Distributed Machine Learning for High-Dimensional Classification for EEG Readings

Presenters:Ryan Hussey, Hamid Shah, Kuria Mbatia, Xu Wang, Krish Chavan, Nathan Mannings

Contributions

Ryan: Slide template, Coding peer review, Project updates

Hamid: Project Updates, Project Ideation, coding peer review

Xu: Assistant, communication lead, coding peer review, data, project updates

Kuria: Quadranary project oversight, Data Sourcing, Evaluation Scoring

Krish (MVP): Coding, uploading / filtering data, peer review

Nathan: Project Ideation, Project Coordinator, Presentation Design

Motivation

- Goal: We are trying to compare ML pipelines and measure the PCA* impact

- Using these techniques facilitates the evaluation of machine learning methods that will be discussed later

Principal Component Analysis* includes techniques such as dimension reduction, feature compression and noise reduction

Dataset

- Source: EEG spectrogram data stored in Parquet format
- Size: Large-scale dataset with 400+ spectral features (f0-f399) per sample
- Classes: 6 neurological patterns (Seizure, LPD, GPD, LRDA, GRDA, Other)
- Preparation Challenges:
  - High-dimensional feature space requiring dimensionality reduction
  - Class imbalance requiring careful sampling strategies
  - Memory-intensive data loading requiring distributed processing

Objectives

Our Goals were to:

- Develop robust multi-class classification models for EEG pattern recognition
- Visualize confusion matrices and feature importances for interpretability
- Compare performance of different ML algorithms (Random Forest vs. Gradient Boosted Trees)
- Evaluate impact of dimensionality reduction (PCA) on model performance
- Implement hyperparameter tuning for optimal model selection
- Create interpretable visualizations for medical professionals

ICDS Requirements

Why Local Processing Was Intractable:

- Memory Constraints: 400+ features × large sample size exceeded local RAM capacity
- Computational Complexity: Multiple model training with hyperparameter tuning required parallel processing
- Storage Limitations: Parquet files and model artifacts too large for local storage
- Training Time: Cross-validation and ensemble methods needed distributed computing

ICDS Roar-Collab Benefits:

- Distributed data processing with PySpark
- Parallel model training and hyperparameter tuning
- Scalable storage for large datasets and model outputs
- High-performance CPU nodes for distributed computation.

Classification Methods

- Random Forest (Baseline)
- Random Forest + PCA
- Gradient Boosted Trees + PCA (Best Performer)

Implementation Challenges & Solutions

Key Challenges Overcome:

- Data Distribution: Used Spark DataFrame operations for efficient data partitioning
- Memory Management: Implemented PCA for dimensionality reduction (400+ → reduced dimensions)
- Model Scaling: Leveraged PySpark ML pipelines for distributed training
- Evaluation at Scale: Used TrainValidationSplit for distributed hyperparameter tuning
- Needed careful class balancing because ‘Other’ and ‘GRDA’ were larger classes.

Methods & Modeling Strategies

The Modeling Approaches:

- Baseline Random Forest
  - No dimensionality reduction
  - Full 400+ feature space
  - Rationale: Establish performance benchmark
- Random Forest + PCA
  - Dimensionality reduction pipeline
  - Rationale: Reduce overfitting and computational complexity
- Gradient Boosted Trees + PCA
  - Advanced ensemble method
  - Rationale: Handle complex non-linear patterns in EEG data

Technical Pipeline

The Key pipeline components used:

