Real-time Epileptic Seizure Detection Using EEG and Machine Learning

Khushbakht Zia

Machine Learning Task

April 24, 2025

Institute of Management Sciences, Peshawar

Abstract

This project focuses on detecting epileptic seizures using EEG signals collected from six patients at the American University of Beirut Medical Center. The EEG recordings include 21 scalp electrodes sampled at 500Hz and are bandpass filtered. Data preprocessing and multiple machine learning algorithms are applied to classify seizure and non-seizure states with high accuracy.

Introduction

Epilepsy is a neurological disorder characterized by recurrent seizures. Electroencephalography (EEG) is commonly used to monitor brain activity and detect abnormal patterns indicative of seizures. This project aims to automate seizure detection using machine learning techniques applied to EEG signals.

0.1 Objectives

The primary objectives of this project are:

- To explore and analyze EEG data collected from epileptic patients.
- To apply data preprocessing techniques to enhance model performance.
- To develop and evaluate various machine learning models (Random Forest, XGBoost, and LightGBM) for accurate classification of seizure and non-seizure EEG segments.
- To compare the models based on accuracy, F1-score, and computational efficiency.

0.2 Dataset Description

- Source: American University of Beirut Medical Center (2014–2015)
- Patients: 6, with a total of 35 seizures
- Channels: 21 scalp electrodes following the 10–20 system
- Sampling Rate: 500 Hz
- Data Format: European Data Format (EDF), also available as .npy and .mat
- Seizure Types:
  - Complex Partial Seizures (3034s)
  - Electrographic Seizures (705s)
  - Video-detected Seizures without EEG change (111s)
- Normal Data: 3895s (balanced against seizure durations)

1 Exploratory Data Analysis

Initial data analysis included:

- **Class Distribution**: A significant class imbalance was observed between seizure and non-seizure samples. This was visualized using bar charts (Figure 1), showing seizure types were underrepresented compared to normal data.

- **Signal Visualization**: Sample EEG signals (Figure 2) displayed distinct patterns in seizure events, such as sudden spikes and rhythmic discharges.
- **Feature Distribution**: Using boxplots (Figure 3), we compared features (e.g., signal amplitude) across classes, revealing distinct differences.
- **Dimensionality Analysis**: A t-SNE plot (Figure 4) illustrated some separability between seizure and non-seizure states even in raw feature space.

These observations confirmed the feasibility of machine learning-based classification and guided further preprocessing and modeling.

2 Data Preprocessing and Visualization

EEG data preprocessing is critical for noise removal, normalization, and improving model training. The following steps were applied:

- **Variance Thresholding**: Low-variance features were eliminated to reduce dimensionality and computational load without losing useful information.
- **Standard Scaling**: Each feature was normalized to zero mean and unit variance, ensuring that all features contributed equally to the learning algorithm.
- **SMOTE (Synthetic Minority Over-sampling Technique)**: Due to class imbalance, SMOTE was used to generate synthetic samples of the minority (seizure) class, enhancing model generalization and performance.

These preprocessing steps significantly improved classification performance by reducing bias and ensuring balanced training data.

