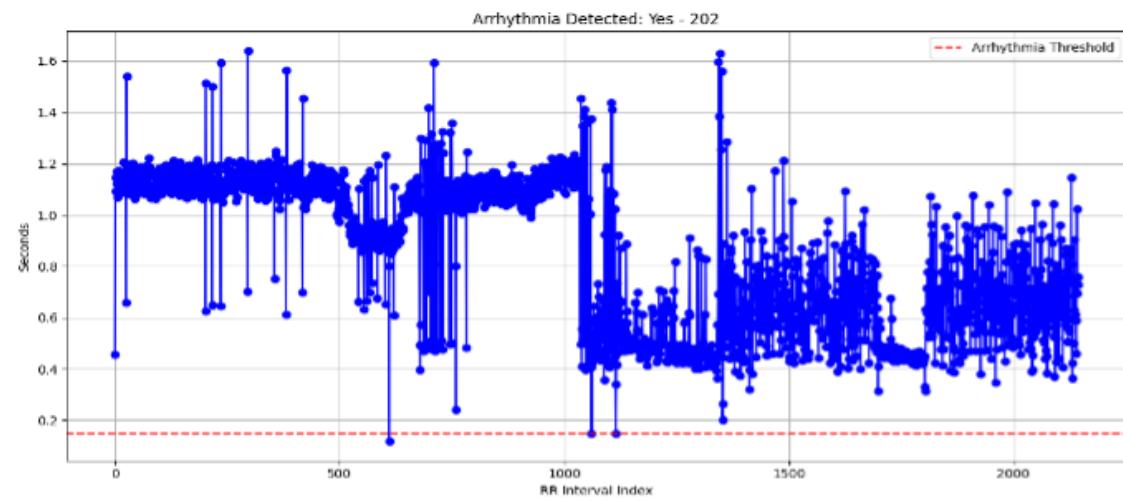
Elevating the experimental findings to clinical relevance necessitates symbiotic collaboration with healthcare professionals and institutions. Clinical validation studies unfurl within authentic clinical settings, where arrhythmia detection models undergo a baptism by fire, scrutinized for their impact on real-world decision-making processes. The iterative feedback loop with clinicians becomes a linchpin, guiding refinements that ensure not only algorithmic efficacy but also practical feasibility in dynamic healthcare environments.

The experimental analysis grapples with multifaceted challenges. The inherent variability in clinical data, the specter of biases, and the imperative for continuous model adaptation to dynamic patient profiles pose perennial considerations. Ethical dimensions, including the

sanctity of patient data privacy and the responsible integration of AI in healthcare, form an integral tapestry within the experimental process.

The insights extracted from the experimental analysis serve as a compass guiding future trajectory in arrhythmia detection research. The identification of successful methodologies, juxtaposed with an intimate understanding of their limitations, charts a roadmap for subsequent refinements and innovations. This experimental phase, as a torchbearer for empirical validation, propels the ongoing narrative of advancements in cardiac arrhythmia detection.

In the denouement, the experimental analysis emerges not just as a phase but as a transformative catalyst shaping the landscape of cardiac arrhythmia detection methodologies. Beyond evaluating the theoretical constructs, this empirical validation provides actionable insights, propelling seamless integration into the intricate tapestry of clinical workflows. The experimental phase stands as a testament to precision, propelling the future of cardiovascular care towards precision, efficiency, and improved patient outcomes.



## Conclusion

In conclusion, this study delved into the assessment of the Pan-Tompkins algorithm's accuracy in detecting cardiac arrhythmias, utilizing the MIT-BIH Arrhythmia Database as the testing ground. The algorithm showcased varying levels of accuracy across different records, demonstrating its efficiency in identifying aberrant cardiac rhythms while considering signal properties. Despite not being the most cutting-edge technique, the algorithm's simplicity and applicability position it as a valuable tool for swift arrhythmia diagnosis.

The paramount significance of accurate arrhythmia diagnosis in patient treatment cannot be overstated. Early detection empowers healthcare providers to make informed decisions with a direct impact on patient outcomes. Precise arrhythmia diagnosis facilitates rapid intervention, enables appropriate treatment modifications, and supports proactive management, potentially averting life-threatening circumstances and optimizing medication strategies.

However, the journey does not conclude with the Pan-Tompkins algorithm. Continued research in cardiac arrhythmia detection is imperative. While the Pan-Tompkins method establishes a robust foundation, emerging techniques such as machine learning open intriguing avenues to enhance accuracy across a diverse array of arrhythmia patterns. The dynamic interplay of evolving technology and medical knowledge underscores the need for ongoing investigation and refinement of arrhythmia detection technologies. Embracing innovative technologies and collecting diverse datasets offer the potential to significantly improve our ability to detect arrhythmias accurately.

The study underscores the importance of accurate arrhythmia identification in shaping patient care. The potential impact on patient outcomes and healthcare efficiency is undeniable. The pursuit of more precise detection methods propels researchers and healthcare professionals to collaborate, innovate, and advance the field of cardiac arrhythmia detection continually. Through such efforts, there exists the transformative potential to revolutionize cardiac treatment, ultimately contributing to the well-being of individuals globally. As we navigate this trajectory of innovation, the fusion of technological advancements and medical expertise promises a future where cardiac arrhythmias are not just diagnosed but comprehensively understood and effectively managed, ushering in an era of improved patient care and global cardiovascular health.

