WINTER PROJECT MIDTERM REPORT

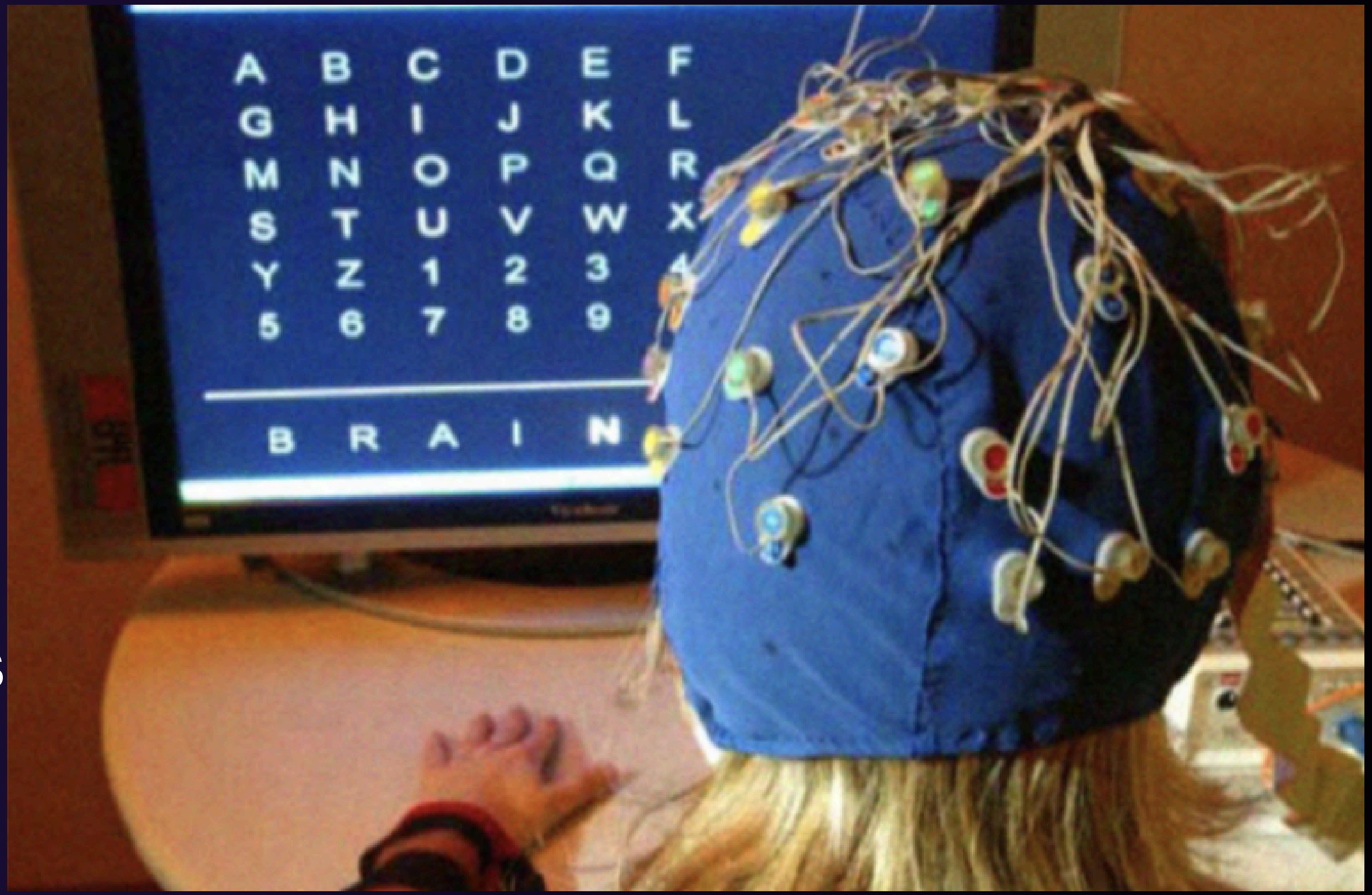
PROJECT NAME : EEG – BASED P300 SPELLER

NAME : AMIT SINGH

ROLL NO. : 240109

SUMMARY OF THE PROJECT

- This project is about building a P300-based EEG speller, which is a type of Brain to Computer Interface where a computer tries to understand what character a user wants to select just by looking at their brain signals.
- The main idea is that when a person sees the character they are focusing on, the brain produces a small response called the P300, and we can detect this from EEG data.



- The procedure involves capturing raw EEG signals from the brain, cleaning them, and then look for this P300 response to understand whether the computer is showing the letter the user is thinking of or not using machine learning.

WORK COMPLETED

1. **Introduction to EEG and P300 speller**

- EEG is a way to record brain activity by placing electrodes on the scalp, which capture very small electrical signals produced. It is fast in time.
- Signals contain different brain rhythms (like delta, theta, alpha, beta, and gamma), and each rhythm is linked to a mental state such as sleep, relaxation, or active thinking.
- An ERP (Event-Related Potential) is a brain response that happens after a specific event, such as seeing a flash or hearing a sound.
- The P300 is a special ERP component that appears as a positive peak around 300 milliseconds when a person notices something important or rare.
- P300 spellers use this brain response to detect which letter a user is focusing on, allowing communication without speaking or moving, which is especially useful for people with severe motor disabilities.

WORK COMPLETED

2. Machine Learning

Machine learning is about teaching computers to learn from data, instead of writing fixed rules for every situation.

Core components:

