# Classification of Physiological Time-series Data with Machine Learning

## *ECG Analysis and Classification on the MIT-BIH Arrhythmia Database*

*Author*: **SHIVANI BATTU**

***Abstract*—This project explores 3 machine learning models to classify the physiological ECG signal into respective rhythms using the MIT-BIH Arrhythmia Database. Several classification algorithms, including Random Forest (RF), Support Vector Machine (SVM), and Neural Network model were trained on the data and were evaluated based on accuracy, sensitivity(recall), and precision to determine their effectiveness in detecting the presence of arrhythmia in patients from their ECG signal.**

***Index Terms*—Arrhythmia, ECG(Electrocardiogram), Noise Filtration**

## I. INTRODUCTION

A human body emits many types of signals based upon electrical and magnetic activities caused by the circulation of an electric charge in the body and one of these is physiological ECG signal.A physiological ECG signal has X-axis representing time, and Y-axis showing the amplitude of the ECG waves in millivolts.

The classification of ECG signals plays a critical role in diagnosing cardiovascular diseases, particularly in detecting arrhythmias that indicate abnormal heart rhythms. The MIT-BIH Arrhythmia Database provides a well-established benchmark for testing algorithms that classify various arrhythmias based on ECG signal patterns. Each heartbeat within this dataset is annotated by cardiologists, allowing for rigorous testing of classification methods.

The aim of this study is to create a machine learning pipeline to classify arrhythmias in the MIT-BIH dataset, utilizing preprocessing techniques, feature extraction, model optimization, and rigorous evaluation. By accurately categorizing arrhythmias, this study contributes towards reliable automated diagnostics, which could assist healthcare professionals in early detection and treatment planning.

## II. METHODOLOGY

### *A. Dataset Description*

The MIT-BIH Arrhythmia Database sourced from Physionet consists of ECG (electrocardiogram) data and corresponding annotations, stored in various file formats such as .atr, .dat, and .hea. containing various events in the ECG, such as arrhythmias with timestamps, the raw ECG signal data, and metadata about the ECG data (sampling rate, number of leads, etc.) respectively.

The MIT-BIH Arrhythmia records typically contain two leads of ECG data per record. This means each record has ECG signals recorded from two different locations on the patient's body (for example, lead II and lead V1). This study has been carried on Channel I depicting lead II ( as in Figure 1) which runs from the right arm to the left leg and is highly sensitive for detecting arrhythmias, such as atrial fibrillation or ventricular arrhythmias.

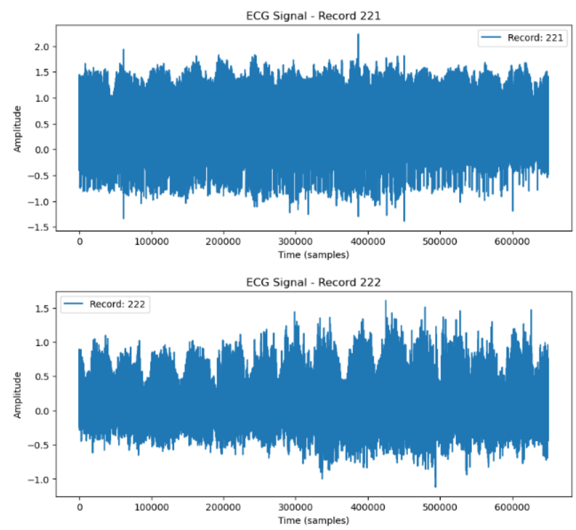


Fig. 1. ECG Signal

### *B. Data Preprocessing*

**Data Acquisition and Initial Exploration :**

The ECG signals and annotations from the MIT-BIH Arrhythmia Database were loaded, focusing specifically on Lead II present in channel 1 data, commonly used for detecting arrhythmias. This lead contains important information about the heart's electrical activity and is often used in medical diagnostics. After loading, the signals were inspected to understand the typical waveform patterns and potential anomalies present in arrhythmia classifications. The annotation labels were then reviewed to confirm the various arrhythmia categories, verifying that all essential classes were available for model training.