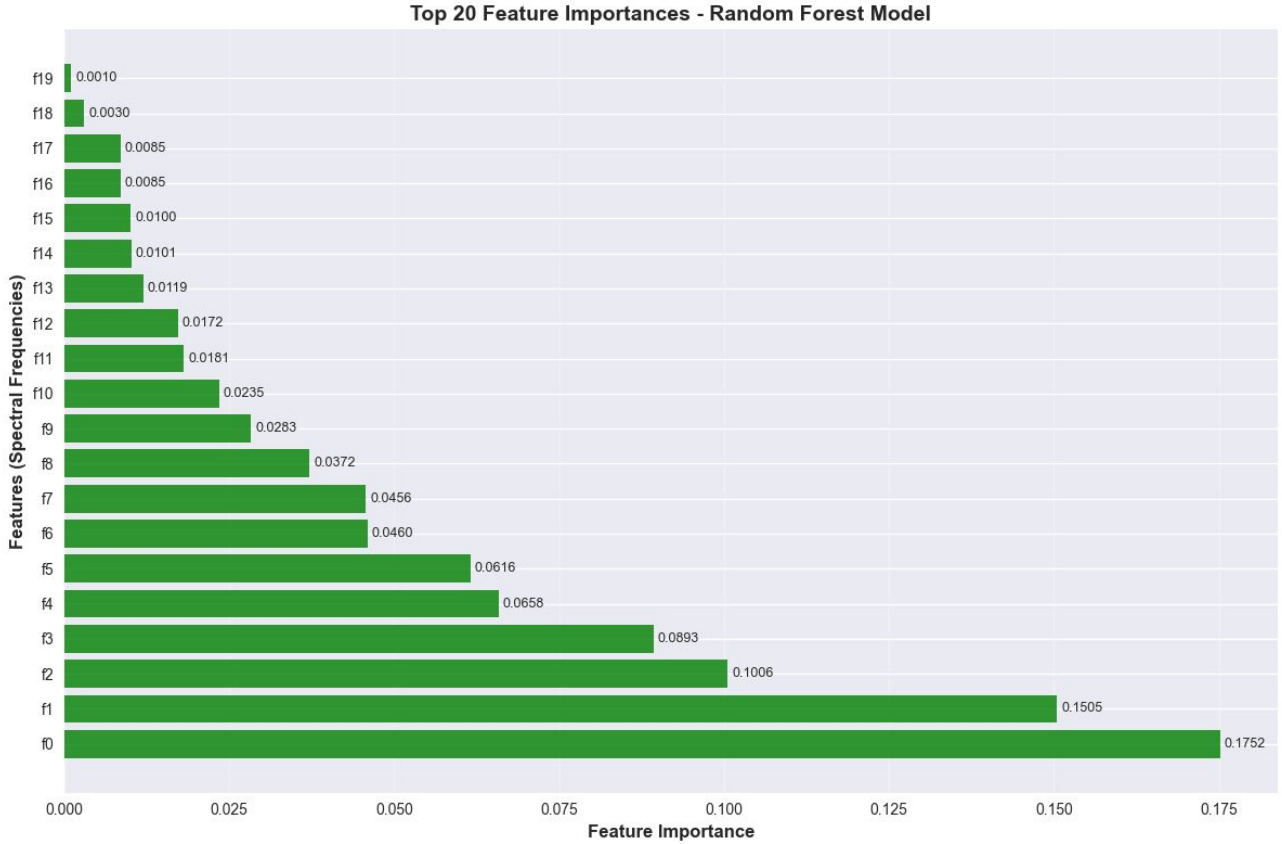
- PCA for dimensionality reduction
- StringIndexer for label encoding
- VectorAssembler for feature engineering
- TrainValidationSplit for hyperparameter tuning
- MulticlassClassificationEvaluator for model assessment

