Classification of Physiological P-Wave in ECG with Machine Learning

P-wave Detection and Classification on the MIT-BIH Arrhythmia P-wave Database

Author: **SHIVANI BATTU**

***Abstract*—This project explores various machine learning models to classify the presence of p-wave in 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), precision to determine their effectiveness in detecting the presence of p-wave in ECG signal. Random Forest stood out to perform consistently better across all factors.**

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

I. INTRODUCTION

The MIT-BIH Arrhythmia Database is a benchmark dataset for the classification of arrhythmia types from electrocardiogram (ECG) signals, widely used in the study of heart conditions and diagnostic research. The focus of this project was to classify the presence of the P-wave, a critical element in ECG interpretation, which signals atrial depolarization and thus holds diagnostic value in identifying arrhythmic events.

The aim of this study was to create a machine learning pipeline to classify the presence or absence of p-wave in an ECG signal.

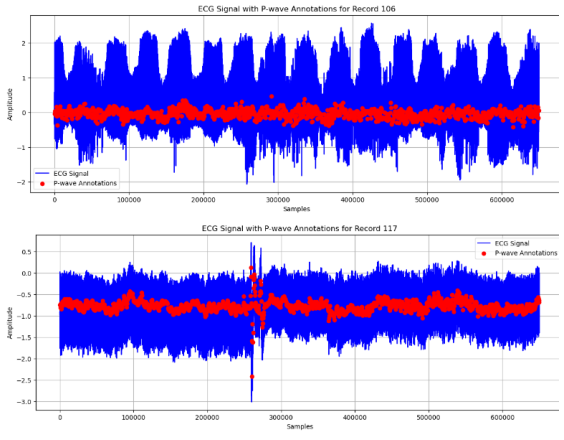

Fig. 1. ECG Signal with P-wave

II. METHODOLOGY

A. Dataset Description

The MIT-BIH Arrhythmia Database is a widely recognized dataset in biomedical research, specifically used for analyzing and classifying various types of arrhythmias based on ECG recordings. It was curated to provide reliable, annotated ECG records that researchers and clinicians could use to develop and evaluate algorithms for detecting arrhythmic events. The dataset consists of 48 half-hour recordings of ECG data from 47 patients, each recording containing annotations that mark different wave patterns and arrhythmic events.

This project focuses particularly on the **P-wave**, a key part of the ECG signal representing atrial depolarization.

Detecting this wave (as in Figure 1) accurately is vital for arrhythmia diagnosis and helps identify any irregularities in heartbeats.

B. Data Pre-processing

The raw ECG signals were reviewed to identify the key features that might contribute to an accurate classification. Firstly, the raw signal was filtered (as in Figure 2). Utilizing window-based segmentation to process each ECG recording, signal patterns were analyzed by dividing them into smaller intervals. For each interval, statistical attributes, such as the minimum, maximum, mean, standard deviation, skewness, and kurtosis, along with other characteristics that could capture the underlying signal structure were computed. Additionally, physiological signal points like local maxima and minima were calculated to observe signal oscillations that might align with the P-wave. To remove redundant information, feature importance was performed to remove features that showed little to no contribution to the model. Missing values were addressed to avoid skewing the classification outcome. Finally, we employed resampling techniques to handle any class imbalances in the P-wave data.

Feature Engineering Feature engineering was a vital part of the pre-processing as it involved transforming ECG signal characteristics into numerical descriptors. This included calculating the amplitude, average local maxima and minima, average RR intervals, and other frequency-based features, such

as maximum frequency and power spectral density. These attributes provided additional granularity to the dataset, helping capture subtle details within the ECG waveforms that could indicate the presence or absence of the P-wave.

Data Splitting and Normalization: We split the dataset into training and testing sets in 80:20 ratio using stratified sampling to ensure an unbiased evaluation. Following this, normalization was applied to scale the feature values, particularly important in neural networks and other machine learning algorithms sensitive to data scaling. Normalization enabled each feature to contribute proportionally to the classification task, improving model convergence during training.

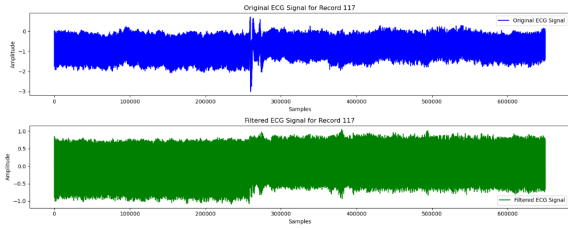

Fig. 2. ECG Signal Filtered

Signal Segmentation and Pre-processing

Given the continuous nature of ECG signals, windowing techniques were applied to segment the data. Each window of size 260 Hz with an overlap of 50% 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, the data within each segment was normalized using Z-score normalization to bring all signals to a common scale, ensuring the model could focus on pattern variations rather than amplitude differences.