**Signal Segmentation and Preprocessing**

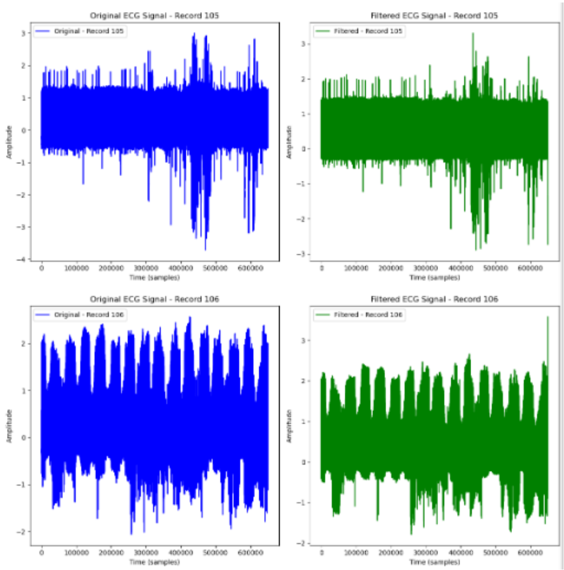


Fig. 2. ECG Signal Filtered

The first step in cleaning up an ECG signal is to filter the noise and we have used band pass filter. A bandpass filter is a combination of bothhigh-pass and low-pass filters that's used in electrocardiograms (ECGs) to extract specific frequency components and reduce noise. The human ECG spectrum ranges from 0.05 Hz to 100 Hz. For diagnostic purposes, a recommended filter setting of 0.05 Hz was used to reduce the noise while displaying the ECG with the maximum available frequency bandwidth.

In the next steps of data preparation, Normalization is implemented which ensured that the data is on the same scale, reducing bias introduced by varying magnitudes across different records or signals. Min Max Scaling was used to normalize the signal.

**Pre-processing:** Given the continuous nature of ECG signals, windowing techniques were applied to segment the data. Each window represented a portion of the ECG signal containing a complete or partial heartbeat. Segmentation was critical for creating a structured dataset suitable for machine learning. To address variability in signal amplitudes and baseline wander, we normalized the data within each segment to bring all signals to a common scale, ensuring the model could focus on pattern variations rather than amplitude differences.

Window Segmentation: Sliding window technique involves moving the window along the time axis by a certain stride or step size (e.g., 1 second or 2 seconds). The window size selected was a 2 second cycle with half-window size(i.e 1 second ) overlap. This allowed to analyze shorter portions of the signal, which is typical in time-series analysis.

Local Maxima and Minima: Within each window segment, 'find_peaks()' function was used to detect local maxima and minima. These represent the peaks and troughs of the ECG waveform, and their positions help calculate features like amplitude and intervals.

Amplitudes and Intervals: Amplitude is the difference between a peak and a trough, which gives insights into the signal's intensity. The intervals between consecutive maxima gave the time between heartbeats (R-R intervals).

R peaks : Since ECG analysis requires specific intervals and peaks (such as R-peak locations), features are calculated from the raw signal.

Heartbeat : Using the sample points provided in the annotation, heartbeat locations were identified.

RR Intervals: Calculated as the difference between consecutive R-peaks, the RR Intervals were computed for every window.

Feature Engineering: Relevant features were extracted from the raw signals, focusing on physiological insights, such as peak amplitudes and interval measurements, especially in the ECG's Lead II, which is prominent for arrhythmia detection. Also, statistical metrics were calculated for segments of the signal (e.g., mean, median, standard deviation, skewness, kurtosis, local maxima and minima, heart rate, maximum frequency, Average PSD (The power spectral density (PSD) of the signal describes the power present in the signal as a function of frequency, per unit frequency)) for better feature representation.

Importing Rhythms : Table of Rhythms was used to input labels to the sliding windows of every record based on the consolidated duration of each label in the raw signals.

**Feature Extraction** As a summary of the above, specific features that are clinically relevant to arrhythmia detection were extracted. The P, Q, R, S, and T peaks in each heartbeat were calculated, as these are key points in the ECG signal that represent different phases of cardiac activity. The distance between peaks, as well as the amplitude variations, were computed to capture the rhythm and morphology of each heartbeat. In addition to peak detection, statistical metrics within each window, such as mean, median, standard deviation, skewness, and kurtosis, which provide insights into the shape and variability of the ECG signals were calculated.