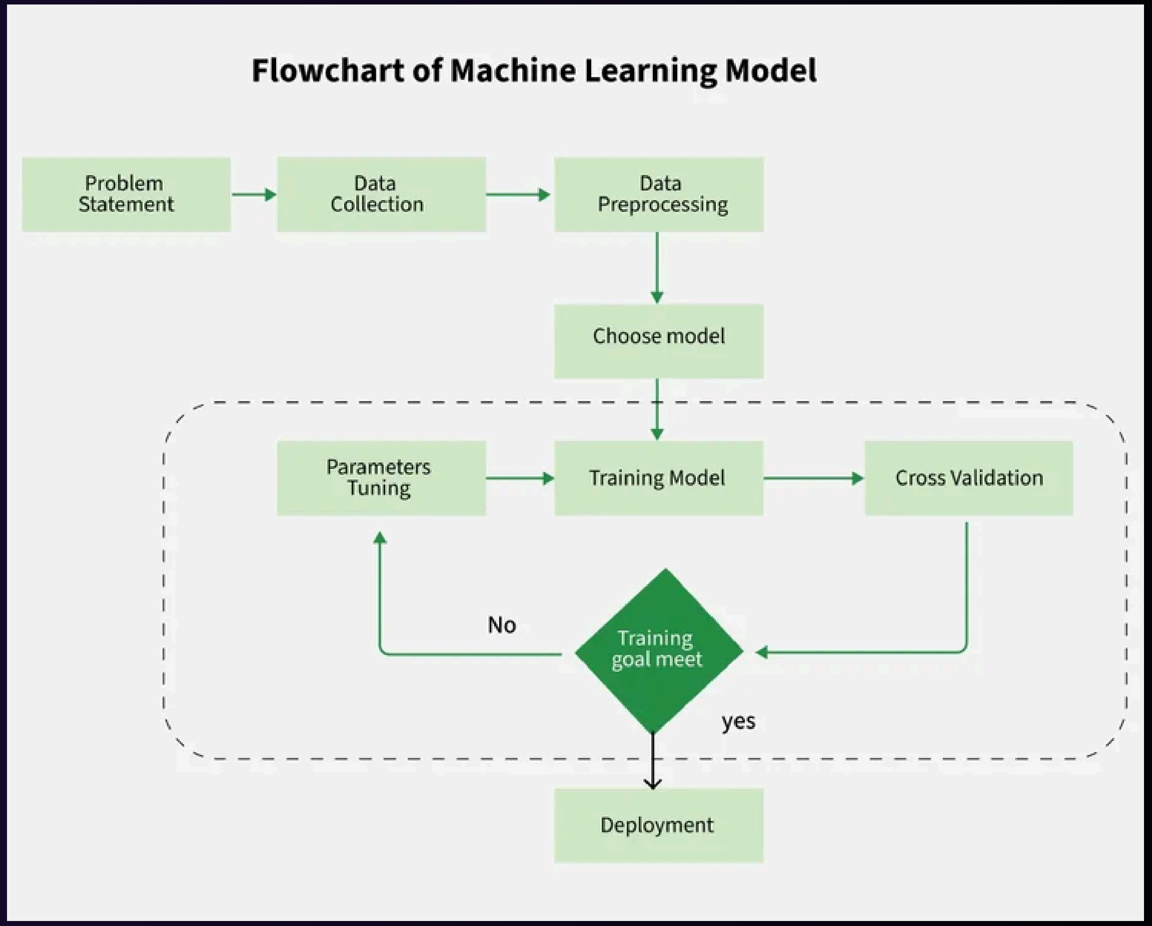
- Data → Features (inputs) & Labels (outputs)
- Model → Function mapping inputs to outputs
- Loss function → Measures prediction error
- Optimization → Adjusting parameters to minimize loss

Types of ML:

- Supervised Learning (labeled data)
- Unsupervised Learning (unlabeled data)
- Reinforcement Learning (reward-based learning)

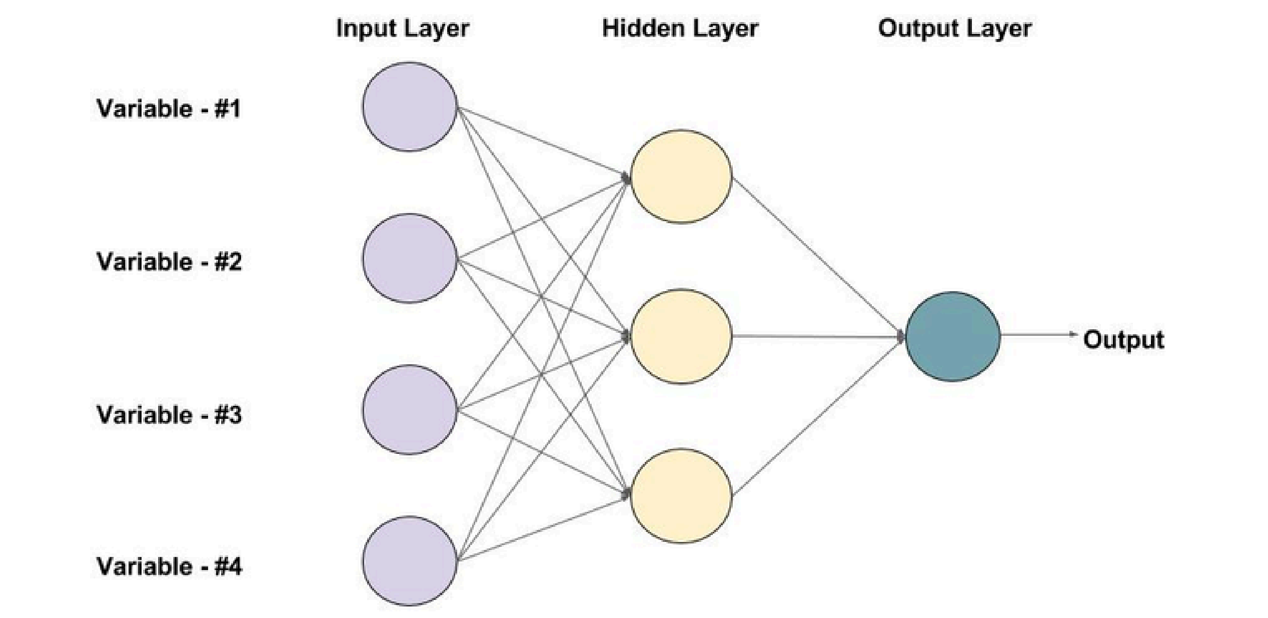
Supervised Learning tasks:

- Regression → Continuous outputs
- Classification → Discrete class labels



WORK COMPLETED

- Neural Networks consist of layers of neurons performing weighted sums followed by nonlinear activation
- Feedforward Neural Networks:
  - Information flows only forward
  - Used for fixed-size input/output problems
- Softmax activation converts raw scores into class probabilities
- Recurrent Neural Networks (RNNs):
  - Designed for sequential / time-dependent data
  - Maintain a hidden state to capture past information
- Challenges in RNNs:
  - Vanishing and exploding gradients
- LSTM networks:
  - Introduce gating mechanisms to preserve long-term dependencies
  - Gates: Forget, Input, Output



An example of a Feed-forward Neural Network with one hidden layer (with 3 neurons)