C. Data Visualization

Primary data visualization involved the plotting of Target variable's class distribution (as in Figure 3)

- The target class distribution

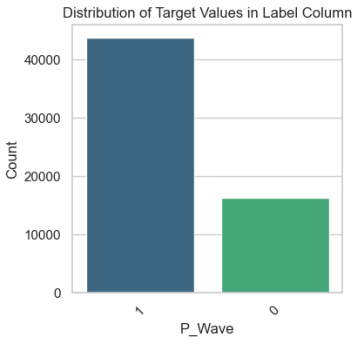

Fig. 3. Distribution of the target variable

- Target Vs Record
  The relation between Target variable(presence of p-wave or absence) was compared against the the records to check how the distribution of p-wave is across records. Figure 4 shows the clear relation among the two attributes.

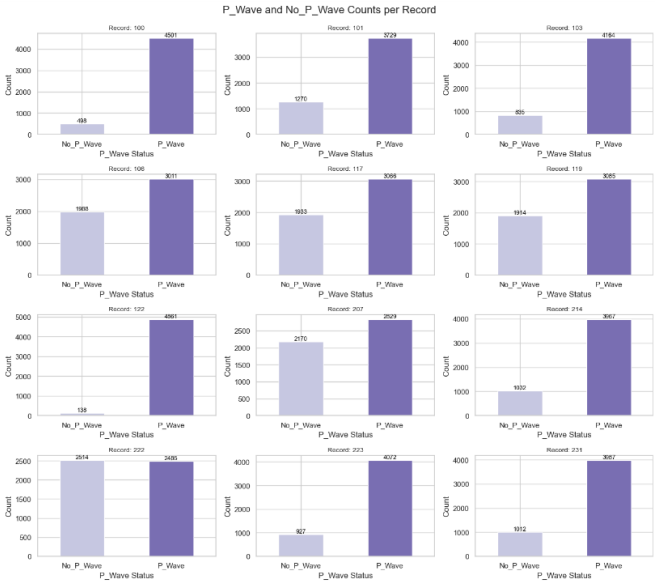

Fig. 4. Target Label Vs Record

- Window Plot of P-wave (Figure 5)

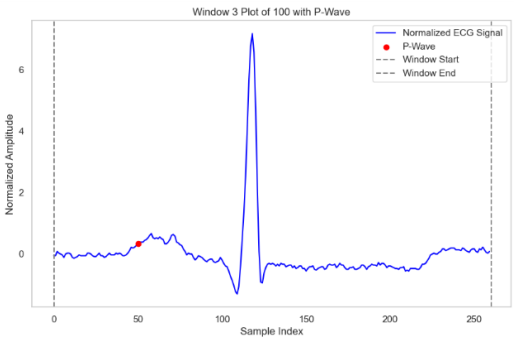

Fig. 5. Window plot of p-wave

- Feature plot (Figure 6)

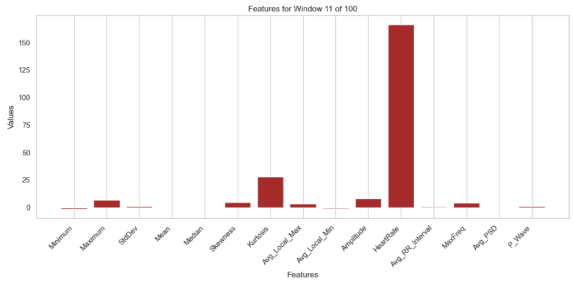

Fig. 6. Feature Plot of a P-wave window

D. Model Training and Development

This paper compares the performance of 3 classification models:

Random Forest (RF)
Support Vector Machine (SVM)
Neural Network Model

- **Model I : Random Forest** The Random Forest algorithm is an ensemble learning method that constructs multiple decision trees and outputs the mode of their predictions. It is known for its robustness against overfitting and its ability to handle large datasets with high dimensionality.

- **Model II : Support Vector Machine** The Support Vector Machine algorithm constructs a hyperplane in a high-dimensional space to separate different classes. SVM is particularly effective in high-dimensional spaces and is effective when the number of dimensions exceeds the number of samples.