**Model Selection and Initial Training**:

To classify the arrhythmia types, multiple machine learning models were initially experimented, including neural networks, support vector machines, and random forests, evaluating each for initial accuracy and interpretability. Random Forest emerged as a strong candidate due to its ensemble nature, which combines multiple decision trees to improve classification accuracy and robustness. After preliminary training, the model demonstrated potential to distinguish between arrhythmias effectively, albeit some limitations in recognizing minority classes due to class imbalance.

**Model Optimization with Randomized Search**:

For improved performance, optimization was used on the Random Forest model using Randomized Search to tune its hyper parameters. This process allowed to identify the optimal settings for tree depth, number of estimators, and other hyper parameters by testing a random subset of parameter combinations. By maximizing accuracy during validation, the optimized model showed marked improvement, especially in capturing subtle distinctions between arrhythmias.

### C. Data Visualization

- The target class distribution :
  The target distribution plot (Figure 3 assesses the balance of arrhythmia classes within the dataset. As this is a case of imbalance, resampling techniques have been used to ensure robust classification and prevent biases toward majority classes. There are a total of 15 classes - N, AFIB, P, B, SBR, T, AFL, PREX, BII, NOD, SVTA, VT, IVR, VFL, AB.

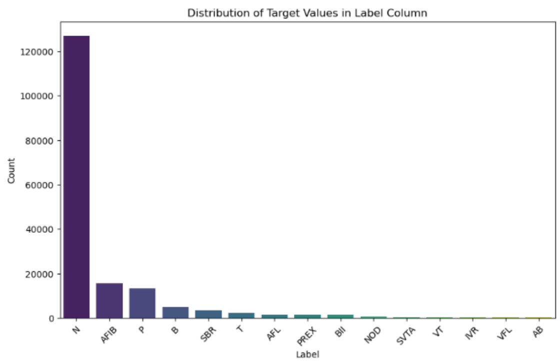


Fig. 3. Distribution of the target variable

- Window plot for R-peaks: The window plot (Figure 4) focuses on identifying and marking R-peaks, as these are the most prominent features in an ECG signal and provide valuable insights for arrhythmia detection. This visualization was crucial for understanding the regularity and characteristics of individual beats, making it easier to identify any abnormalities in the ECG signal.

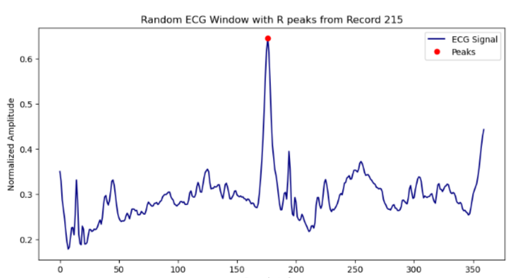


Fig. 4. R peaks in a window plot

### D. Model Training and Development

This paper compares the performance of 3 classification models:

**Random Forest (RF)** : Random Forest is an ensemble learning method used for classification, regression, and other tasks, which operates by constructing multiple decision trees during training. For classification tasks, it aggregates the output of each tree to make a final decision based on majority voting, enhancing accuracy and reducing overfitting by relying on diverse models.

Random Forest is used as one of the classification models to distinguish between different arrhythmia classes. After preprocessing the ECG data (including feature extraction, normalization, and peak detection), Random Forest model is used to learn patterns from these features, leveraging its robustness and ability to handle complex patterns in biomedical data. Performance was evaluated with accuracy, recall, and precision metrics, ensuring it captured relevant features across various ECG signals.

**Support Vector Machine (SVM):**

**Neural Network Model:** To explore the potential of deep learning, neural network model was tailored for multi-class classification. Using one-hot encoding to represent each arrhythmia type, the network was trained on the ECG segments, aiming to learn complex patterns across the arrhythmia classes. While the neural network provided an alternative approach to arrhythmia detection, it was noted that it required a substantial amount of training data and time to converge. After multiple training and validation rounds, the network achieved a modest accuracy on the test data, showing strengths in some classes but struggling with minority arrhythmias.