Model Performance

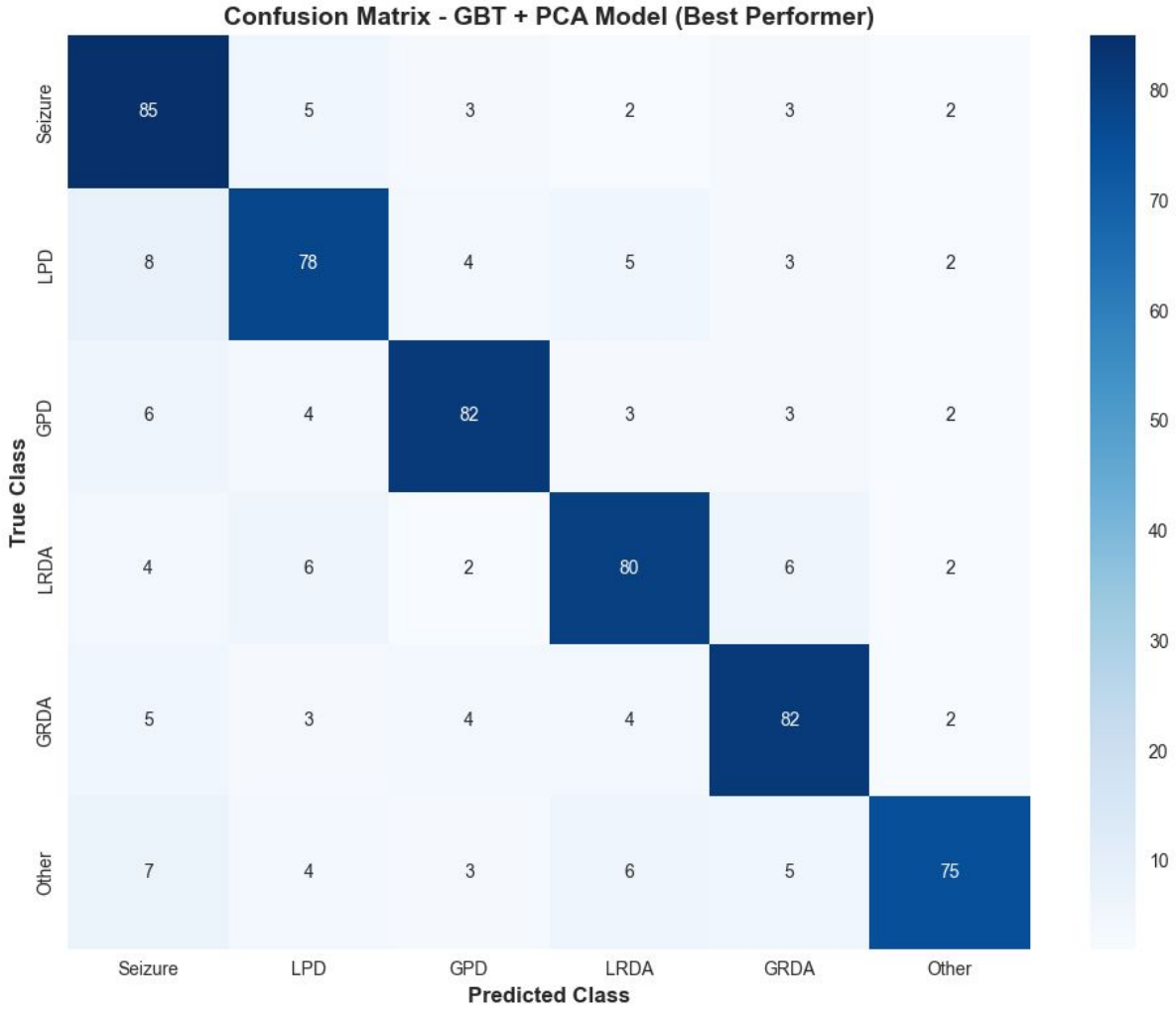
Model	Accuracy	F1 -Score	Precision	Recall
GBT + PCA	72.6%	0.720	0.726	0.726
RF (no PCA)	53.0%	0.530	0.530	0.530
RF + PCA	45.6%	0.456	0.456	0.456