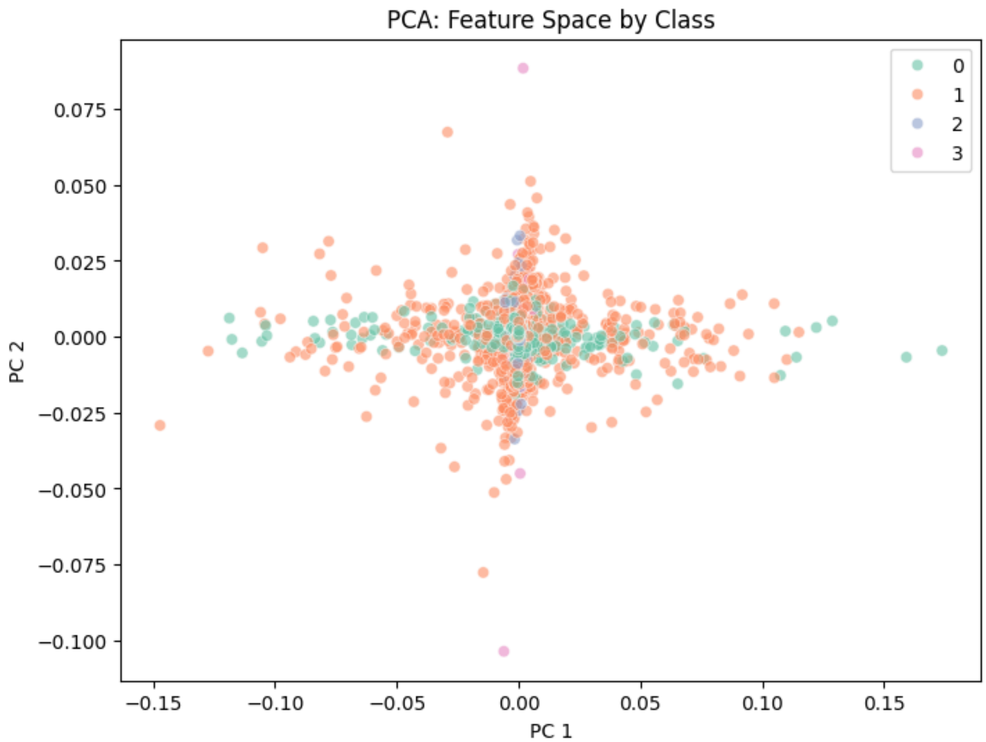

Figure 1: Visualization 1: Dimensionality Reduction Class Separation

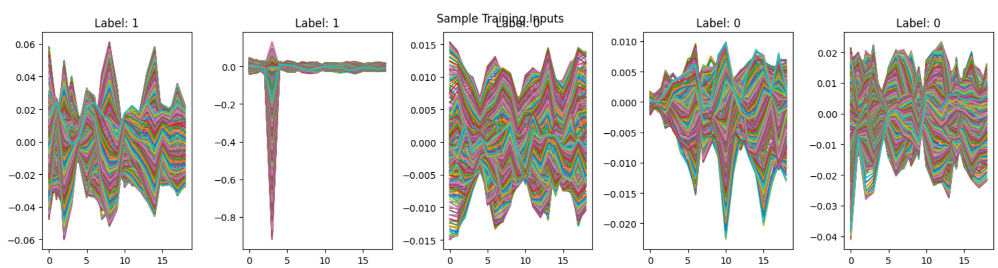

Figure 2: Visualization 2: Visualize Sample Signals

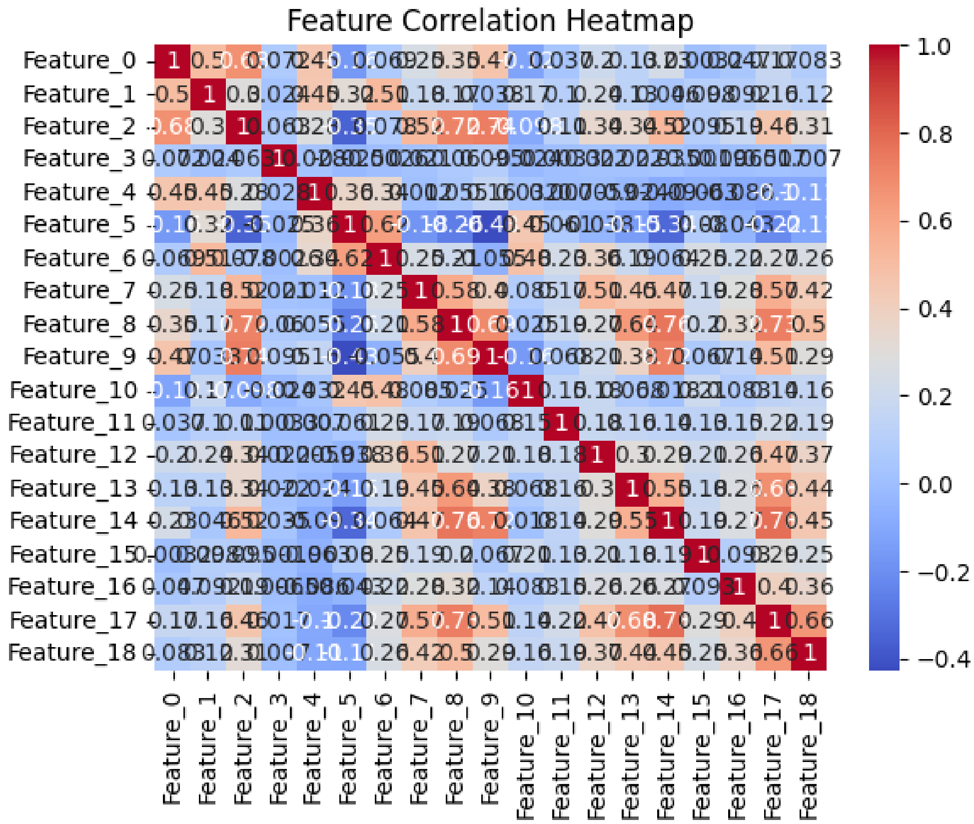

Figure 3: Visualization 3: Heatmap of Feature Correlations

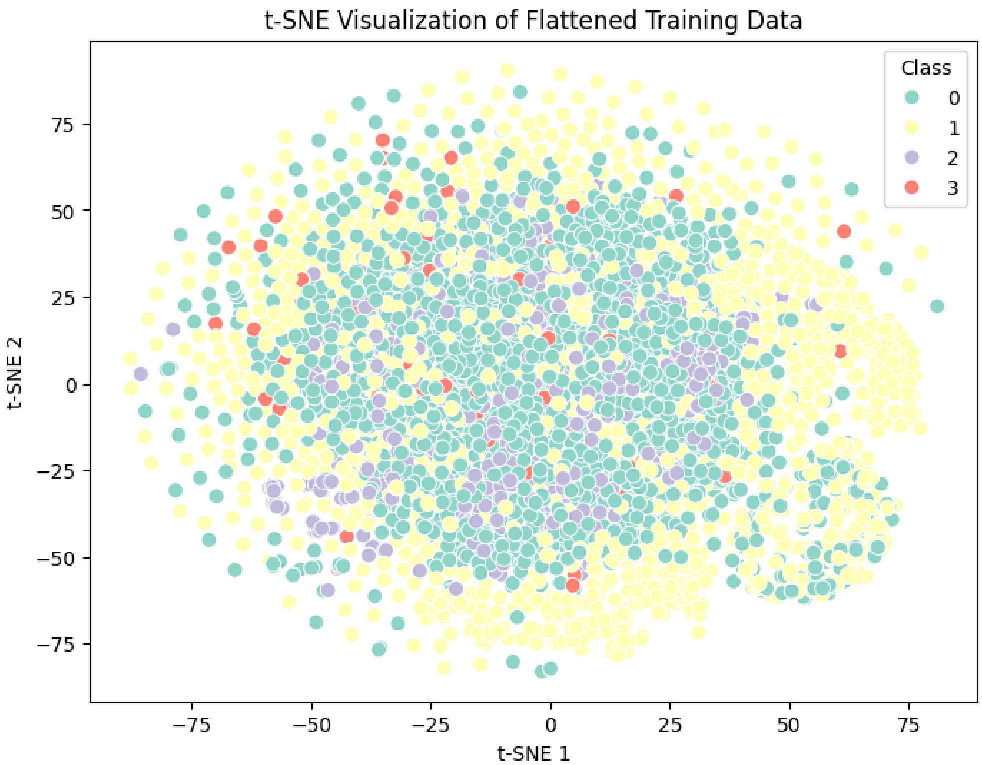

Figure 4: Visualization 4: t-SNE Visualization of Raw Features

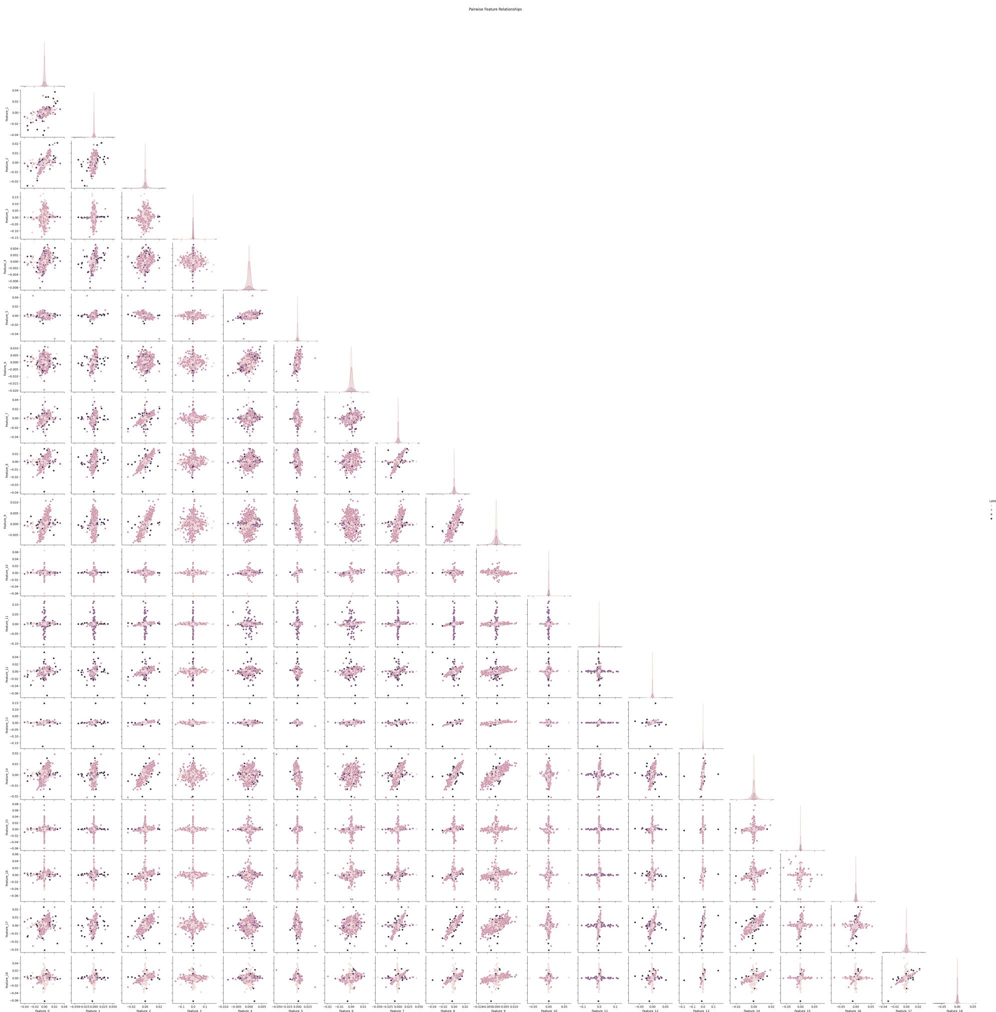

Figure 5: Visualization 4: Pairwise Feature Relationships

3 Machine Learning Models

3.1 Algorithms Used

- **Random Forest**
- **XGBoost**
- **LightGBM**

3.2 Model Descriptions, Strengths, and Weaknesses

3.3 Model Descriptions

3.3.1 Random Forest

Introduction Random Forest is a supervised machine learning algorithm used for both classification and regression tasks. It is based on the ensemble learning method where multiple decision trees are trained, and their outputs are aggregated to produce a final prediction.