In multiclass classification with neural networks, the goal is to classify input data into one of several categories. Neural networks achieve this through multiple layers of interconnected neurons that learn complex patterns in the data. For multiclass problems, one-hot encoding is often applied to the target labels to make them compatible with neural network architectures.

With one-hot encoding , each category is represented by a binary vector where only the index corresponding to the target class is set to 1, and all others are 0. This transformation allows the neural network to treat each class distinctly and calculate probabilities for each class independently in the output layer, typically using a softmax activation function. Softmax normalizes the network outputs into a probability distribution across the classes, enabling clear predictions based on the highest probability score.

The dataset was split into train, validation, and test sets in a 70:15:15 ratio using stratified sampling. This approach ensured that we have a dedicated validation set for model tuning and a final test set for evaluating generalization performance.

### E. Model Evaluation Metrics

Accuracy: The proportion of correctly predicted observations.

$$Accuracy = TP + TN/(TN + FN + TP + FP) \quad (1)$$

Recall (True Positive Rate) ): The ability of the model to correctly predict positive instances.

$$Recall = TP/(TP + FN) \quad (2)$$

Precision: The proportion of positive identifications that were actually correct.

$$Precision = TP/(TP + FP) \quad (3)$$

where ,

TP = True Positives =Arrhythmia patient of Class X correctly classified as class X

FP = False Positives = Non-Arrhythmia patients wrongly identified as Arrhythmia of any class.

FN = False Negatives = Arrhythmia patients incorrectly identified as non-Arrhythmia or Arrhythmia patients of class X incorrectly classified as Class Y or other.

TN = True Negatives = Non-Arrhythmia patients correctly identified as normal.

***Confusion Matrix:*** For each model, confusion matrices were generated and analyzed to determine the True Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN).

Comparison: A summary of these metrics across all models was presented in a grid format to highlight the strengths and weaknesses of each model.

## III. RESULTS

### A. *Inferences from the Evaluation Metrics*

***1. RANDOM FOREST:***

**Performance of Random Forest Baseline model:** After the initial model training and validation, Random Forest demonstrated a validation accuracy of 90.49% when hyper parameters were used to perform Randomized search cross validation for model optimization to be able to increase the performance accuracy.

**Performance of Random Forest model - After Model Optimization using Random Search CV:**

The optimization and feature selection for the Random Forest model on the MIT-BIH Arrhythmia dataset both showed high performance, with accuracy remaining relatively stable around 90.6% after optimization and slightly dropping to 90.45% after feature selection. Post-optimization, the Randomized Search CV yielded a test accuracy of 90.6%, with improved classification for certain arrhythmias, particularly normal beats (N) and premature ventricular complexes (P), which saw high recall and precision. However, minority classes like AB, AFL, and VT had poorer recall, indicating difficulty in classifying these underrepresented types even with optimization.

After feature selection, there was a slight reduction in overall accuracy (90.45%), and precision (88.93%), showing that feature reduction didn't significantly affect model accuracy, but slightly impacted precision. The feature selection process may have removed some irrelevant or less impactful features, potentially streamlining the model without compromising general performance. These results suggest that while both optimization and feature selection maintain robust accuracy, optimization better addresses class imbalances.

Feature selection did not seem to be an effective approach in case of Arrhythmia classification using Random Forest as the accuracy did not improve significantly. There was an increase in the accuracy only with Randomized Search Cross Validation.

TABLE I
PERFORMANCE EVALUATION METRICS FOR RANDOM FOREST

| **Random Forest** | **Accuracy** | **Recall** | **Precision** |
|---|---|---|---|
| Baseline model(Val) | 0.9049 | | |
| After Model optimization | 0.9060 | | |
| After Random Search CV | 0.9045 | 0.9045 | 0.8893 |

***SUPPORT VECTOR MACHINE:***

**Performance of SVM model:**

The SVM model achieved a reasonable accuracy of 73.28%, but its low precision (53.69%) suggests a high rate of false positives, potentially due to class overlap or imbalance. Hyperparameter tuning was used to improve performance.

**Performance of SVM model - After hyperparameter tuning**:

After hyperparameter tuning the performance has improved significantly. With an accuracy improvement from 73% to 79% and precision and recall having significant improvement to 78.66% and 79.05% respectively, it is safe to say that Hyperparameter tuning has improved the performance of SVM model.