WORK COMPLETED

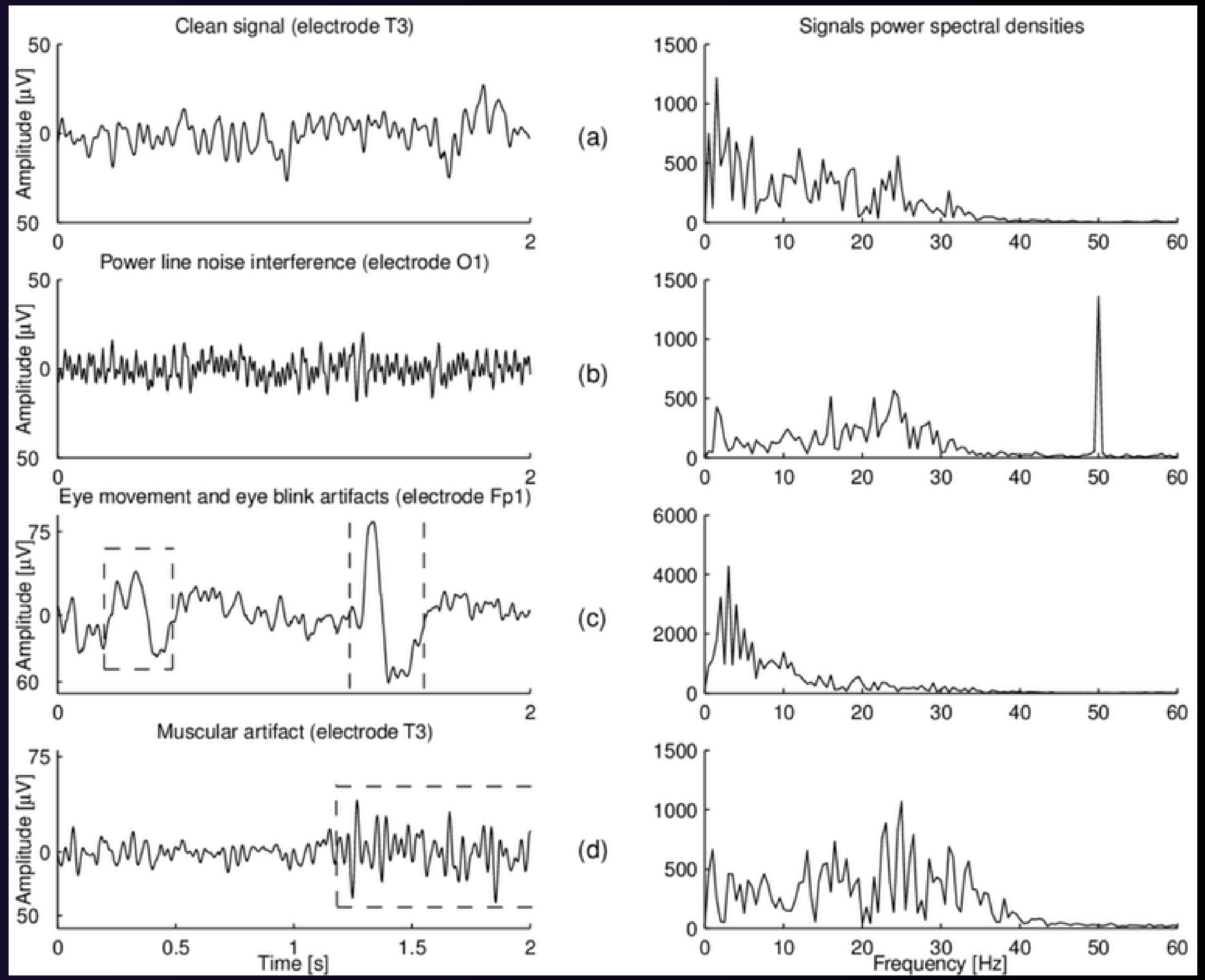
3. **Noise in EEG Signals**

- EEG signals are low amplitude (5–100 µV) and highly susceptible to noise
- Noise refers to unwanted signals not originating from brain activity
- Major noise sources:
  - Power-line interference (50 Hz / 60 Hz)
  - Electrode impedance mismatch
  - Environmental electromagnetic noise
  - Thermal and amplifier noise
- Noise can overlap with EEG frequency bands, making removal challenging
- Observed EEG signal

  $x(t)=s(t)+n(t)$

  $s(t)$ = true brain signal

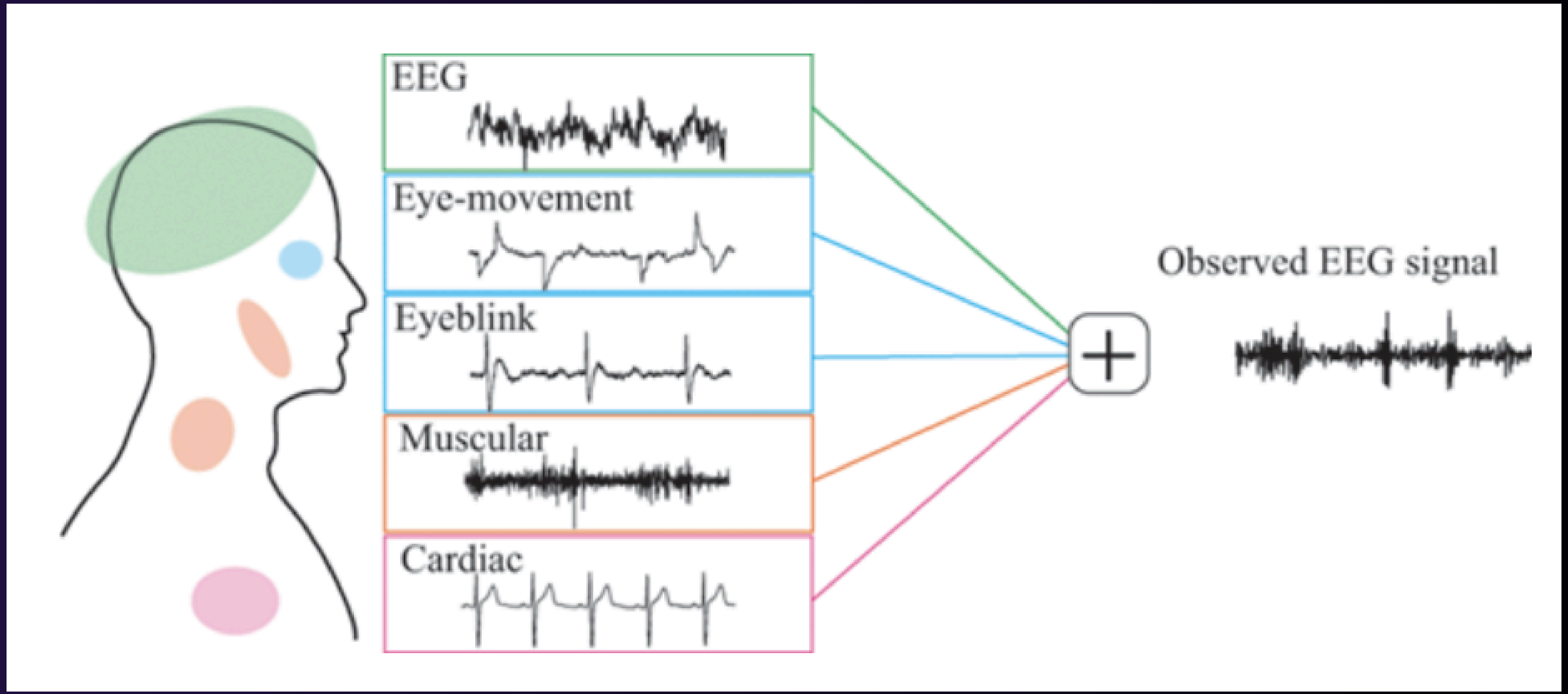
  $n(t)$ = noise



WORK COMPLETED

4. **Artifacts in EEG Recordings**

- Artifacts are structured signals that contaminate EEG but are not neural
- Physiological artifacts:
  - Eye blinks & eye movements
  - Muscle activity (EMG)
  - Cardiac activity (ECG)
- Non-physiological artifacts:
  - Cable movement
  - Electrode pops
  - Subject motion