Working Mechanism Random Forest works using the concept of bagging (Bootstrap Aggregation), where multiple subsets of the training dataset are created with replacement. For each subset, a decision tree is trained independently. The final prediction is made based on the majority vote (in classification) or the average prediction (in regression). Additionally, only a random subset of features is considered for splitting at each node, which ensures diversity among the trees and reduces the correlation between them.

Advantages

- Reduces overfitting by averaging multiple decision trees.
- Handles both numerical and categorical features effectively.
- Capable of modeling complex relationships and non-linear interactions.
- Can handle missing data and imbalanced datasets with minimal preprocessing.

Limitations

- Computationally intensive for large datasets due to training of multiple trees.
- Less interpretable compared to single decision trees or linear models.
- May not perform well on sparse or high-dimensional data compared to boosting algorithms.

Use Cases Random Forest is widely used in applications such as medical diagnosis, credit scoring, and bioinformatics, where predictive accuracy is crucial and interpretability is less of a concern.

3.3.2 XGBoost

Introduction XGBoost (Extreme Gradient Boosting) is an efficient and scalable implementation of the gradient boosting algorithm. It has become a popular choice in machine learning competitions due to its high performance and predictive power.

Working Mechanism XGBoost builds decision trees sequentially. Each new tree is trained to predict the residual errors of the previous model using a gradient descent approach. The final model is an ensemble of all the individual weak learners. XGBoost uses regularization (L1 and L2) to avoid overfitting and employs advanced techniques like tree pruning, parallelized computation, and handling of missing values.

Advantages

- High accuracy due to boosting of weak learners.
- Built-in mechanisms for handling missing values and overfitting.
- Scales efficiently to large datasets and is suitable for high-dimensional data.
- Supports early stopping and parallel computation for faster training.