TABLE II
PERFORMANCE EVALUATION METRICS FOR SVM

| **SVM** | **Accuracy** | **Recall** | **Precision** |
|---|---|---|---|
| Baseline Model | 0.7328 | 0.7328 | 0.5369 |
| After hyperparameter tuning | 0.79 | 0.7866 | 0.7905 |

***NEURAL NETWORK MODEL:***

**Performance of Neural Network Baseline model:**

Neural network classifier was implemented on the MIT-BIH Arrhythmia dataset to identify various arrhythmia types. After encoding the labels with one-hot encoding, the network was trained with the processed time-series data, which had undergone feature extraction and normalization steps. The neural network learned to classify the data into multiple arrhythmia types based on patterns identified during training.

**Performance of Neural Network model- After optimization:** The neural network's parameters were optimized using adam optimizer, including the number of neurons and layers, using randomized search and monitored its performance with metrics such as accuracy, precision, and recall. This approach allowed to evaluate how well the model distinguishes between different types of arrhythmias and where it may need further tuning, particularly for minority classes. The model showed strong performance in the majority classes, though challenges remained in accurately classifying less frequent arrhythmia types, emphasizing the need for continued optimization and data balancing strategies. The results of Neural Network on the test set show that the neural network classifier achieved an overall test accuracy of 77.47%, with precision and recall scores of 71.12% and 77.47%, respectively. However, the confusion matrix and classification report reveal mixed performance across different arrhythmia classes. Notably, the classifier performed well on the majority class (N), achieving a recall of 96% and an F1-score of 87%, indicating that the model learned the predominant patterns in this class. Additionally, other classes like P, BII, and SBR showed moderate precision and recall, suggesting some success in identifying these arrhythmias but with significant misclassifications into the more frequent classes.

In contrast, the model struggled with minority classes such as AB, AFIB, AFL, NOD, SVTA, and VT, with extremely low or zero precision and recall for these classes. This result indicates that the neural network's ability to capture the unique characteristics of these classes was limited, possibly due to class imbalance in the dataset and insufficient distinguishing features for rare arrhythmias. While the neural network can capture patterns for common arrhythmia types, it may need further optimization, such as implementing balanced training strategies or employing ensemble models, to improve recognition of underrepresented arrhythmia classes.

TABLE III
PERFORMANCE EVALUATION METRICS FOR NEURAL NETWORK

| **Neural Network** | **Accuracy** | **Recall** | **Precision** |
|---|---|---|---|
| Validation Accuracy | 0.7767 | | |
| With Adam Optimizer | 0.7747 | 0.7747 | 0.7112 |

## IV. DISCUSSION

The Random Forest model stands out as the best model here, achieving the highest accuracy (90.45%), recall (90.45%), and precision (88.93%). This strong performance across all metrics indicates that Random Forest effectively balances capturing true positives with minimizing false positives.

In terms of consistent performance, Random Forest outperforms SVM and Neural Network models, both of which fall below 80% accuracy. The lower recall and precision in SVM and Neural Network suggest they struggle with generalization compared to the Random Forest model.

TABLE IV
BEST MODEL COMPARISON

| **Model** | **Accuracy** | **Recall** | **Precision** |
|---|---|---|---|
| Random Forest | 90.45% | 90.45% | 88.93% |
| SVM | 79.0% | 78.66% | 79.05% |
| Neural Network | 77.47% | 77.47% | 71.12% |

## V. CONCLUSION

This arrhythmia classification project evaluated the efficacy of multiple machine learning models, including Random Forest, SVM, and Neural Network, in identifying arrhythmias from ECG data. The models were assessed based on accuracy, recall, and precision, with Random Forest emerging as the most effective due to its high accuracy (90.45%) and balanced precision and recall. This model's performance indicates robust generalization and reliability in capturing arrhythmic patterns. Pre-processing techniques, feature engineering, and peak detection played a crucial role in optimizing classification performance. Overall, Random Forest shows promise for arrhythmia detection, though future work could explore further tuning and deep learning approaches to enhance results further.