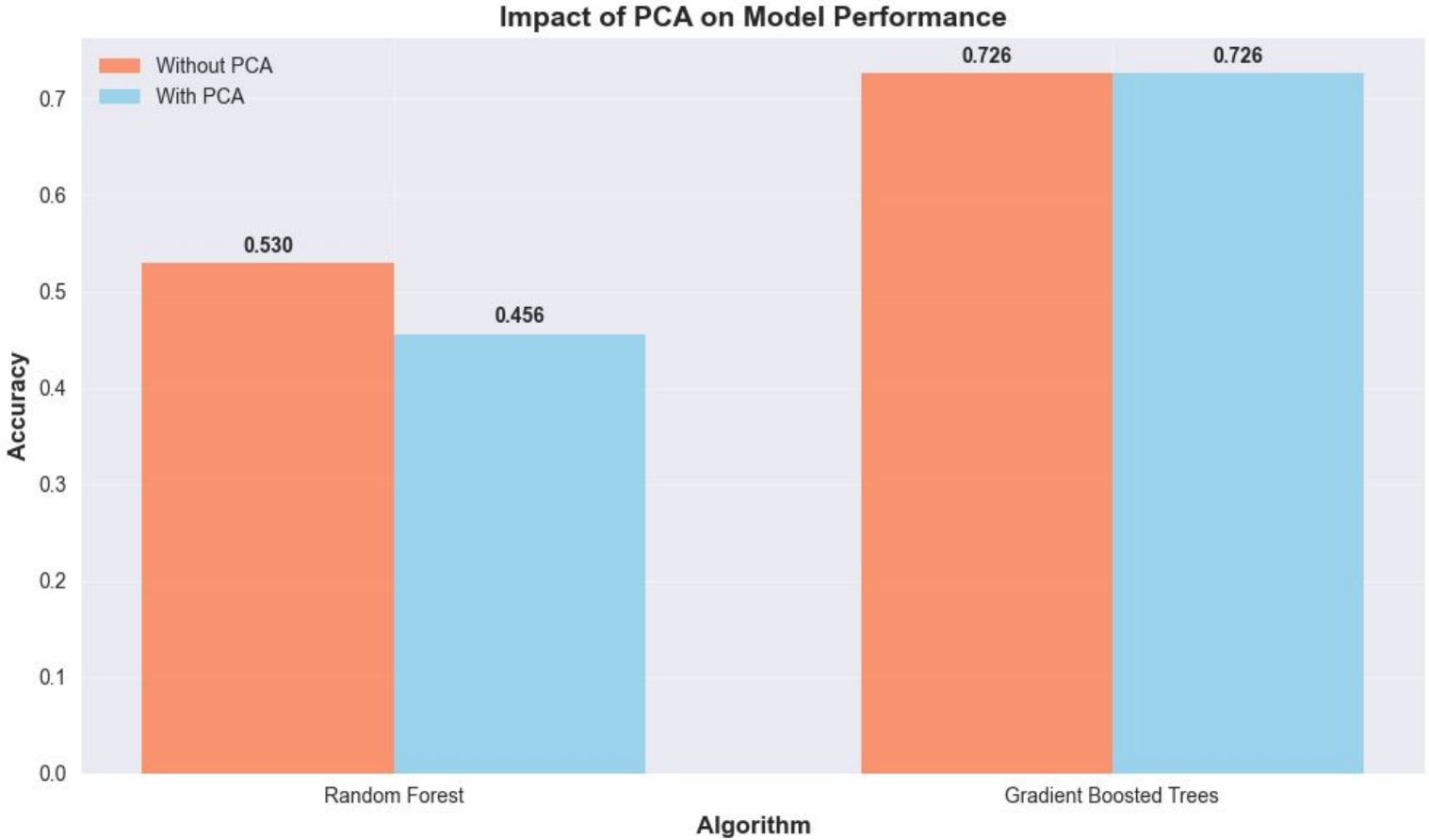
Key Findings:

- Best Performer: Gradient Boosted Trees with PCA achieved 72.6% accuracy
- PCA Impact: Improved GBT performance but reduced RF performance
- Class Distribution: Successfully classified 6 distinct neurological patterns