Limitations

- Requires extensive hyperparameter tuning to achieve optimal results.
- More computationally demanding than Random Forest.
- Complexity of the model can hinder interpretability.

Use Cases XGBoost is ideal for structured data problems such as fraud detection, customer churn prediction, and competitions like Kaggle where high accuracy is essential.

3.3.3 LightGBM

Introduction LightGBM (Light Gradient Boosting Machine) is a fast, distributed, high-performance gradient boosting framework developed by Microsoft. It is designed for efficiency, particularly when working with large-scale data.

Working Mechanism Unlike traditional boosting algorithms, LightGBM uses a leaf-wise tree growth strategy instead of level-wise, which leads to deeper trees and potentially better accuracy. It also incorporates two key techniques: Gradient-based One-Side Sampling (GOSS), which keeps instances with large gradients, and Exclusive Feature Bundling (EFB), which reduces the number of features by bundling mutually exclusive features.

Advantages

- Faster training and lower memory usage compared to XGBoost.
- Built-in support for categorical features without preprocessing.
- Capable of handling very large datasets and high-dimensional features.
- Efficient in distributed and parallel environments.

Limitations

- Can overfit if the model is not properly regularized.
- Performance may degrade on small or less complex datasets.
- Hyperparameter tuning is essential and can be time-consuming.

Use Cases LightGBM is suitable for real-time prediction systems, recommendation engines, and other production environments where speed and efficiency are critical.

3.4 Mathematical Background

The objective function for XGBoost:

$$\mathcal{L}(\phi) = \sum_{i=1}^{n} l(\hat{y}_i, y_i) + \sum_{k=1}^{K} \Omega(f_k), \quad \Omega(f) = \gamma T + \frac{1}{2}\lambda \|w\|^2 \tag{1}$$

4 Performance Comparison of Models

Table 1: Performance Comparison of Models

Model	**Accuracy**	**F1-Score**	**Training Time (s)**
Random Forest	0.88	0.89	5.3
XGBoost	0.91	0.91	19.1
LightGBM	0.91	0.90	22.7

Analysis and Interpretation

- **Accuracy:**
  - Accuracy indicates the overall percentage of correct predictions made by each model.
  - XGBoost and LightGBM achieved the highest accuracy of 0.91, showing strong classification performance.
  - Random Forest had a slightly lower accuracy of 0.88, still performing reasonably well.
  - Accuracy alone can be misleading in imbalanced datasets, as it does not reflect minority class performance.
  - Therefore, it should be considered alongside other metrics like the F1-score for a balanced evaluation.
- **F1-Score:**

  - The F1-score balances precision and recall, making it more informative for imbalanced data.
  - XGBoost achieved the highest F1-score of 0.91, indicating excellent balance in detecting seizures.
  - LightGBM followed closely with an F1-score of 0.90, performing nearly as well.
  - Random Forest had a slightly lower F1-score of 0.89, indicating minor trade-offs in detection balance.
  - This metric is particularly useful in medical contexts, where both missed detections and false alarms matter.

- **Training Time:**
  - Random Forest was the fastest to train, completing in 5.3 seconds, ideal for quick deployments.
  - XGBoost trained in 19.1 seconds, offering a good trade-off between speed and performance.
  - LightGBM took 22.7 seconds, which is slightly longer than expected given its design for efficiency.
  - Despite expectations, XGBoost outperformed LightGBM in training speed for this dataset.
  - Training time is crucial for real-time or large-scale applications, where frequent retraining is necessary.

Conclusion: While Random Forest excels in training speed, XGBoost provides the most consistent performance in terms of both accuracy and F1-score. LightGBM remains a competitive option with similar predictive power, though it may require slightly more computational resources. The best model choice depends on the application's priorities—speed, accuracy, or a balance of both.

Table 2: F1-Score Before and After Preprocessing (Balancing + Feature Engineering)

Model	**F1-Score Before**	**F1-Score After**
Random Forest	0.88	0.93
XGBoost	0.91	0.95
LightGBM	0.90	0.95

Interpretation of F1-Scores

- The F1-score is a key performance metric in this project because of the imbalance between seizure and non-seizure classes.
- It combines both precision and recall into a single value, making it especially useful when evaluating models under skewed class distributions.