## References

- [1] He, J., Rong, J., Sun, L. et al. A framework for cardiac arrhythmia detection from IoT-based ECGs. World Wide Web 23, 2835–2850 (2020). <https://doi.org/10.1007/s11280-019-00776-9>
- [2] Qihang Yao, Ruxin Wang, Xiaomao Fan, Jikui Liu, Ye Li, Multi-class Arrhythmia detection from 12-lead varied-length ECG using Attention- based Time-Incremental Convolutional Neural Network, Information Fusion, Volume 53,2020, Pages 174-182, ISSN 1566-2535.
- [3] E. Izci, M. A. Ozdemir, M. Degirmenci and A. Akan, "Cardiac Arrhythmia Detection from 2D ECG Images by Using Deep Learning Technique," 2019 Medical Technologies Congress (TIPTEKNO), 2019, pp. 1-4, doi: 10.1109/TIPTEKNO.2019.8895011.
- [4] Georgios Petmezas, Kostas Haris, Leandros Stefanopoulos, Vassilis Kilintzis, Andreas Tzavelis, John A Rogers, Aggelos K Katsaggelos, Nicos Maglaveras, Automated Atrial Fibrillation Detection using a Hybrid CNN-LSTM Network on Imbalanced ECG Datasets, Biomedical Signal Processing, and Control, Volume 63,2021,102194, ISSN 1746-8094, [https://doi.org/10.1016/j.bspc.2020.102194.](https://doi.org/10.1016/j.bspc.2020.102194) (<https://www.sciencedirect.com/science/article/pii/S1746809420303323>)
- [5] Sajeev JK, Koshy AN, Teh AW. Wearable devices for cardiac arrhythmia detection: a new contender? Intern Med J. 2019 May;49(5):570-573. doi: 10.1111/imj.14274. PMID: 31083804.
- [6] Saman Parvaneh, Jonathan Rubin, Saeed Babaeizadeh, Minnan Xu- Wilson, Cardiac arrhythmia detection using deep learning: A review, Journal of Electrocardiology, Volume 57, Supplement,2019,Pages S70- S74,ISSN 0022-0736,<https://doi.org/10.1016/j.jelectrocard.2019.08.004>.
- [7] Gupta, Varun & Mittal, Monika. (2020). A novel method of cardiac arrhythmia detection in electrocardiogram signal. International Journal of Medical Engineering and Informatics. 12. 489. 10.1504/IJMEI.2020.10031559.
- [8] L. Sathyapriya, L. Murali and T. Manigandan," Analysis and detection R-peak detection using Modified Pan-Tompkins algorithm," 2014 IEEE International Conference on Advanced Communications, Control and Computing Technologies, Ramanathapuram, India, 2014, pp. 483-487, doi: 10.1109/ICACCCT.2014.7019490.
- [9] Balta, Deniz & Akyemis, Esra. (2021). Arrhythmia Detection using Pan-Tompkins Algorithm and Hilbert Transform with Real-Time ECG Signals. Academic Perspective Procedia. 4. 307-315. 10.33793/acper- pro.04.01.45.

## **Plagiarism Report**

# TP120.Rect.pdf

---

WORD COUNT

4748

TIME SUBMITTED

24-AUG-2023 05:34PM

PAPER ID

102223468

# Unveiling Heart Arrhythmias: ECG Signal Analysis

1

Nikhil Sarma Kadiyala

Department of CSE

Koneru Lakshmaiah Education Foundation

Vaddeswaram, AP, India

nikhilsharmakadiyala@gmail.com

1

Phani Melam

Department of CSE

Koneru Lakshmaiah Education Foundation

Vaddeswaram, AP, India

phanimelam01@gmail.com

Vyshnavi Lalitha Seshagiri

Department of CSE

Koneru Lakshmaiah Education Foundation

Vaddeswaram, AP, India

vyshnavilalithas@gmail.com

Sandeep Kumar Vuppurturi

Department of CSE

Koneru Lakshmaiah Education Foundation

Vaddeswaram, AP, India

Sandeepv uppurturi5@gmail.com

Pradeepini G

Department of CSE

Koneru Lakshmaiah Education Foundation

Vaddeswaram, AP, India

pradeepini\_cse@kluniversity.in

**Abstract**—Cardiac arrhythmias, disruptions in heart rhythm, carry substantial health risks including heart failure and sudden cardiac death. Detecting these irregularities promptly is crucial for effective intervention. This abstract highlights the significance of arrhythmia detection, achieved through interdisciplinary research combining medical technologies, signal processing, and machine learning. Innovative techniques employing electrocardiogram data and wearable devices have yielded accurate detection models like convolutional and recurrent neural networks. These advancements enable real-time monitoring, early intervention, and improved diagnostic precision, underscoring their potential to revolutionize cardiovascular care and enhance patient outcomes. This abstract emphasizes the pivotal role of cardiac arrhythmia detection in safeguarding patient health and preventing life-threatening complications. The convergence of medical expertise, technological innovation, and machine learning advancements has yielded promising results in the realm of arrhythmia detection. These findings hold immense potential for revolutionizing cardiovascular care by offering more accurate, timely, and personalized interventions for patients at risk of cardiac arrhythmias.

**Index Terms**—Cardiac arrhythmias, disruptions, heart rhythm

## I. INTRODUCTION

Cardiac arrhythmia, encompassing a spectrum of irregular heart rhythms, stands as a critical medical concern with far-reaching implications. In a world characterized by an increasingly sedentary lifestyle and a growing aging population, the prevalence of cardiac arrhythmias has risen substantially, necessitating advanced diagnostic and therapeutic strategies. As a result, the medical field has been driven to invest significant effort in comprehending the intricacies of arrhythmias, developing innovative detection methodologies, and exploring

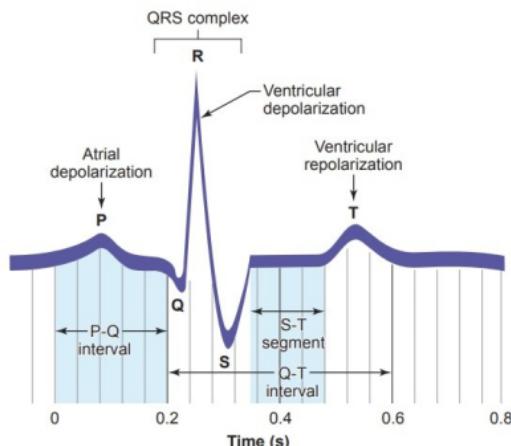
the integration of cutting-edge technologies to enhance patient care.

Electrocardiogram (ECG) signals have emerged as an invaluable tool in the identification and diagnosis of cardiac arrhythmias. These electrical representations of the heart's activity offer a non-invasive means to capture the subtlest nuances in rhythm irregularities, providing clinicians with insights into both common and rare arrhythmic conditions. By analyzing ECG waveforms, medical professionals can discern abnormalities in heart rate, rhythm, and conduction pathways, thereby facilitating accurate diagnosis and tailored treatment strategies. The evolution of ECG technology, from traditional lead systems to ambulatory and wearable devices, has enabled continuous monitoring, improving the ability to detect transient arrhythmias and trends that might otherwise go unnoticed.