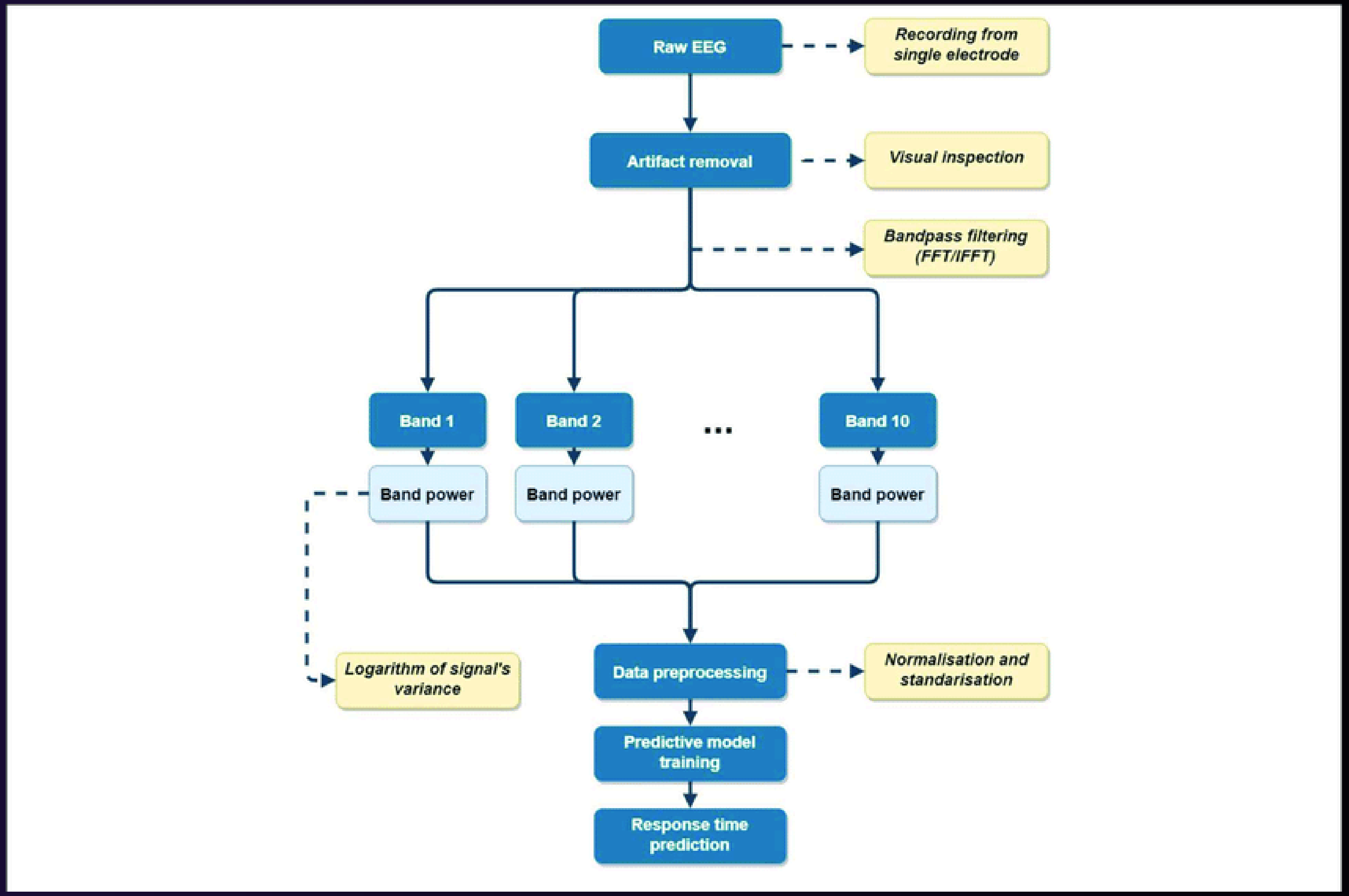


- Artifacts often have much higher amplitude than EEG
- They distort ERPs and affect downstream analysis
- Artifacts are not random noise, hence require targeted correction, not simple filtering

WORK COMPLETED

5. **Standard EEG Processing Pipeline**

- Preprocessing improves data quality and interpretability
- Typical pipeline:
- Raw EEG acquisition
- Filtering (remove unwanted frequencies)
- Resampling (reduce computational load)
- Event extraction
- Epoching (time-locking to events)
- Artifact correction
- Averaging → ERP extraction
- Each step progressively increases signal clarity
- Poor preprocessing leads to misleading neural interpretations