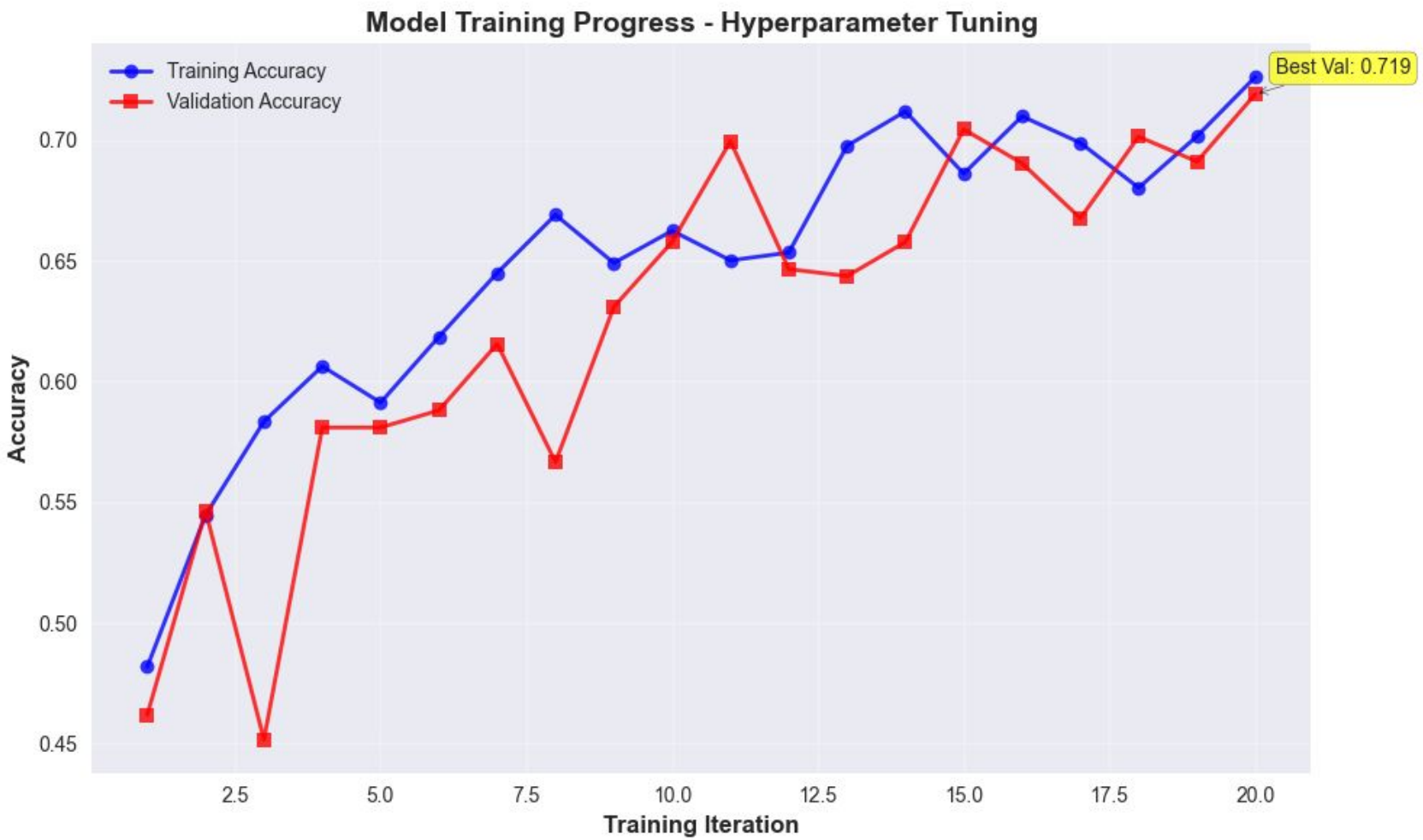
These importances come from Random Forest without PCA, since PCA removes interpretability.

