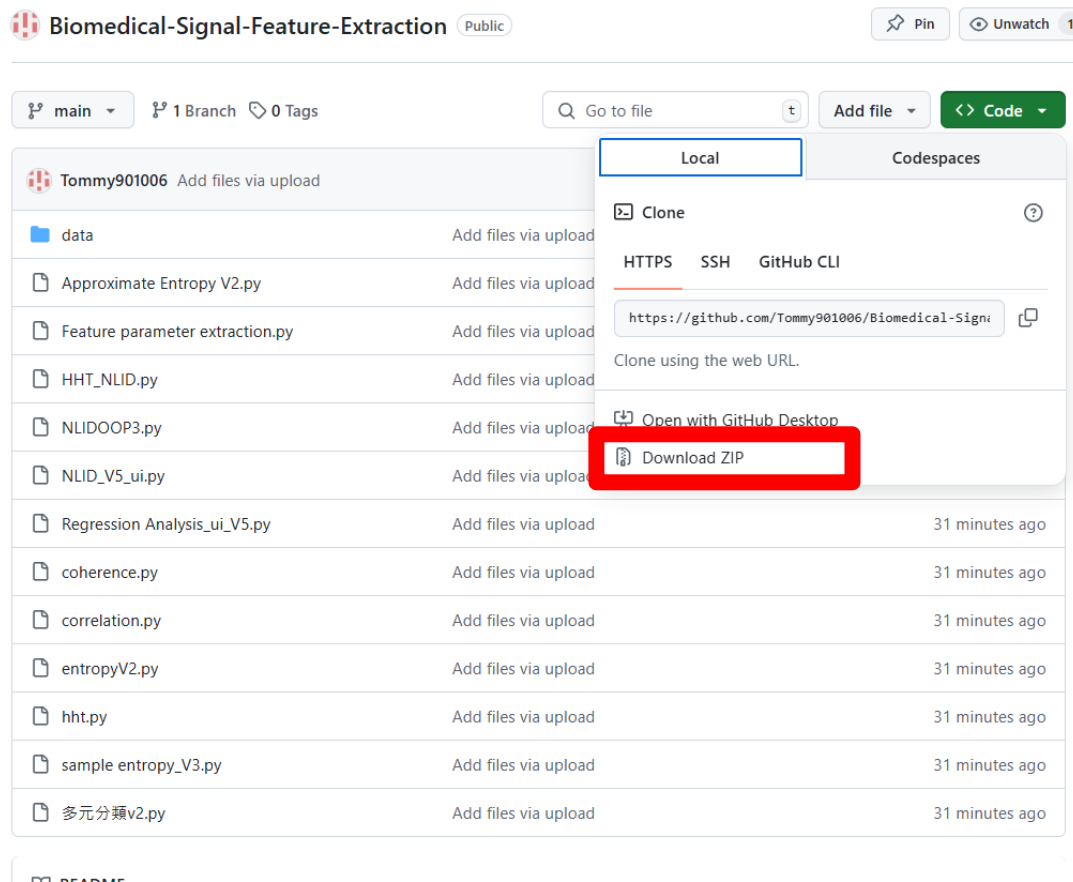


Github : <https://github.com/Tommy901006/Biomedical-Signal-Feature-Extraction.git>



1. 按下 code 後再按 Download zip
2. 下載 anaconda，來加速環境建置

<https://ithelp.ithome.com.tw/articles/10229662>

path 記得勾選

3. 安裝相關套件

打開 CMD



打上

```
pip install EntropyHub
```

```
pip install seaborn
```

```
pip install smote
```

```
pip install pyhht
```

```
pip install xgboost
```

```
pip install lightgbm
```

```
pip install scipy=1.10.1
```

```
pip install nolds
```

```
pip install openpyxl
```

4. 程式使用教學

4.1. Approximate Entropy V2

```
# 設定資料夾路徑
folder_path = r'C:\Users\User\Desktop\T\eye\內差法後excel'
```

改成你的資料夾路徑(例如：E:\Biomedical-Signal-Feature-Extraction-main\data)

```
# 確保數據是一維數組
x_data = data['X'].values.flatten()
y_data = data['Y'].values.flatten()
```