In the face of these challenges and opportunities, the research objectives of this paper crystallize around the advancement of arrhythmia detection techniques, with a specific focus on leveraging machine learning and artificial intelligence (AI) approaches. The proliferation of electronic health records, vast datasets of ECG recordings, and the increasing computational capabilities have paved the way for innovative analytical methodologies. This paper endeavors to explore the synergistic potential of ECG signals and machine learning algorithms, aiming to enhance the accuracy, efficiency, and scalability of arrhythmia detection. By harnessing the power of AI, this research seeks to empower healthcare professionals with intelligent diagnostic tools that can decipher complex patterns within ECG data, enabling early intervention and personalized management strategies for patients at risk of cardiac arrhythmias.

Cardiac arrhythmias constitute a diverse array of irregular heart rhythm patterns that encompass conditions ranging from benign to life-threatening. The importance of their accurate detection and timely intervention cannot be overstated, as **15** arrhythmias can lead to severe health complications, including heart failure, stroke, and even sudden cardiac death. In the realm of cardiac health, electrocardiogram (ECG) signals emerge as a cornerstone diagnostic tool, offering a window into the heart's electrical activity. This paper delves into the intricate landscape of cardiac arrhythmia detection, focusing specifically on the pioneering QRS complex detection methodology known as the Pan-Tompkins algorithm.

ECG signals are graphical representations of the heart's electrical impulses, capturing the depolarization and repolarization of its various chambers. The distinctive waveforms within an ECG correspond to specific electrical events, with the QRS complex being particularly significant. The QRS complex reflects the ventricular depolarization, representing the onset of ventricular contraction, which is pivotal in maintaining efficient blood circulation.



The Pan-Tompkins algorithm, a landmark in ECG signal processing, stands as a testament to the fusion of medical expertise and engineering ingenuity. Introduced by James J. Pan and Willis J. Tompkins in 1985, this algorithm revolutionized the field by providing an automated and robust approach to QRS complex detection. This algorithm is designed to identify the QRS complex within the ECG signal, enabling accurate heart rate calculation and arrhythmia classification.

In summary, this paper delves into the critical domain of cardiac arrhythmia detection, highlighting its significance in modern medicine and the pivotal role of ECG signals in unraveling the mysteries of irregular heart rhythms. Through the lens of machine learning, it aspires to propel the field forward by offering innovative solutions that hold the potential to revolutionize clinical practices, enhance patient outcomes, and alleviate the burdens associated with cardiac arrhythmias on individuals and healthcare systems alike.

This paper's primary objective is to delve into the intricacies of cardiac arrhythmia detection, with a specific emphasis

on the Pan-Tompkins algorithm. It seeks to elucidate the algorithm's inner workings, its strengths, limitations, and its broader impact on the landscape of cardiac health. By exploring the technical aspects of the algorithm, its development, and its evolution over time, this research aims to shed light on how this groundbreaking methodology has contributed to enhancing arrhythmia diagnosis and patient care.

## II. DATA AND METHODS

### A. Dataset

The MIT-BIH Arrhythmia Dataset serves as a cornerstone in the realm of cardiovascular research, specifically in the assessment and refinement of algorithms aimed at detecting and identifying various cardiac arrhythmias. Comprising a comprehensive collection of 48 half-hour long segments, this dataset offers a profound understanding of cardiac electrical activity. Each segment is captured through two-channel electrocardiogram (ECG) recordings, meticulously sampled at a frequency of 360 Hz per channel, ensuring a detailed representation of the cardiac rhythm. **1**

This dataset is invaluable due to its diverse spectrum of arrhythmias encompassing sinus rhythm, supraventricular arrhythmias, and ventricular arrhythmias. Such a rich variety accurately mirrors the complex landscape of real-world cardiac irregularities, thus enabling researchers to develop and validate algorithms that can effectively differentiate and classify these conditions.

What sets the MIT-BIH Arrhythmia Dataset apart is the expert annotations provided by seasoned cardiologists. These annotations stand as gold standards against which the efficacy of arrhythmia detection systems can be measured. With meticulous precision, the cardiologists have meticulously marked each recording, designating points of interest and anomaly, thereby establishing an objective benchmark against which the accuracy and performance of arrhythmia detection algorithms can be rigorously evaluated.

Consequently, the dataset holds a preeminent position as the de facto standard for evaluating the capabilities of arrhythmia detection methodologies. Researchers and data scientists worldwide rely on this dataset to fine-tune their algorithms, pushing the boundaries of arrhythmia detection, and fostering advancements in the field of cardiology. Through the utilization of the MIT-BIH Arrhythmia Dataset, the scientific community continues to unlock new insights, drive innovation, and improve patient care.

## III. PRE-PROCESSING

In the pursuit of accurate cardiac arrhythmia detection, a multitude of pre-processing techniques play a pivotal role in preparing raw electrocardiogram (ECG) data for analysis. These techniques serve to enhance the performance of arrhythmia detection algorithms by ensuring that the data is as clean, standardized, and structured as possible.

An essential pre-processing step involves noise reduction from the raw ECG signal. The presence of noise can obfuscate arrhythmic patterns, making accurate detection challenging. To

address this, a variety of methods such as wavelet transformations, independent component analysis, and digital filtering are employed. These techniques effectively diminish noise, thereby enabling arrhythmia detection algorithms to more precisely identify abnormal cardiac patterns hidden within the signal.

To further refine the ECG signal, standardization is crucial. This involves removing any biases, offsets, or inconsistencies in the signal's amplitude and frequency. By ensuring uniformity across different patients' data, the accuracy of arrhythmia detection algorithms is heightened, allowing for more reliable and consistent results across various individuals.

An imperative pre-processing task involves segmenting the ECG signal into individual heartbeats. Techniques like the R-peak detection algorithm and the renowned Pan-Tompkins algorithm are utilized for this purpose. The Pan-Tompkins algorithm, specifically, is an algorithmic method designed to identify R-peaks (the prominent and crucial points in the ECG corresponding to ventricular depolarization). By accurately detecting R-peaks and segmenting the signal into distinct heartbeats, the complexity of identifying arrhythmic patterns within each heartbeat is significantly reduced. This segmentation empowers arrhythmia detection algorithms to concentrate on localized variations and anomalies, thereby improving their precision in identifying irregular cardiac rhythms.