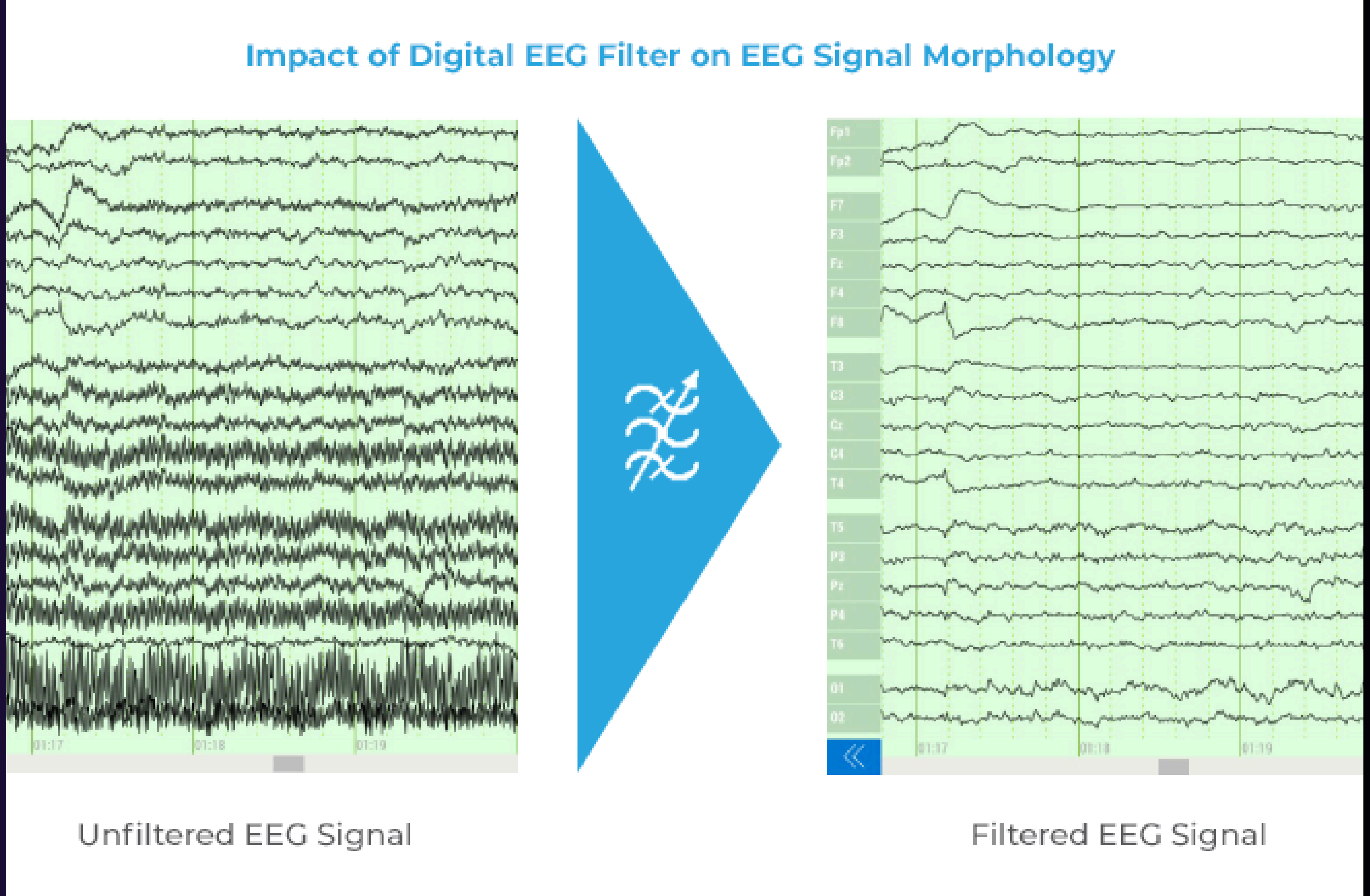
WORK COMPLETED

6. **Filtering, Resampling & Event Handling**

- Filtering removes frequency-specific noise:
  - High-pass → remove slow drifts
  - Low-pass → remove muscle noise
  - Notch → remove power-line noise
- Resampling:
  - Reduces sampling rate while preserving info
  - Improves computational efficiency
- Event detection:
  - Identifies stimulus markers in EEG
  - Enables time-locked analysis
- Proper event alignment is critical for ERP analysis
- Filtering Expression

  $y(t)=x(t) * h(t)$

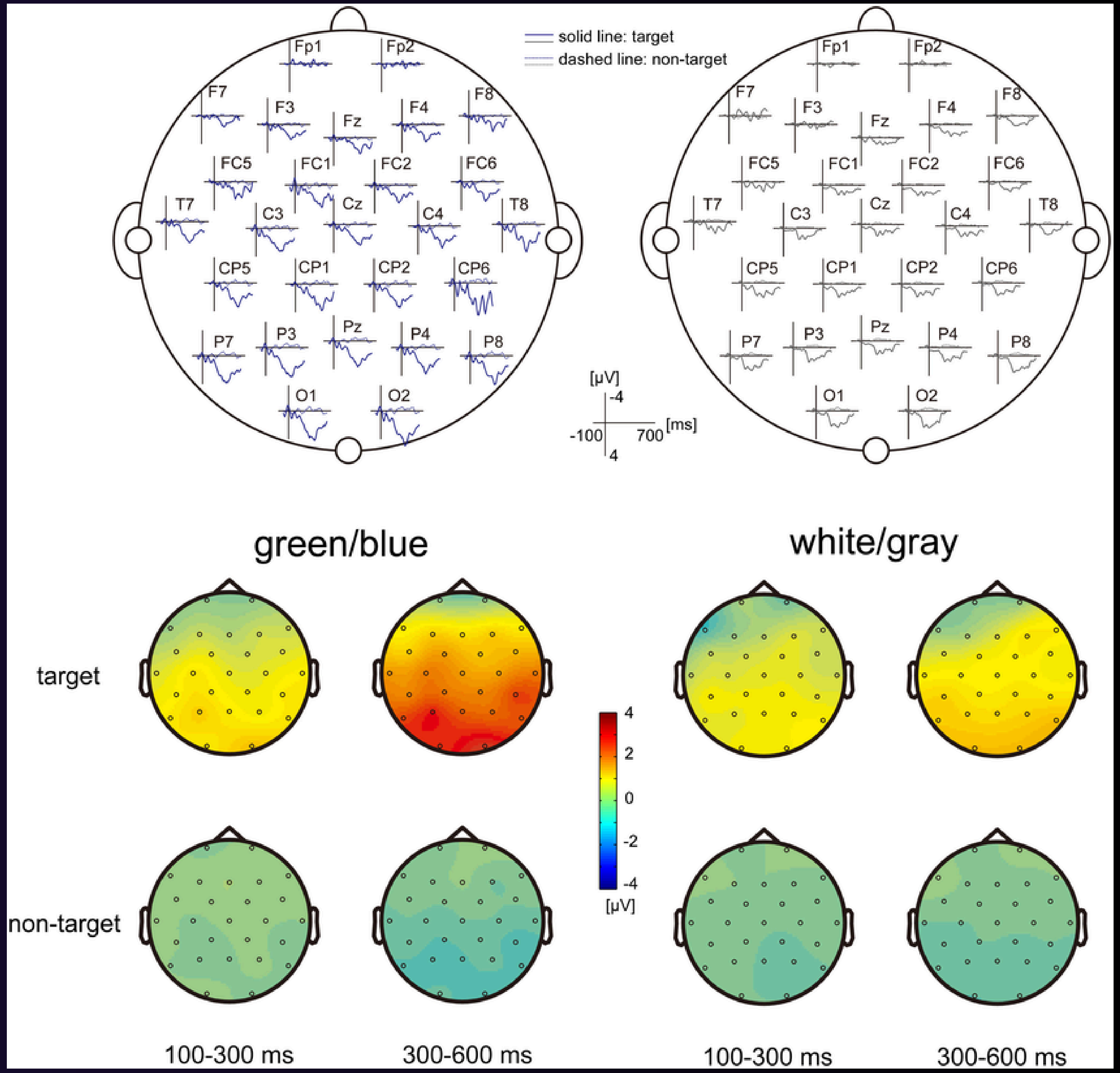
  where h(t) is the filter impulse response