- **Model III : Neural Network Model** : The neural network model implemented for binary classification of the MIT-BIH arrhythmia P-wave data aims to distinguish between the presence and absence of P-waves in ECG signals. With a structured architecture comprising dense layers and dropout for regularization, the model seeks to effectively learn from the provided features while minimizing overfitting. The use of the Adam optimizer enhanced the model's convergence speed and overall performance.
  For Neural networks, number of epochs need to be specified, the training will default to a preset value (often 1, but this can vary depending on the library and model). This means the model may not have enough iterations to learn effectively, potentially leading to poor performance. Not specifying epochs(Static training) for Random Forest and SVM does not make a difference since they don't utilize epochs in the same way. Instead, they focus on hyper,parameters relevant to their learning algorithms.

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.

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

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

RANDOM FOREST:

Performance of Random Forest Baseline model:

The Random Forest model demonstrates strong performance metrics across the board, with an accuracy of 92%, indicating that the majority of predictions are correct.

The high recall of 95% shows the model effectively identifies positive cases, minimizing false negatives, which is particularly important in medical diagnostics where missing positive cases can have serious consequences.

The precision of 94% further confirms the model's reliability, meaning it accurately classifies positive predictions with few false positives.

To check for performance improvement, Feature selection was done and evaluated on the test set.

Performance of Random Forest model - After feature Importance:

The model retains high effectiveness with only a slight reduction in accuracy, now at approximately 91.99%. Interestingly, recall has remained robust at 95.5%, suggesting that even with fewer features, the model continues to capture nearly all positive cases. Precision remains high at 93.66%, indicating that the model's positive predictions are still reliable. This shows that the feature selection process did not compromise the model's ability to identify true positives accurately, while maintaining an efficient and possibly more interpretable model with fewer features. (Figure 7

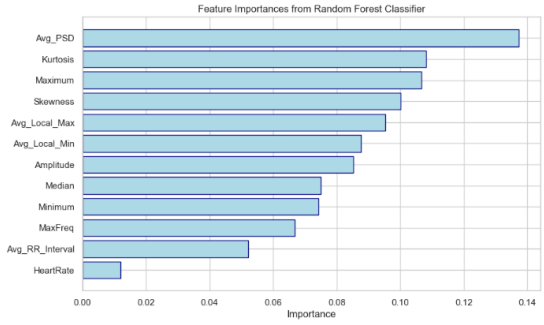

Fig. 7. Feature Importance for Random Forest model

Performance of Random Forest model - After hyperparameter tuning with Randomized Search CV:

The model's performance remained strong after hyperparameter tuning with Randomized Search CV, resulting in a slight improvement in metrics. The test set accuracy increased to 92.14%, and recall improved to 95.55%, indicating the model's enhanced ability to correctly identify positive cases.

The precision of 93.79% shows that the positive predictions are reliable, reflecting a balanced trade-off between precision and recall.

This confirms that hyperparameter tuning has refined the model, improving its predictive capacity without overfitting, and achieving a high level of generalization across the test data.

Performance of Random Forest model - After SMOTE oversampling:

After applying SMOTE (Synthetic Minority Over-sampling Technique), the model achieved a test accuracy of 91.30%, recall of 92.40%, and precision of 95.52%.

SMOTE has improved the recall slightly by creating synthetic samples for underrepresented classes, allowing the model to recognize positive cases more effectively. However, there was a small decrease in overall accuracy compared to previous results, which is typical after balancing techniques since the model now focuses more on the minority class. This shift improves the model's fairness and performance across classes, even if it slightly sacrifices overall accuracy.

SMOTE has thus balanced the model's performance by reducing bias toward the majority class, and the high precision shows it still effectively discriminates between positive and negative predictions.

TABLE I
PERFORMANCE EVALUATION METRICS FOR RANDOM FOREST

Random Forest	Accuracy	Recall	Precision
Baseline Model	0.9214	0.9555	0.9379
After Feature Selection	0.9199	0.9550	0.9366
After Hyperparameter tuning	0.9214	0.9555	0.9379
After Class Balancing	0.9130	0.9240	0.9552

SUPPORT VECTOR MACHINE:

Performance of SVM model:

SVM achieved an accuracy of 88.4%, with strong recall (94.1%) and precision (90.4%), showing effective detection of P-waves but slightly lower performance overall compared to other models. This result highlights SVM's sensitivity in identifying true positives, though at a minor cost to precision.

Performance of SVM model - After performing PCA:

After applying PCA to retain 98% of the variance, the features were reduced to 9 as shown in Figure 8 and the SVM model achieved an improved accuracy of 90.7%, along with a recall of 95.1% and precision of 92.5%. This indicates that dimensionality reduction effectively enhanced the model's performance, particularly in identifying true positives while maintaining a solid balance between sensitivity and precision.