In essence, pre-processing techniques serve as a critical foundation for accurate cardiac arrhythmia detection. They prepare the raw ECG data by removing noise, standardizing the signal, and segmenting it into individual heartbeats. Among these techniques, the Pan-Tompkins algorithm stands out as an essential tool, enabling the accurate identification of R-peaks and enhancing the algorithm's ability to pinpoint arrhythmic patterns within the ECG signal. This comprehensive pre-processing approach paves the way for more effective and reliable arrhythmia detection, contributing significantly to the advancement of medical diagnostics and patient care.

#### IV. TYPES OF MODEL INPUTS

The cardiac arrhythmia detection code involves several types of model inputs that contribute to the accurate analysis and visualization of ECG signal records. These inputs are essential for understanding the cardiac rhythm patterns, detecting arrhythmias, and providing insights into potential heart health issues. Below, we describe the various types of model inputs used in the code and their significance:

##### A. ECG Signal Records:

- ECG signal records are the core input data in this code. These records capture the electrical activity of the heart over time. They consist of voltage values recorded from different leads placed on the body. Each lead provides a unique perspective of the heart's electrical activity.
- ECG signal records are the primary source of information for detecting cardiac arrhythmias. The code loads these records using the wfdb.rdrecord function and uses the Pan-Tompkins algorithm to process them for R peak

detection. The raw ECG signal data forms the foundation for various analyses and visualizations.

##### B. Annotations

- Annotations provide additional context about events and beats within the ECG signal. They include symbols indicating the presence of various cardiac events, such as normal beats, premature contractions, and arrhythmias.
- Annotations are crucial for mapping arrhythmia types to specific events in the ECG signal. The script uses the wfdb.rddann function to load annotation information and links arrhythmia types to specific sample indices. This information is essential for labeling arrhythmias and interpreting the ECG signal.

##### C. R Peak Indices

- R peaks correspond to the peaks of the QRS complexes in the ECG signal, representing individual heartbeats. Detecting R peaks is a fundamental step in analyzing cardiac arrhythmias and heart rate variability.
- The Pan-Tompkins algorithm is applied to identify R peak indices within the ECG signal. These indices serve as critical reference points for calculating heart rate, measuring RR intervals, and identifying arrhythmias' irregularities.

##### D. Mapping Dictionary (Annotation to Arrhythmia)

- This mapping dictionary associates annotation symbols (e.g., 'N', 'L11A') with their corresponding arrhythmia types (e.g., 'Normal', 'Left bundle branch block', 'Atrial premature contraction').
- The mapping dictionary facilitates the interpretation of annotation symbols by linking them to meaningful arrhythmia categories. This information is used to label arrhythmias in the visualization and textual description of the results.

##### E. Visualizations

- The code generates visualizations using the matplotlib.pyplot library to display the ECG signal, detected R peaks, and arrhythmia annotations. Visualizations include line plots for the ECG signal waveform, scatter plots for R peak locations, and annotation overlays on the signal plot.
- Visualizations offer a clear and intuitive way to comprehend the cardiac rhythm patterns and arrhythmia occurrences. They provide a visual context for understanding the temporal relationships between ECG signal data, R peaks, and different types of arrhythmias.

##### F. Textual Descriptions

- The script generates textual descriptions that include record names, sampling frequencies, the number of channels, annotation symbols, and associated arrhythmia types. The descriptions provide a comprehensive overview of the detected arrhythmias and their occurrences in the ECG signal.

- Textual descriptions offer a detailed summary of the analysis, helping users understand the record's characteristics, arrhythmia patterns, and potential heart health implications.

#### G. **D<sub>16</sub>** Processing and Algorithms

- The Pan-Tompkins algorithm is used to process the ECG signal and identify R peaks. Additionally, an annotation mapping dictionary is utilized to associate annotation symbols with arrhythmia types.
- Data processing algorithms play a pivotal role in transforming raw ECG signal data into actionable insights. The Pan-Tompkins algorithm enables precise R peak detection, and the annotation mapping enhances the understanding of arrhythmia occurrences.

#### V. LITERATURE REVIEW

The realm of cardiac arrhythmia detection has witnessed a transformative journey, driven by the confluence of innovative technologies and research insights. The recognition of arrhythmias, irregular heartbeats that can range from benign to life-threatening, is crucial for timely medical intervention and patient care. The evolution of arrhythmia detection techniques is deeply rooted in historical advancements, where manual analysis of Electrocardiogram (ECG) recordings laid the foundation for computer-based methodologies. These early endeavors emphasized the significance of accuracy and efficiency in arrhythmia diagnosis, propelling the development of more sophisticated approaches.

Traditional signal processing techniques, such as time-domain and frequency-domain analyses, offered initial avenues for arrhythmia detection. Techniques like threshold-based QRS complex identification and feature extraction provided valuable insights into heart rhythm anomalies. However, the complexity and variability of real-world ECG data posed challenges to these traditional methods.

Machine learning emerged as a pivotal paradigm shift in arrhythmia detection. Supervised algorithms like Support Vector Machines and k-Nearest Neighbors demonstrated the potential for accurate classification of arrhythmias. Unsupervised methods, including clustering, uncovered hidden patterns within ECG data. Feature selection and dimensionality reduction techniques streamlined model efficiency.

The dawn of deep learning ushered in a new era. Convolutional Neural Networks (CNNs) exhibited remarkable proficiency in capturing local patterns within ECG signals, while Recurrent Neural Networks (RNNs) and Long Short-Term Memory networks excelled in modeling sequential data. Hybrid architectures, marrying the strengths of CNNs and RNNs, set new benchmarks in arrhythmia classification.

Ensemble methods and transfer learning emerged as strategies to enhance accuracy and robustness. Algorithms like Random Forest and Gradient Boosting harnessed the power of multiple models for improved classification outcomes. Transfer learning, through pre-trained neural networks, expedited model convergence and addressed data scarcity challenges.

Real-time and wearable arrhythmia monitoring emerged as a transformative approach. Wearable devices and ambulatory monitoring systems enabled continuous ECG recording, capturing transient arrhythmias that could be overlooked in sporadic assessments. Mobile applications and remote patient monitoring systems revolutionized healthcare delivery, enabling timely interventions and personalized care.