WORK COMPLETED

7. **Artifact Correction, ICA & Visualization**

- Independent Component Analysis (ICA) separates EEG into independent sources
- Components corresponding to artifacts can be identified and removed
- ICA assumes:
  – Linear mixing
  – Statistical independence of sources
- Post-ICA EEG is cleaner and preserves neural signals
- Topographic visualization helps understand spatial distribution of brain activity
- Evoked response maps show how activity evolves across the scalp

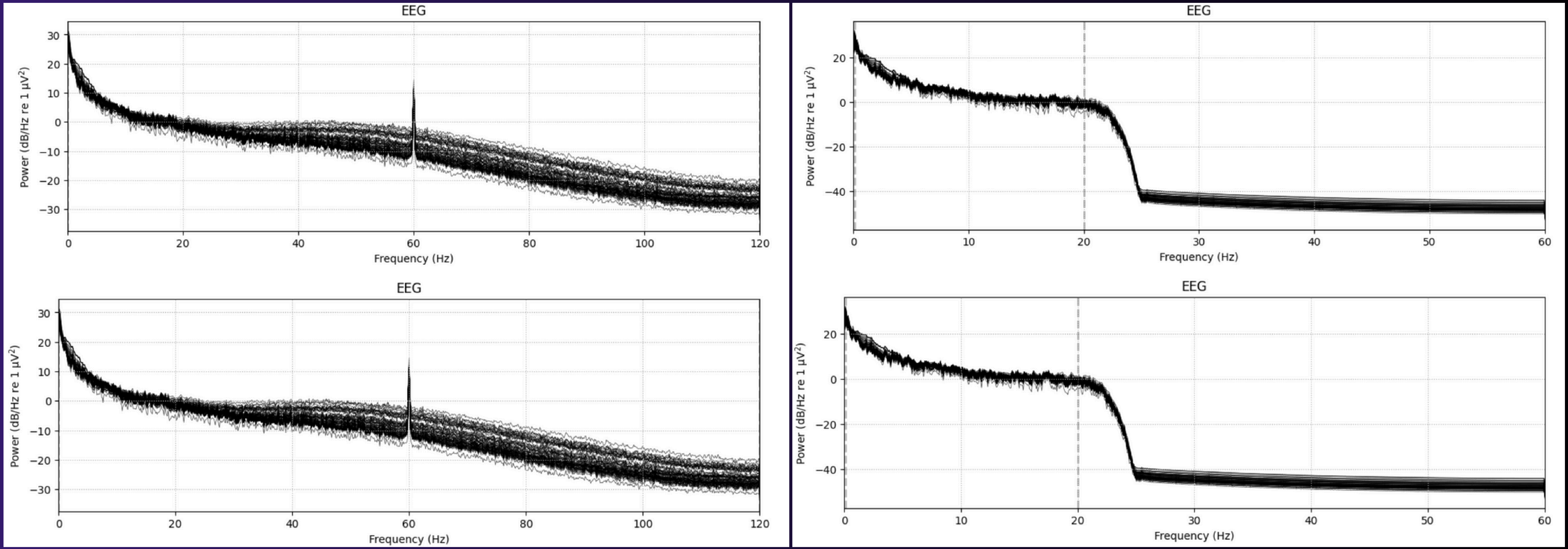


ASSIGNMENT AND CODE LINKS

- Assignment 0 : https://github.com/amits248/Winter-projects-25-26/blob/main/EEG-Based%20P300%20Speller/assignments/assignment_0/240109_Amit_Singh.ipynb
- Assignment 1 : https://github.com/amits248/Winter-projects-25-26/blob/main/EEG-Based%20P300%20Speller/assignments/assignment_1/240109_Amit_Singh.pdf
- Assignment 2 : https://github.com/amits248/Winter-projects-25-26/blob/main/EEG-Based%20P300%20Speller/assignments/assignment_2/240109_Amit_singh_assignment2.ipynb
- Assignment 3 : https://github.com/amits248/Winter-projects-25-26/blob/main/EEG-Based%20P300%20Speller/assignments/assignment_3/EEG_assignment3_Amit_Singh_240109.ipynb