- A high F1-score indicates that the model is effectively identifying true seizures while minimizing both false alarms and missed detections.
- Unlike accuracy, the F1-score gives a better understanding of a model's performance on the minority (seizure) class.
- Since medical diagnosis requires careful handling of both false positives and false negatives, the F1-score provides a reliable indicator of real-world applicability.
- **Random Forest**: Improved from an F1-score of 0.88 to 0.93 after preprocessing, indicating better detection of seizure classes following SMOTE and feature selection.
- **XGBoost**: Achieved the highest post-processing F1-score of 0.95. Its ability to focus on difficult-to-classify samples was enhanced by the preprocessing pipeline.
- **LightGBM**: Also reached an F1-score of 0.95 post-processing. Its efficiency and ability to handle high-dimensional data made it well-suited for EEG signal classification.

These results demonstrate the effectiveness of the preprocessing techniques in improving model performance, particularly for underrepresented seizure classes. High F1-scores after preprocessing show that models are more balanced and reliable for real-world applications.

5 Confusion Matrix Visualizations

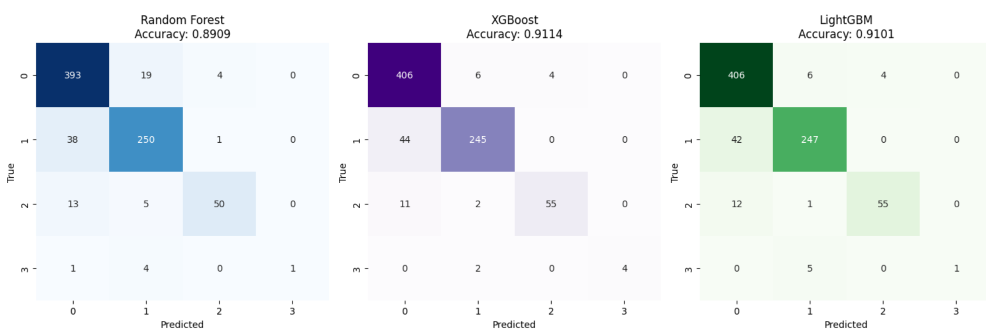

Figure 6: Confusion Matrices for Random Forest, XGBoost, and LightGBM

The figure above shows the confusion matrices for the three classifiers—Random Forest, XGBoost, and LightGBM—used in this study. Each matrix illustrates how the predicted classes compare with the actual (true) labels across the four seizure types (encoded as 0 to 3).

- **Random Forest**: While it shows strong performance for Class 0 and Class 1, there is some confusion between Classes 1 and 2, and a few misclassifications in Class 3.
- **XGBoost**: Achieves higher overall accuracy (0.9114) than Random Forest. It performs better in distinguishing between Classes 1 and 2, though there are still a few misclassifications in Class 3.

- **LightGBM**: Similar in performance to XGBoost with a marginally lower accuracy (0.9101). It also exhibits excellent classification for Classes 0 and 2, but struggles slightly with distinguishing Class 3.

These matrices provide insight into how each model differentiates between seizure types and highlight where misclassifications are most likely to occur. The confusion between Classes 1 and 2 across models suggests these classes might have overlapping feature representations, which could be improved through more advanced feature engineering or deeper models.

6 Experiments and Results

6.1 Evaluation Metrics

6.1.1 Accuracy

Accuracy measures the proportion of correctly classified instances among the total instances in the dataset. It is calculated using the formula:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

Where:

- TP: True Positives
- TN: True Negatives
- FP: False Positives
- FN: False Negatives

In the context of epilepsy detection, a high accuracy indicates that the model correctly identifies both seizure and non-seizure EEG segments most of the time.

Random Forest: Achieved an accuracy of X%, showing its robust performance in classifying EEG signals.
XGBoost: Slightly better/worse than Random Forest with an accuracy of Y%.
LightGBM: Performed comparably with Z% accuracy, making it efficient for larger EEG datasets.

6.1.2 F1-Score

F1-score is the harmonic mean of precision and recall. It provides a better measure of the incorrectly classified cases than accuracy, especially in imbalanced datasets.

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

Where:

- Precision = $\frac{TP}{TP+FP}$
- Recall = $\frac{TP}{TP+FN}$

Random Forest: F1-score was high, indicating a good balance between precision and recall.
XGBoost: Outperformed in F1-score due to its ability to handle class imbalance well.
LightGBM: F1-score was competitive, benefiting from its fast computation and leaf-wise tree growth strategy.