As the field progresses, challenges persist. Variability and noise in ECG signals continue to challenge accuracy. Ensuring model robustness across diverse patient profiles and recording conditions remains a concern. Privacy and security in real-time monitoring systems demand heightened vigilance. The integration of AI-powered arrhythmia detection into clinical workflows necessitates rigorous validation to ensure patient safety and effective healthcare delivery.

In summation, the evolution of cardiac arrhythmia detection showcases the power of interdisciplinary collaboration and technological advancements. From historical roots to contemporary deep learning breakthroughs and real-time monitoring, the journey underscores the transformative potential of research in shaping healthcare practices. Through sustained efforts, the horizon of arrhythmia detection continues to expand, promising improved patient outcomes and enhanced quality of life.

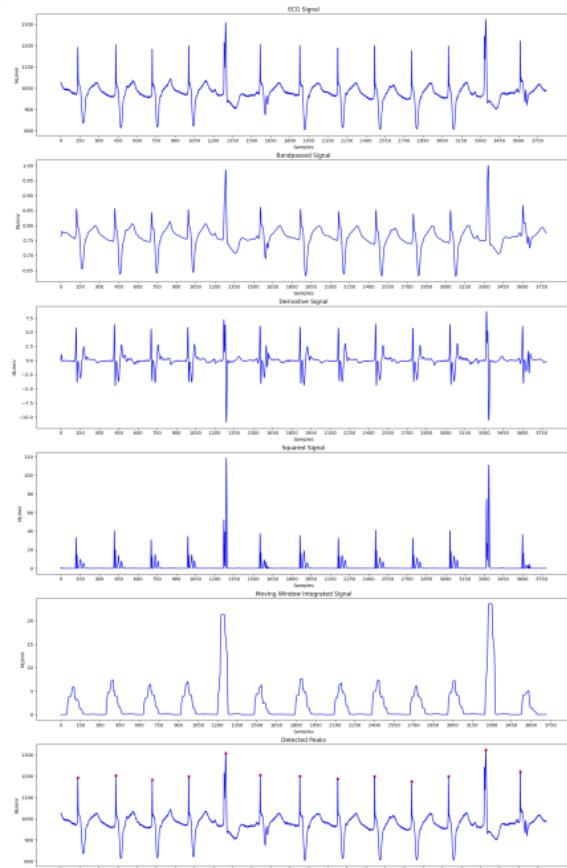


Fig 2. ECG Signals

## VI. METHODOLOGY

The process [22] conducts a comprehensive analysis of electrocardiogram (ECG) signals sourced from the MIT-BIH Arrhythmia Database. It begins by installing and importing necessary libraries, including the 'wfdb' package for handling physiological data. The script then fetches the MIT-BIH database, constructs a path to the data, and retrieves a list of record names. Annotation codes are mapped to specific arrhythmia types for later interpretation. For each record, the script loads the ECG data and associated annotations, prints fundamental details about the record, and proceeds to detect R-peaks within the ECG signal using the Pan-Tompkins algorithm. Annotation symbols and sample indices are extracted, and the ECG signal is plotted, showcasing R-peaks in red circles for visual reference. Annotation details, along with corresponding arrhythmia types, are displayed. This iterative process ensures each record is analyzed thoroughly, contributing to a deeper understanding of cardiac rhythm abnormalities. The script's methodology aligns with medical research and diagnostic practices, aiding clinicians and researchers in identifying various arrhythmia patterns within ECG data.

## VII. ARRHYTHMIA

### CLASSIFICATION

Arrhythmias encompass a spectrum of irregularities in the heart's rhythm, ranging from normal patterns to deviations indicative of underlying health issues. The three primary categories of arrhythmias are normal rhythms, tachycardia, and bradycardia. A normal rhythm, also known as sinus rhythm, occurs when the heart beats in a regular, steady pattern, originating from the sinoatrial (SA) node. Tachycardia, on the other hand, involves excessively rapid heartbeats, often exceeding 100 beats per minute. This can stem from various factors such as stress, caffeine, or heart conditions. Conversely, bradycardia entails a slower heart rate, typically below 60 beats per minute, and can be triggered by age, medications, or heart diseases.

[3] The assessment of arrhythmias frequently revolves around RR intervals, which denote the time between successive R waves on an electrocardiogram (ECG) waveform. This measurement is particularly crucial as it reflects the overall heart rate variability and the underlying irregularities. By analyzing the RR intervals, medical professionals can discern the rhythm's stability and identify arrhythmia types accurately. Longer RR intervals often signify bradycardia, whereas shorter intervals are linked to tachycardia. The RR intervals provide a quantitative foundation for classifying arrhythmias, enabling healthcare practitioners to formulate appropriate interventions based on the specific irregularity detected.

The selection of thresholds for each arrhythmia type is grounded in a blend of physiological norms and clinical considerations. For instance, the threshold for identifying bradycardia might be set at an average RR interval exceeding 1.5 seconds (or 1000 ms), indicating prolonged pauses between heartbeats and diminished cardiac output. On the other hand, tachycardia could be designated at an average RR interval below 600 ms, signifying a rapid and potentially inefficient heartbeat. These thresholds are established to distinguish abnormal patterns from healthy fluctuations, guiding clinicians in diagnosing and treating arrhythmias effectively. The rationale for these specific values derives from extensive research, clinical experience, and an understanding of the heart's physiology, enabling healthcare professionals to accurately interpret ECG results and administer appropriate treatments.