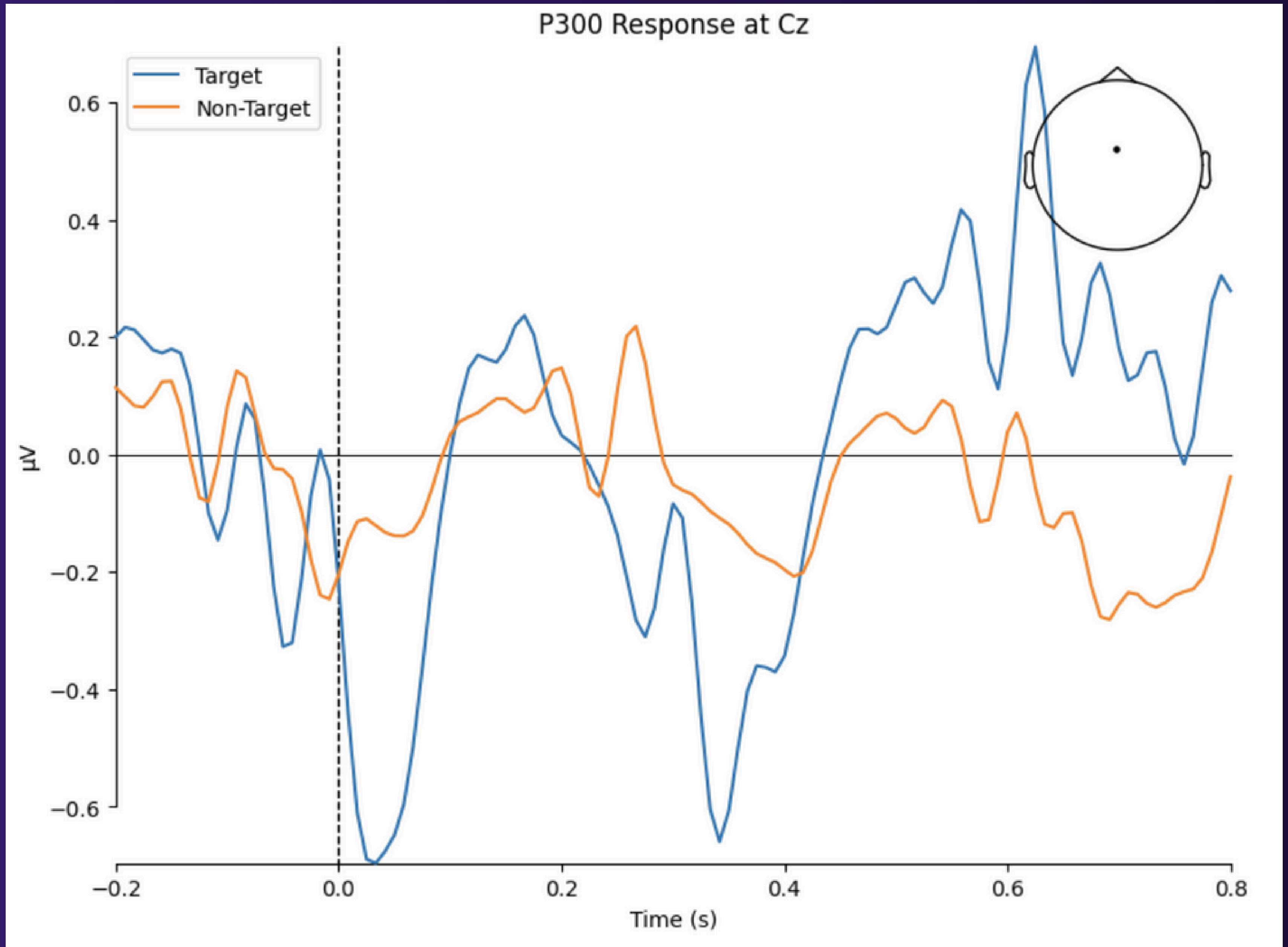
RESULTS AND OBSERVATIONS

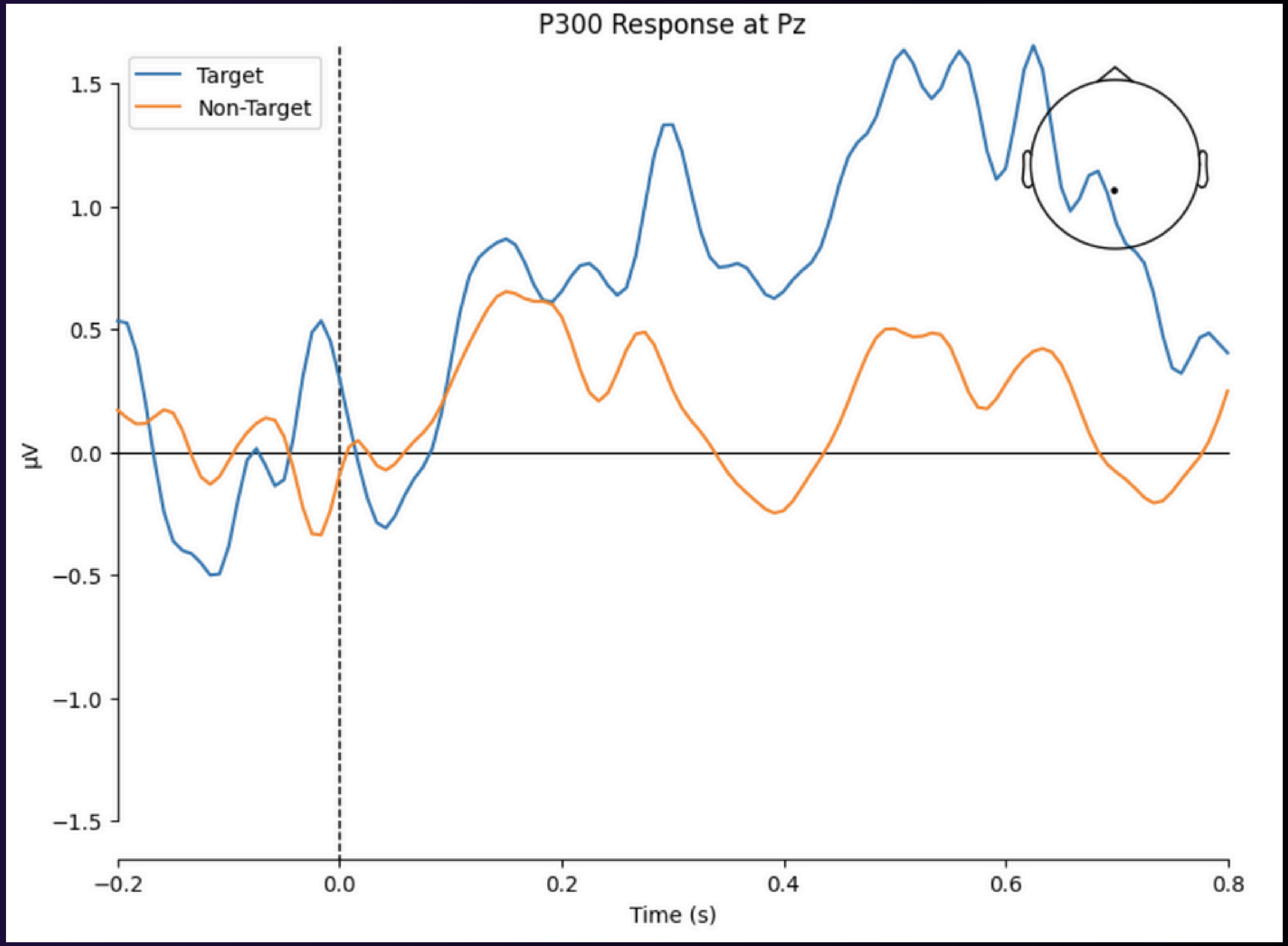
- Visualisation of data before and after filtering and resampling (Assignment 2)



RESULTS AND OBSERVATIONS

- P300 responses at Cz and Pz (Assignment 2)



