

實驗三：腦機介面應用 - 分類器程式說明

1. 專案概述

本專案提供一個基於 Python 的機器學習程式，用於分析 EEG 訊號，並將其分類為「放鬆 (Relax)」或「專注 (Concentration)」兩種狀態。

你將使用一個 **多層感知器 (MLP)** 模型，並透過 **留一受試者交叉驗證 (Leave-One-Subject-Out, LOSO)** 的方式來評估效能。

👉 你的任務是：

- **不能更換模型類型**（必須使用 `MLPClassifier`）
- 可以 **調整 MLP 的超參數** (hidden layer 結構、學習率、正則化、batch size...等)
- 設計合適的 **資料前處理與後處理流程**，以提升分類準確率

基線準確率 (raw data, 無處理) 約 **55%**，實驗目標為 **≥ 60%**。

2. 環境設定

請確保執行環境已安裝以下軟體與套件：

1. **Python**: 建議使用 **Python 3.8 或以上版本**。
2. **必要套件安裝**: 在終端機輸入以下指令：

```
pip install numpy matplotlib seaborn scikit-learn scipy
```

3. 檔案結構

請確保專案資料夾結構如下，程式與資料需放在同一目錄下：

```
your_project_folder/
├── BCI_raw.py           # 主要 Python 程式 (已提供)
├── bci_dataset_113-2/   # 放所有受試者資料的資料夾
│   ├── S01/
│   │   ├── 1.txt       # 放鬆狀態 (Relax)
│   │   └── 2.txt       # 專注狀態 (Focus)
│   ├── S02/
│   │   ├── 1.txt
│   │   └── 2.txt
│   └── ... (S03, S04, ... , S18)
└── bci_dataset_114-1_gX/ # 放你這組當受試者資料的資料夾 (for part 2)
    ├── 學號(小寫)/
    │   ├── 1.txt       # 放鬆狀態 (Relax)
    │   └── 2.txt       # 專注狀態 (Focus)
    ├── 學號(小寫)/
    │   ├── 1.txt
    │   └── 2.txt
    └── 學號(小寫)/
        ├── 1.txt
        └── 2.txt
```

- 每個 `.txt` 檔為單欄 EEG 時域數值訊號，每行一筆取樣點。
- 取樣率：**500 Hz**。
- 每位受試者各有兩份檔案 (`1.txt = Relax` , `2.txt = Focus`)。

4. 執行程式

1. 開啟終端機。
2. 使用 `cd` 指令切換到專案資料夾：

```
cd your_project_folder
```

3. 執行程式：

```
python BCI_raw.py
```

4. 程式會載入資料並進行 **LOSO 交叉驗證**，過程可能需數分鐘。完成後，你將看到：
 - **終端機結果摘要**
 - **圖檔輸出** `bci_results_raw_data.png`

5. 可修改與不可修改的 HPs

類別	參數	是否可調整	建議範圍與限制
模型結構	HIDDEN_LAYER_SIZES	✓	可自由設定，如 (128, 64, 32)、(256, 128)
	activation	✗	固定為 relu
	solver	✗	固定為 adam
訓練超參數	LEARNING_RATE_INIT	✓	0.005 ~ 0.02
	ALPHA (L2 正則化)	✓	0.0001 ~ 0.05
	BATCH_SIZE	✓	32 ~ 128
	MAX_ITER	✓	10 ~ 60
資料切片	SEGMENT_LENGTH	✓	2 ~ 6 秒
	OVERLAP_RATIO	✓	0.0 ~ 0.8
	FEATURE_SELECTION	✓	True / False
	N_FEATURES_SELECT	✓	10 ~ 200 (需 \leq segment 長度)
其他	early_stopping	✗	固定不開放

6. 可修改的程式區塊

1. 前處理：例如濾波 (1–40 Hz)、z-score 標準化、band power 特徵。
2. 後處理：例如多數決投票、使用 predict_proba 調整閾值。

⚠ 請在程式碼中標明修改區域，例如：

```
# === student preprocessing ===  
# === student postprocessing ===
```

7. 輸出結果解讀

a. 終端機輸出 (範例)

需要把跳出的圖關掉才會跑喔

```
=====
```

```
Overall Mean Accuracy: 0.552 ± 0.091
```

```
Relax Class:
```

- Accuracy (Recall): 0.561 (2234/3983)
- Precision: 0.544 (2234/4100)

```
Concentration Class:
```

- Accuracy (Recall): 0.543 (2201/4050)
- Precision: 0.561 (2201/3920)

```
Results saved to 'bci_results_raw_data.png'
```

b. 圖片輸出

1. 每位受試者的準確率長條圖
2. 整體混淆矩陣 (2x2)
3. 訓練損失曲線

8. 實驗目標

需兩種類別之準確率同時達到以下目標才算成功。

- **Baseline (無前處理)**：約 55%
- **目標**：≥ 60%
- **加分**：≥ 65%

9. 評分標準


- **規範遵守 (20%)**：模型未被非法修改
- **準確率 (40%)**：Part 1 準確率達到 ≥60% 拿 25%，≥65% 拿 35%
- **報告分析 (40%)**：詳細說明方法與比較結果

10. 繳交報告


```
114_1_gX_exp3.zip/
├─ 114_1_gX_exp3.pdf    # 報告
├─ 114_1_gX_exp3_1/      # 放 part 1 程式
│   └─ 114_1_gX_exp3_1.py
│   └─ ...
├─ 114_1_gX_exp3_2/      # 放 part 2 程式
│   └─ 114_1_gX_exp3_2.py
│   └─ ...
└─ bci_dataset_114-1_gX/ # 放你這組當受試者資料的資料夾
    └─ 學號(小寫)/
        ├── 1.txt    # 放鬆狀態 (Relax)
        └─ 2.txt    # 專注狀態 (Focus)
    └─ 學號(小寫)/
        ├── 1.txt
        └─ 2.txt
    └─ 學號(小寫)/
        ├── 1.txt
        └─ 2.txt
```

11. FAQ

Q: 可以換 CNN 或 LSTM 嗎？

A:  不行，只能用 MLP。

Q: 可以改輸入長度嗎？

A:  可以，僅能透過 `SEGMENT_LENGTH`。

Q: 為什麼要用 LOSO？

A: 因為 EEG 訊號有強烈個體差異，LOSO 更能測試跨受試者泛化能力。