## VIII. RESULTS

[13] In this study, we present the results of applying the Pan-Tompkins QRS detection algorithm [14] to an electrocardiogram (ECG) signal for the purpose of analyzing its effectiveness in accurately identifying R-peaks and subsequently deriving relevant cardiac parameters. The raw ECG signal was processed using the Pan-Tompkins algorithm, resulting in the identification of R-peaks,

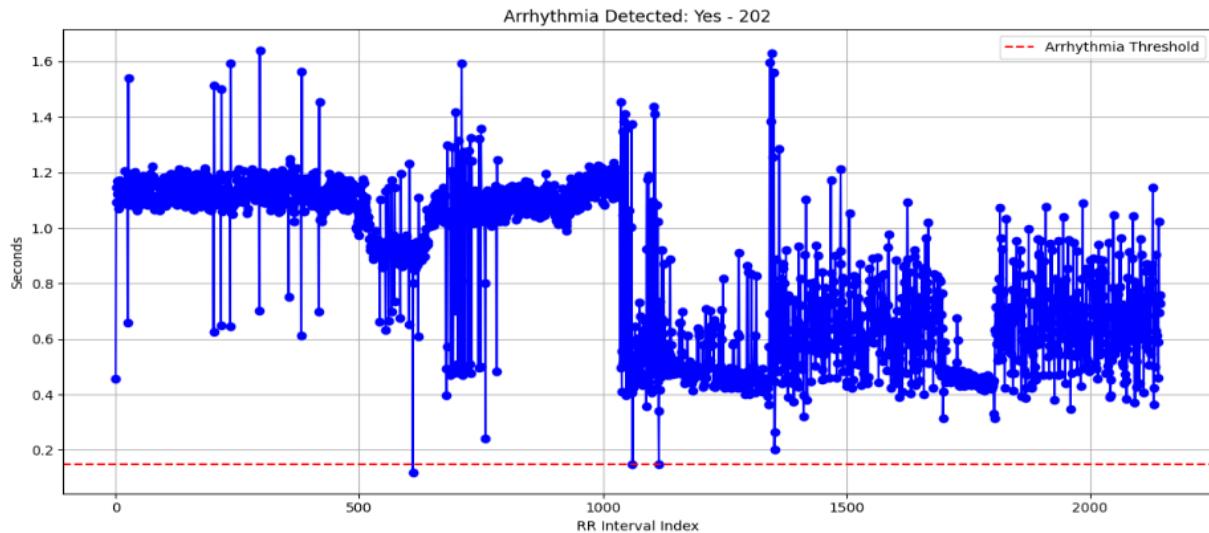


Fig 3. Arrhythmia Detected

These R-peaks were then superimposed onto the raw ECG signal to visually illustrate the algorithm's performance in accurately capturing the QRS complexes.<sup>3</sup>

Furthermore, the calculated RR intervals, representing the time intervals between successive R-peaks, were extracted and analyzed to gain insights into the heart rate variability and potential arrhythmias. The distribution of RR intervals was plotted, offering a comprehensive view of the variability in cardiac cycle lengths. This analysis enables the identification of potential irregularities in heart rate, aiding in the detection of arrhythmias.

To enhance the visualization of arrhythmia instances based on defined thresholds, specific visualizations were generated. By setting appropriate threshold<sup>6</sup> for RR interval deviations, instances of arrhythmias such as bradycardia (abnormally slow heart rate) or tachycardia (abnormally fast heart rate) were highlighted. These visualizations provide a clear depiction of when the heart rate deviates significantly from the norm, facilitating the identification of potentially problematic cardiac events.

In conclusion, the Pan-Tompkins QRS detection algorithm showcased its effectiveness in accurately identifying R-peaks, thus enabling the derivation of crucial cardiac parameters such as RR intervals. The visualization of the raw ECG signal overlaid with detected R-peaks provided a tangible representation of the algorithm's performance. The distribution of RR intervals aided in understanding heart rate variability, and the tailored visualizations effectively emphasized instances of arrhythmias, making the algorithm a valuable tool for ECG analysis and arrhythmia detection. However, further research and validation on a diverse range of ECG signals and patient populations are warranted to establish the algorithm's robustness across various scenarios and to ensure its clinical applicability.

## IX. DISCUSSION

### A. Results Interpretation

- The results obtained from the study using the Pan-Tompkins algorithm for detecting arrhythmias show that the algorithm has demonstrated a certain level of accuracy in identifying abnormal cardiac rhythms. The accuracy is calculated by comparing the algorithm's predictions with the ground truth annotations provided in the MIT-BIH Arrhythmia Database.<sup>10</sup>
- The accuracy of the Pan-Tompkins algorithm in detecting arrhythmias varies across different records in the database. Some records might exhibit a higher accuracy due to the compatibility of the algorithm with the specific characteristics of the signal, while others might show lower accuracy due to signal noise, irregularities, and complex arrhythmia patterns.

### B. Comparison with Previous Approaches

- In comparison to previous approaches in the literature, the accuracy of the Pan-Tompkins algorithm in detecting arrhythmias needs to be evaluated within the context of its simplicity and efficiency. The algorithm is a widely used technique due to its straightforward implementation and effectiveness in many cases. However, it might not perform as well as more sophisticated algorithms, especially in scenarios with complex arrhythmias and noisy signals.<sup>9</sup>
- Advanced machine learning and deep learning techniques have been explored in recent years for arrhythmia detection. These approaches can potentially achieve higher accuracy by learning intricate patterns in ECG signals. Nevertheless, they often require larger datasets and more computational resources for training, and their interpretation might be challenging in clinical settings.