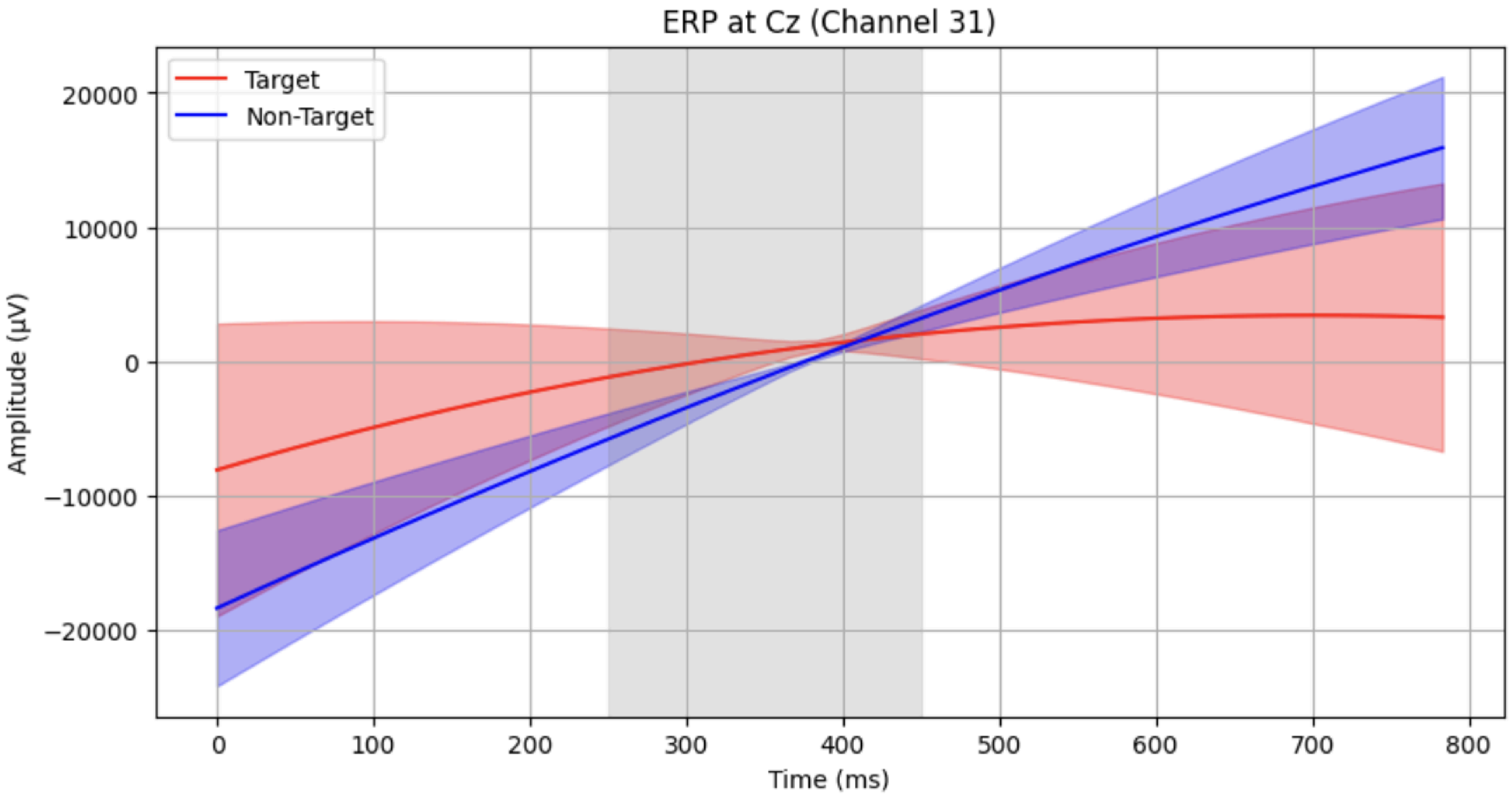


RESULTS AND OBSERVATIONS

- Visualisation of ERP (Assignment 3)

```
======================================================================
STEP 4: VISUALIZING ERP RESPONSES
======================================================================

--- Subject A ---
```




```
P300 amplitude difference: 1800.21 µV
```

RESULTS AND OBSERVATIONS

- Comparison of Feature Extraction Methods (Assignment 3)

```
======================================================================
STEP 5: FEATURE EXTRACTION
======================================================================

--- Subject A: Feature Comparison ---
  Target epochs for CSP: 637
  Non-target epochs for CSP: 3188

======================================================================
FEATURE COMPARISON (Balanced Classifiers)
======================================================================
PCA (20 comp):      Accuracy=0.7176, F1=0.2117
PCA (50 comp):      Accuracy=0.7072, F1=0.1318
CSP (6 comp):       Accuracy=0.5830, F1=0.3198
Time-Domain (3072): Accuracy=0.6209, F1=0.2204

Best feature method (Subject A): CSP

Subject A splits: Training=3060, Validation=765
Test features: (4500, 6)

--- Subject B: Feature Extraction ---

Using CSP (same as Subject A)...
  Target epochs for CSP: 646
  Non-target epochs for CSP: 3179
Subject B splits: Training=3060, Validation=765, Test features: (4500, 6)

Stored feature configuration:
{'method': 'csp', 'n_components': 6, 'transformer': <__main__.CSPTransformer object at 0x7fadae9ef560>}
```

RESULTS AND OBSERVATIONS

- Performance Evaluation of Baseline Classifiers (Assignment 3)

```
======================================================================
STEP 6: BASELINE CLASSIFIERS (Subject A)
======================================================================

===== LDA Evaluation =====
Accuracy: 0.5830065359477125
F1 Score: 0.31982942430703626
Precision: 0.21929824561403508
Recall: 0.5905511811023622

Confusion Matrix:
[[371 267]
 [ 52  75]]

Classification Report:
              precision    recall  f1-score   support

         0.0       0.88      0.58      0.70       638
         1.0       0.22      0.59      0.32       127

    accuracy                           0.58       765
   macro avg       0.55      0.59      0.51       765
weighted avg       0.77      0.58      0.64       765


===== Logistic Regression Evaluation =====
Accuracy: 0.5673202614379085
F1 Score: 0.3231083844580777
Precision: 0.21823204419889503
Recall: 0.6220472440944882

Confusion Matrix:
[[355 283]
 [ 48  79]]

Classification Report:
              precision    recall  f1-score   support

         0.0       0.88      0.56      0.68       638
         1.0       0.22      0.62      0.32       127

    accuracy                           0.57       765
   macro avg       0.55      0.59      0.50       765
weighted avg       0.77      0.57      0.62       765
```

RESULTS AND OBSERVATIONS

- Performance Comparison of Classical Machine Learning Classifiers (Assignment 3)

```
======================================================================
STEP 7: CLASSICAL MACHINE LEARNING (Subject A)
======================================================================

  Training Random Forest (n_estimators=100)...

  Training Gradient Boosting (n_estimators=100)...

Evaluating LDA...

Evaluating Logistic Regression...

Evaluating SVM...

Evaluating Random Forest...

Evaluating Gradient Boosting...

===== Model Comparison === Development Mode Activated!!!
LDA                  | Acc: 0.583 | F1: 0.320 | Prec: 0.219 | Recall: 0.591
Logistic Regression  | Acc: 0.567 | F1: 0.323 | Prec: 0.218 | Recall: 0.622
SVM                  | Acc: 0.519 | F1: 0.267 | Prec: 0.179 | Recall: 0.528
Random Forest        | Acc: 0.830 | F1: 0.000 | Prec: 0.000 | Recall: 0.000
Gradient Boosting    | Acc: 0.593 | F1: 0.254 | Prec: 0.183 | Recall: 0.417
```