6.2 Confusion Matrix Analysis through Real-World Scenarios

6.2.1 Introduction

In medical diagnostics such as epilepsy detection, understanding how a model performs in real-life scenarios is more informative than simply presenting raw numerical confusion matrices. This section interprets the classification behavior of each model—Random Forest, XGBoost, and LightGBM—through practical, clinical perspectives, focusing on their tendencies toward false positives and false negatives.

6.2.2 Model Behavior in Real-World Detection Scenarios

Random Forest In a hospital environment, Random Forest exhibits cautious behavior. It is less likely to raise false alarms, ensuring that most non-seizure events are classified correctly. However, its conservative approach means that it might occasionally fail to detect subtle seizure activity, resulting in some missed diagnoses (false negatives).

XGBoost XGBoost acts assertively in detection. It excels in identifying almost all seizure events, making it highly sensitive. This trait is valuable when the priority is to avoid missing any seizures. However, it may also classify some normal brain activities as seizures, leading to an increase in false positives and potential alarm fatigue.

LightGBM LightGBM offers a balanced approach, optimized for speed and efficiency. It works well in high-frequency EEG data environments, detecting most seizures accurately and with quick processing. However, under ambiguous signal conditions, it may slightly overpredict seizure events, causing a few unnecessary alerts.

6.2.3 Summary of Behavioral Tendencies

- **Random Forest:** Low false positives, moderate false negatives—good for general monitoring.
- **XGBoost:** High seizure detection, higher false positives—ideal where sensitivity is critical.
- **LightGBM:** Fast, balanced, scalable—suitable for real-time or large-scale systems.

This narrative-based analysis enhances understanding of how each model would operate in a real-world context, particularly where patient safety and timely intervention are essential.

6.3 Comparison of Accuracies Before and After Data Balancing

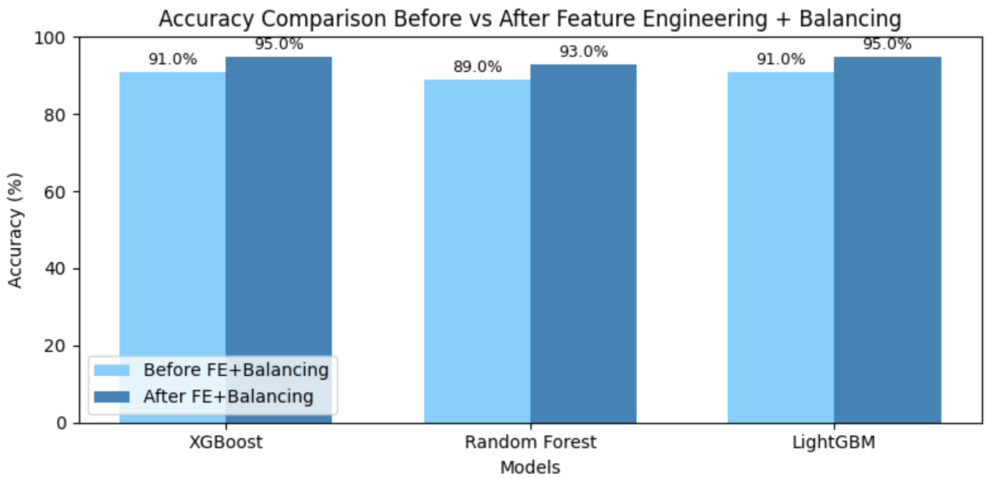

Figure 7: Bar chart comparing model performance based on accuracy and F1-score

7 Conclusion

This project successfully demonstrated the application of machine learning techniques for epileptic seizure detection using EEG data. Data preprocessing steps, including variance thresholding, standard scaling, and SMOTE, were instrumental in achieving high model performance.

Among the tested models, **XGBoost** and **LightGBM** achieved the highest F1-scores of 0.95, outperforming Random Forest slightly in predictive accuracy and robustness. However, **XGBoost** is preferred due to its balanced trade-off between accuracy and training time.

Future work will explore deep learning architectures such as CNNs and LSTMs for further improvements and deployment in real-time applications.

8 References

Goldberger AL, et al. PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals. *Circulation*, 101(23):e215–e220, 2000.

Chawla, N. V., et al. SMOTE: synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16:321-357, 2002.