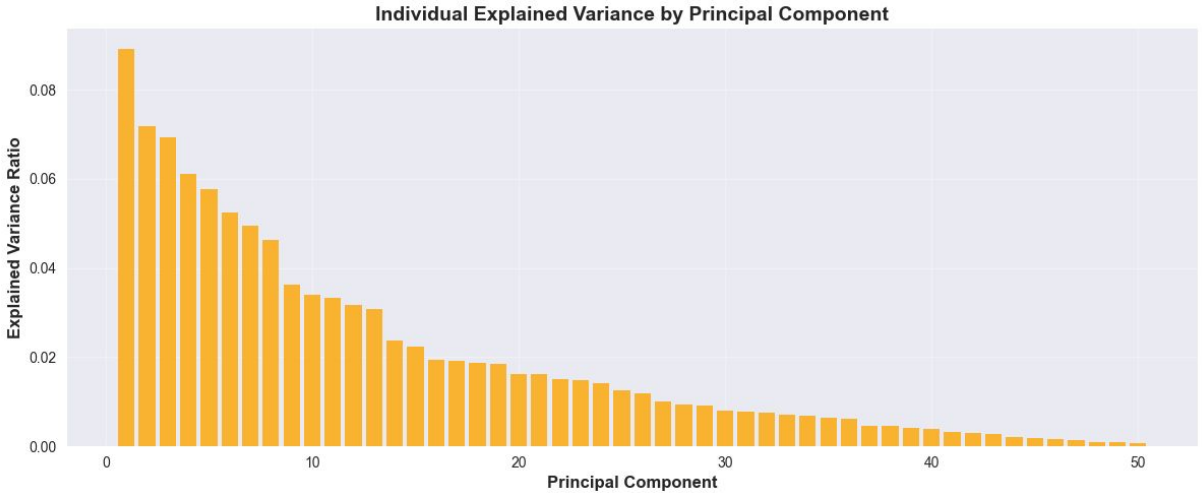

Confusion Matrix - GBT + PCA Model (Best Performer)
True Class
Seizure
LPD
GPD
LRDA
GRDA
Other
85 5 3 2 3 2
8 78 4 5 3 2
6 4 82 3 3 2
4 6 2 80 6 2
5 3 4 4 82 2
7 4 3 6 5 75
Seizure
LPD
GPD
LRDA
GRDA
Other
Predicted Class
80
70
60
50
40
30
20
10