NEURAL NETWORK MODEL:

Performance of Neural Network Baseline model:

The neural network model demonstrated an accuracy of 88.7%, reflecting a decent level of overall correctness in predictions. However, the recall of 89.0% indicates some missed positive cases, suggesting room for improvement in capturing all relevant instances. In contrast, the high precision of 95.2% signifies that when the model predicts a positive outcome, it is highly likely to be correct, highlighting its effectiveness in minimizing false positives.

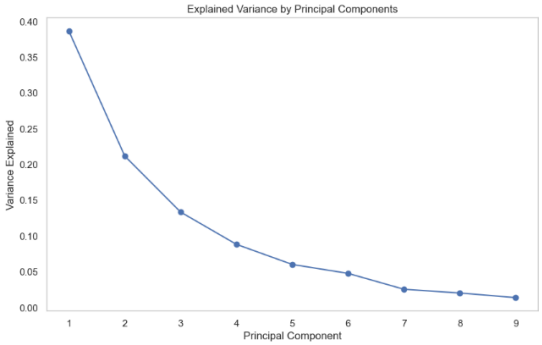

Fig. 8. PCA for SVM model

TABLE II
PERFORMANCE EVALUATION METRICS FOR SVM

SVM	Accuracy	Recall	Precision
Baseline Model	0.884	0.941	0.904
After PCA	0.907	0.951	0.925

To improve the performance of the Neural Network Model, Adam optimizer was used.

Performance of Neural Network model - With Adam Optimizer:

The neural network model utilizing the Adam optimizer achieved an overall accuracy of 89.15%, indicating a strong performance in correctly classifying instances. The precision of 92.53% suggests that the model is highly reliable when predicting positive cases, minimizing false positives effectively. Additionally, the recall of 92.60% indicates a commendable ability to capture true positive instances, showcasing the model's balance between precision and recall in its predictions.

Optimization Improved the performance!

The model utilizing the Adam optimizer demonstrates improved overall accuracy (89.15% vs. 88.71%) and a better balance between precision and recall. While the precision without the optimizer is higher (95.20%), the Adam-optimized model provides a more rounded performance, effectively capturing more true positives (higher recall) while still maintaining strong precision.

IV. DISCUSSION

Throughout this analysis, the impact of feature engineering and preprocessing techniques on model performance was

TABLE III
PERFORMANCE EVALUATION METRICS FOR NEURAL NETWORK

Neural Network	Accuracy	Recall	Precision
Baseline Model	0.8871	0.8902	0.9520
With Adam Optimizer	0.8915	0.9253	0.9260

observed. The statistical descriptors of ECG segments were instrumental in differentiating between the presence and absence of the P-wave, highlighting the value of combining both time and frequency domain features for ECG classification tasks. Furthermore, it was noted that while neural networks offered the best performance after tuning, their training time and sensitivity to parameter changes were higher than traditional models like random forests. The choice of normalization and the use of dropout regularization were pivotal in controlling over-fitting and enhancing generalizability in the neural network. An additional observation was the role of class imbalance and its effect on recall and precision. The application of Synthetic Minority Over-sampling Technique (SMOTE) helped in balancing the dataset, thus enhancing the performance and stability of the models.

Based on these observations, the below conclusions can be drawn:

1. Best Performing Models:

Random Forest had the highest accuracy (92.14%) and recall (95.55%) among the three models, suggesting it is the most effective at correctly identifying both classes, especially for high recall (true positives).

2. Consistency across metrics:

- Random Forest also showed strong consistency across accuracy, recall, and precision (93.79%), with only minor variation, which is a sign of balanced performance in detecting both true positives and negatives.

- SVM had similar consistency, with high recall (95.10%) but slightly lower accuracy (90.70%) and precision (92.5%), indicating it performs well but may have slightly more false positives than Random Forest.

- Neural Network had lower accuracy (89.15%) but still high precision (92.60%) and recall (92.53%), though its lower accuracy suggests more variability and slightly lower overall performance.

TABLE IV
BEST MODEL COMPARISON

Model	Accuracy	Recall	Precision
Random Forest	92.14%	95.55%	93.79%
SVM	90.70%	95.10%	92.5%
Neural Network	89.15%	92.53%	92.60%

V. CONCLUSION

The study demonstrated that advanced preprocessing, feature engineering, and model tuning techniques can achieve robust performance in ECG classification tasks, specifically in identifying the P-wave. The use of Random Forest can achieve the highest scores across all metrics and maintain a balanced accuracy, precision, and recall.

Future work may include exploring real-time classification capabilities and examining deep learning architectures that can automate feature extraction for further improved diagnostic accuracy.

This study contributes to the growing body of research focused on leveraging machine learning for reliable and accurate arrhythmia detection in healthcare, underscoring the potential of data-driven approaches in improving patient outcomes.