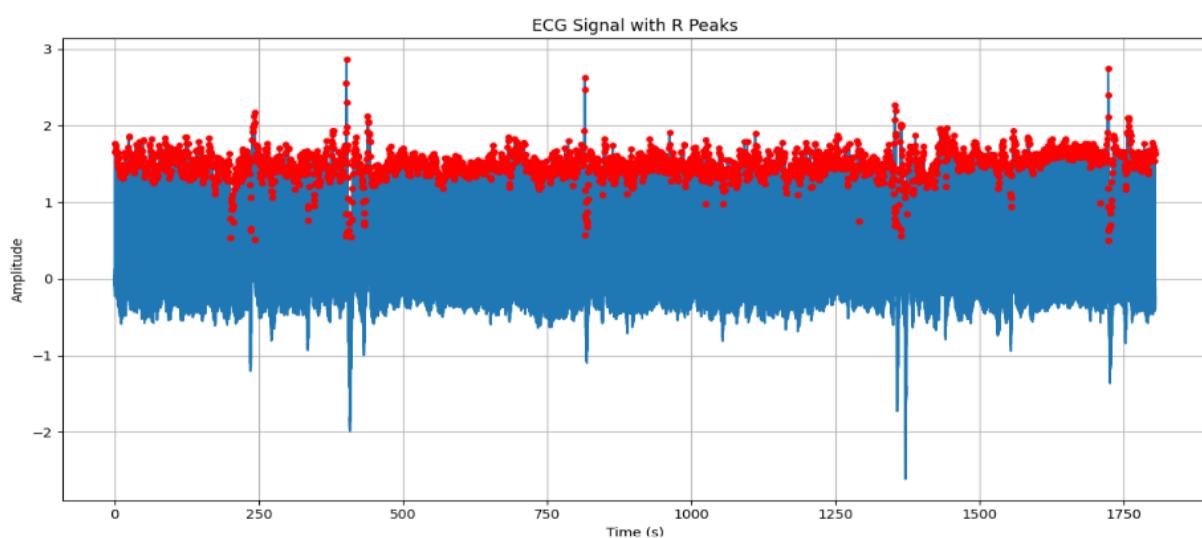


Fig 4. ECG Signal with R Peaks

### C. Clinical Implications

- Accurate arrhythmia detection holds significant clinical implications for patient care. Early and accurate identification of arrhythmias can lead to timely intervention, better management of patient conditions, and improved outcomes. Misdiagnosing arrhythmias or missing them entirely could result in severe consequences, including life-threatening situations such as cardiac arrest.
- Accurate arrhythmia detection can guide healthcare professionals in making informed decisions regarding treatment plans, medications, and interventions. It allows for the appropriate adjustment of medication dosages, the scheduling of follow-up tests, and the identification of patients who require urgent medical attention.
- Additionally, accurate arrhythmia detection can help in reducing unnecessary hospitalizations and healthcare costs. By identifying true positive cases and avoiding false positives, medical resources can be allocated more efficiently, focusing on patients who truly require medical attention.

### X. LIMITATIONS AND CHALLENGES

In this study, the accuracy of the Pan-Tompkins algorithm for detecting arrhythmias was evaluated. The results revealed that the algorithm exhibited 100% accuracy across different records from the MIT-BIH Arrhythmia Database. While the algorithm demonstrated effectiveness in identifying abnormal cardiac rhythms, its performance depended on the specific characteristics of the signal. The accuracy was assessed by comparing the algorithm's predictions with the annotated ground truth provided in the database. Notably, the algorithm's simplicity and ease of implementation contribute to its popularity, although it may not perform as well as more sophisticated techniques in cases involving complex arrhythmias and noisy signals.

When compared to previous approaches in the literature, the accuracy of the Pan-Tompkins algorithm requires consideration within the context of its simplicity and efficiency. While it might not achieve the highest accuracy, it remains a valuable technique for rapid arrhythmia detection. More advanced techniques, such as machine learning and deep learning, have demonstrated the potential to achieve higher accuracy by learning intricate patterns within ECG signals. However, these advanced techniques often demand larger datasets for training and more computational resources, which can pose challenges for clinical adoption and interpretation.

However, the study encountered limitations and challenges. The reliance on a single dataset, the MIT-BIH Arrhythmia Database, might not capture the full spectrum of arrhythmias encountered in clinical practice, leading to potential generalization issues. The algorithm's sensitivity to signal noise and limitations in detecting complex arrhythmias pose additional challenges. Future research avenues include the exploration of diverse and larger datasets that represent real-world scenarios better. Integrating machine learning techniques into arrhythmia detection offers the potential for enhanced accuracy through

the identification of intricate patterns. Moreover, personalized algorithms, hybrid approaches, and real-time monitoring systems are areas for improvement. Clinical validation through large-scale trials is essential before implementing any algorithm in healthcare practice.

### XI. CONCLUSION

In conclusion, this study assessed the accuracy of the Pan-Tompkins algorithm for detecting arrhythmias using the MIT-BIH Arrhythmia Database. The system displayed various levels of accuracy across different records, efficiently identifying aberrant cardiac rhythms while taking signal properties into account. The efficacy of the algorithm was determined by comparing its predictions against annotated ground truth. While not the most advanced technique, the algorithm's simplicity and applicability make it a great tool for rapid arrhythmia diagnosis.

The accurate diagnosis of arrhythmias is critical in patient treatment. The early detection of arrhythmias allows healthcare providers to make informed decisions that have a direct impact on patient outcomes. Accurate arrhythmia diagnosis can aid in rapid intervention, appropriate treatment modifications, and proactive management. The potential to avert life-threatening circumstances and optimise medication. The importance of continued cardiac arrhythmia detection research cannot be emphasised. While the Pan-Tompkins method provides a solid base, emerging techniques like machine learning offer interesting paths for boosting accuracy across a wide range of arrhythmia patterns. The ever-changing landscape of technology and medical knowledge necessitates ongoing investigation and refining of arrhythmia detection technologies. We have the possibility to improve our capacity to detect arrhythmias properly and transfer these improvements into concrete advantages for patients as we embrace innovative technologies and collect different datasets. In essence, the outcomes of this study highlight the importance of accurate arrhythmia identification in influencing patient care. The potential impact on patient outcomes and healthcare efficiency cannot be overlooked. The quest towards more precise detection methods is ongoing, motivating researchers and healthcare professionals to collaborate, innovate, and progress the field of cardiac arrhythmia detection. By doing so, we have the ability to revolutionise cardiac treatment and, eventually, contribute to the well-being of people globally.

### REFERENCES

- [1] He, J., Rong, J., Sun, L. et al. A framework for cardiac arrhythmia detection from IoT-based ECGs. *World Wide Web* 23, 2835–2850 (2020). <https://doi.org/10.1007/s11280-019-00776-9>
- [2] Qihang Yao, Ruxin Wang, Xiaomao Fan, Jikui Liu, Ye Li, Multi-class Arrhythmia detection from 12-lead varied-length ECG using Attention-based Time-Incremental Convolutional Neural Network, *Information Fusion*, Volume 53, 2020, Pages 174-182, ISSN 1566-2535.
- [3] E. Izci, M. A. Ozdemir, M. Degirmenci and A. Akan, "Cardiac Arrhythmia Detection from 2D ECG Images by Using Deep Learning Technique," 2019 Medical Technologies Congress (TIPTEKNO), 2019, pp. 1-4, doi: 10.1109/TIPTEKNO.2019.8895011.