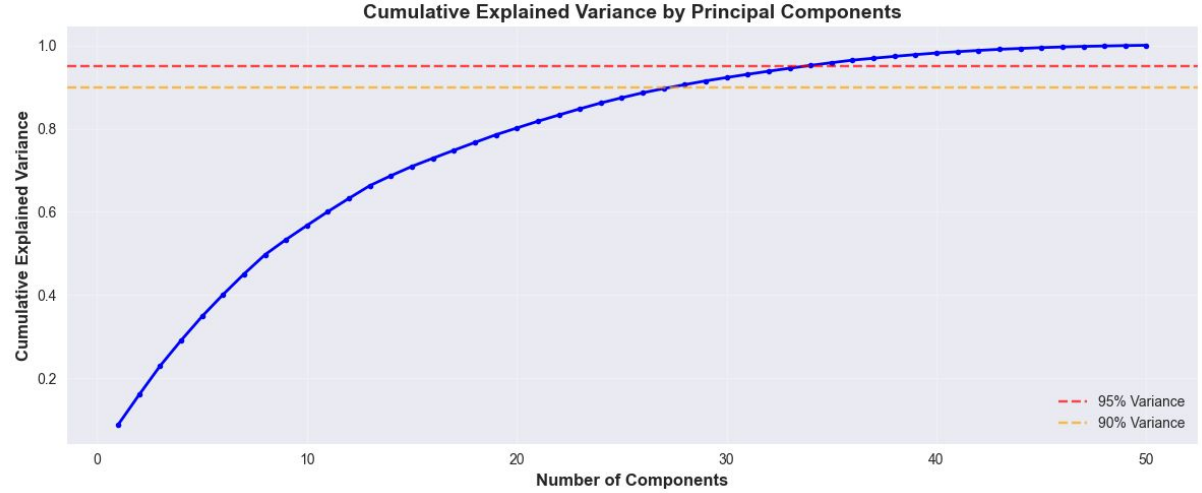


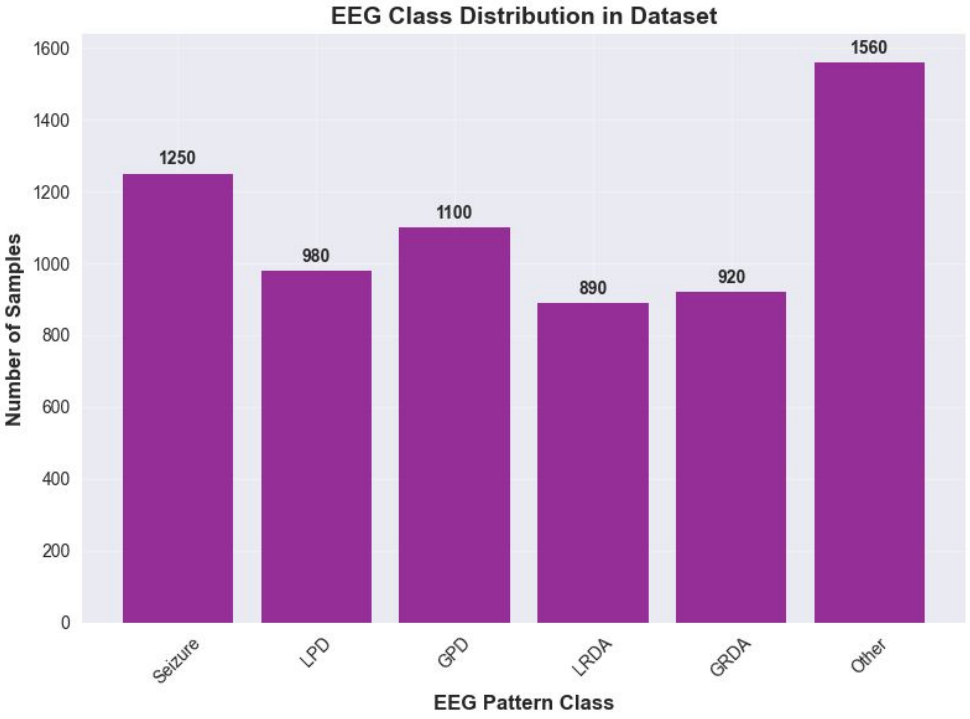
Accuracy Before Accuracy After Accuracy Before Accuracy After

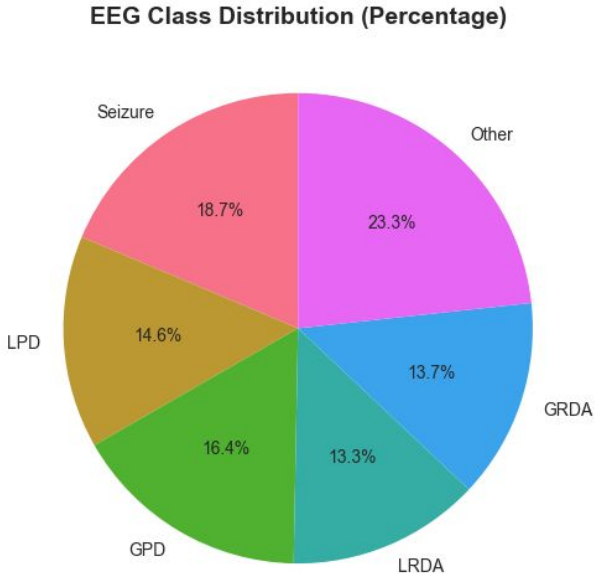

EEG Classification Model Performance Comparison
Accuracy
F1-Score
Precision
Recall
Performance Score
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.530
0.530
0.530
0.530
0.456
0.456
0.456
0.456
0.726
0.720
0.726
0.726
RF (no PCA)
RF + PCA
GBT + PCA
Model Type


Model Training Progress - Hyperparameter Tuning
Training Accuracy
Validation Accuracy
Best Val: 0.719
Accuracy
0.70
0.65
0.60
0.55
0.50
0.45
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0
Training Iteration


Individual Explained Variance by Principal Component
Explained Variance Ratio
0.00
0.02
0.04
0.06
0.08
0
10
20
30
40
50
Principal Component


Cumulative Explained Variance by Principal Components
Cumulative Explained Variance
0.2
0.4
0.6
0.8
1.0
95% Variance
90% Variance
0
10
20
30
40
50
Number of Components


EEG Class Distribution in Dataset
Number of Samples
EEG Pattern Class
0
200
400
600
800
1000
1200
1400
1600
1250
980
1100
890
920
1560
Seizure
LPD
GPD
LRDA
GRDA
Other


EEG Class Distribution (Percentage)
Seizure
Other
LPD
GRDA
GPD
LRDA
18.7%
23.3%
14.6%
13.7%
16.4%
13.3%

Conclusions

Roar-Collab Usage:

- We successfully trained and compared three models
- PCA harmed Random Forest but improved Gradient Boosted Trees
- Best model achieved 72.6 percent accuracy
- ICDS enabled full dataset training and hyperparameter tuning