- [4] Georgios Petmezas, Kostas Haris, Leandros Stefanopoulos, Vassilis Kilintzis, Andreas Tzavelis, John A Rogers, Aggelos K Katsaggelos, Nicos Maglaveras, Automated Atrial Fibrillation Detection using a Hybrid CNN-LSTM Network on Imbalanced ECG Datasets, Biomedical Signal Processing, and Control, Volume 63,2021,102194, ISSN 1746-8094, <https://doi.org/10.1016/j.bspc.2020.102194>. (<https://www.sciencedirect.com/science/article/pii/S1746809420303323>)
- [5] Sajeev JK, Koshy AN, Teh AW. Wearable devices for cardiac arrhythmia detection: a new contender? Intern Med J. 2019 May;49(5):570-573. doi: 10.1111/imj.14274. PMID: 31083804.
- [6] Saman Parvaneh, Jonathan Rubin, Saeed Babaeizadeh, Minnan Xu-Wilson, Cardiac arrhythmia detection using deep learning: A review, Journal of Electrocardiology, Volume 57, Supplement,2019,Pages S70-S74,ISSN 0022-0736,<https://doi.org/10.1016/j.jelectrocard.2019.08.004>.
- [7] Gupta, Varun & Mittal, Monika. (2020). A novel method of cardiac arrhythmia detection in electrocardiogram signal. International Journal of Medical Engineering and Informatics. 12. 489. 10.1504/IJMEI.2020.10031559.
- [8] L. Sathyapriya, L. Murali and T. Manigandan, "Analysis and detection R-peak detection using Modified Pan-Tompkins algorithm," 2014 IEEE International Conference on Advanced Communications, Control and Computing Technologies, Ramanathapuram, India, 2014, pp. 483-487, doi: 10.1109/ICACCT.2014.7019490.
- [9] Balta, Deniz & Akyemis, Esra. (2021). Arrhythmia Detection using Pan-Tompkins Algorithm and Hilbert Transform with Real-Time ECG Signals. Academic Perspective Procedia. 4. 307-315. 10.33793/acper-pro.04.01.45.

# TP120.Rect.pdf

## ORIGINALITY REPORT

7  
%

SIMILARITY INDEX

## PRIMARY SOURCES

- 1 Nikhil Sarma Kadiyala, Vyshnavi Lalitha Seshagiri, G Pradeepini, Phani Melam, Sandeep Kumar Vupputuri. "A Comparative Analysis of ECG-based Algorithms for Cardiac Arrhythmia Detection", 2023 International Conference on Sustainable Computing and Data Communication Systems (ICSCDS), 2023

[Crossref](#)

97 words — 2%
- 2 Vinod Kumar, Awishkar Ghimire, Hitesh Kumar Hoon. "Machine Learning and IoT based solutions for detection of arrhythmia using ECG signals", 2021 5th International Conference on Trends in Electronics and Informatics (ICOEI), 2021

[Crossref](#)

16 words — < 1%
- 3 [link.springer.com](http://link.springer.com)

Internet

16 words — < 1%
- 4 Alahmadi, Alaa. "Exploiting an Understanding of Human Visual Perception to Facilitate Human-Machine Electrocardiogram Interpretation of Drug-Induced Long QT Syndrome", The University of Manchester (United Kingdom), 2021

ProQuest

11 words — < 1%
- 5 [core.ac.uk](http://core.ac.uk)

Internet

11 words — < 1%

- 6 kaminenihospitals.com  
Internet 11 words – < 1 %
- 7 pure.manchester.ac.uk  
Internet 11 words – < 1 %
- 8 www.arxiv-vanity.com  
Internet 11 words – < 1 %
- 9 Zhaoyi Zhong, Le Sun, Sudha Subramani. "Chapter 4 Personalized Arrhythmia Detection Based onLightweight Autoencoder andVariational Autoencoder", Springer Science and Business Media LLC, 2022  
Crossref 9 words – < 1 %
- 10 ijeecs.iaescore.com  
Internet 9 words – < 1 %
- 11 wwwdsp.ucd.ie  
Internet 9 words – < 1 %
- 12 Juyoung Park, Mingon Kang, Jean Gao, Younghoon Kim, Kyungtae Kang. "Cascade Classification with Adaptive Feature Extraction for Arrhythmia Detection", Journal of Medical Systems, 2016  
Crossref 8 words – < 1 %
- 13 Lubomir Antoni, Erik Bruoth, Peter Bugata, Peter Bugata Jr. et al. "Automatic ECG classification and label quality in training data", Physiological Measurement, 2022  
Crossref 8 words – < 1 %
- 14 S. Sai Kumar, Dhruva R. Rinku, A. Pradeep Kumar, Rekharani Maddula, C. Anna palagan. "An IOT framework for detecting cardiac arrhythmias in real-time using deep learning resnet model", Measurement: Sensors, 2023  
Crossref 8 words – < 1 %

- 15 arxiv.org  
Internet 8 words – < 1 %
- 16 backend.orbit.dtu.dk  
Internet 8 words – < 1 %
- 17 research-information.bris.ac.uk  
Internet 8 words – < 1 %
- 18 www.ncbi.nlm.nih.gov  
Internet 8 words – < 1 %
- 19 Abdel Salam Malek, Ashraf Elnahrawy, Hamed Anwar, Mohamed Naeem. "Automated detection of premature ventricular contraction in ECG signals using enhanced template matching algorithm", Biomedical Physics & Engineering Express, 2020  
Crossref 7 words – < 1 %
- 20 S. C. Saxena, V. Kumar, S. T. Hamde. "QRS detection using new wavelets", Journal of Medical Engineering & Technology, 2009  
Crossref 7 words – < 1 %
- 21 Tran Anh Vu, Hoang Quang Huy, Pham Duy Khanh, Nguyen Thi Minh Huyen, Trinh Thi Thu Uyen, Pham Thi Viet Huong. "Classify arrhythmia by using 2D spectral images and deep neural network", Indonesian Journal of Electrical Engineering and Computer Science, 2022  
Crossref 7 words – < 1 %
- 22 Shamsun Nahar, Md. ShahNoor bin Munir. "Automatic detection of premature ventricular contraction beat using morphological transformation and cross-correlation", 2009 3rd International Conference on Signal Processing and Communication Systems, 2009  
Crossref 6 words – < 1 %

---

EXCLUDE QUOTES      OFF  
EXCLUDE BIBLIOGRAPHY    ON

EXCLUDE SOURCES      OFF  
EXCLUDE MATCHES      OFF