請換為你的檔案欄位（參考下圖，我的檔案欄位是X跟Y）

	A	B	C
1	Time	X	Y
2	0	1,731	892
3	1	1,730.39	885.96
4	2	1,729.79	879.93
5	3	1,729.18	873.89
6	4	1,728.57	867.86
7	5	1,727.96	861.82
8	6	1,727.36	855.79

m（維度）可以自己條

```
# 定義自訂的 Approximate Entropy 函數
def ApEn(Datalist, r=0.2, m=2):
    th = r * np.std(Datalist)
    return EH.ApEn(Datalist, m, r=th)[0][-1]
```

4.2. Coherence

```
# 設定資料夾路徑，請根據實際情況修改
folder_path = r"C:\Users\User\Desktop\eyetracker sample data\採樣結果\採樣後post"
```

改成你的資料夾路徑(例如：E:\Biomedical-Signal-

Feature-Extraction-main\data)

```
def parse_excel(file_path):  
    # 檢查是否含有 'X' 與 'Y' 欄位  
    if 'X' in df.columns and 'Y' in df.columns:  
        coh = calculate_coherence(df['X'], df['Y'])
```

請換為你的檔案欄位（參考下圖，我的檔案欄位是 X 跟 Y）

	A	B	C
	Y	Y	Y
1	Time	X	Y
2	0	1,731	892
3	1	1,730.39	885.96
4	2	1,729.79	879.93
5	3	1,729.18	873.89
6	4	1,728.57	867.86
7	5	1,727.96	861.82
8	6	1,727.36	855.79

須改紅色字的地方

```
if 'X' in df.columns and 'Y' in df.columns:  
    coh = calculate_coherence(df['X'], df['Y'])  
    results.append({'檔名': file, 'coherence': coh})
```

4.3. Correlation

```
# 設定資料夾路徑，請自行更新  
folder_path = r"C:\Users\User\Desktop\eyetracker sample data\採樣結果\採樣後post"
```

改成你的資料夾路徑（例如：E:\Biomedical-Signal-

Feature-Extraction-main\data)

```
# 檢查是否包含 'X' 與 'Y' 欄位  
if 'X' in df.columns and 'Y' in df.columns:  
    correlation = Pearson_correlation(df['X'], df['Y'])  
    results.append({'檔名': file, 'correlation': correlation})  
else:
```

請換為你的檔案欄位（參考下圖，我的檔案欄位是 X 跟

Y)

	A	B	C
	Y	Y	Y
1	Time	X	Y
2	0	1,731	892
3	1	1,730.39	885.96
4	2	1,729.79	879.93
5	3	1,729.18	873.89
6	4	1,728.57	867.86
7	5	1,727.96	861.82
8	6	1,727.36	855.79

須改紅色字的地方

```
# 檢查是否包含 'X' 與 'Y' 欄位
    if 'X' in df.columns and 'Y' in df.columns:
        correlation = Pearson_correlation(df['X'], df['Y'])
        results.append({'檔名': file, 'correlation':
correlation})
    else:
        print(f"檔案 {file} 中不含有 'X' 與 'Y' 欄位")
except Exception as e:
    print(f"處理檔案 {file} 時發生錯誤: {e}")
```

4.4. entropyV2

```
# 設置你的数据文件夹路径
data_folder = r'E:\Biomedical-Signal-Feature-Extraction-main\data'
output_file = r'E:\Biomedical-Signal-Feature-Extraction-main\data\post結果.xlsx'
```

Data_folder 改成你的資料夾路徑(例如：E:\Biomedical-Signal-Feature-Extraction-main\data)

Output_file 改成你要存放檔案的位置(例如：

E:\Biomedical-Signal-Feature-Extraction-main\data\post 結果.xlsx)一定要路徑+?. Xlsx

```
df['X'] = df['X'].round().astype(int)
df['Y'] = df['Y'].round().astype(int)

entropy_value_X = calculate_entropy(df['X'])
entropy_value_Y = calculate_entropy(df['Y'])

prob_X = calculate_probabilities(df['X'])
prob_Y = calculate_probabilities(df['Y'])
```

請換為你的檔案欄位（參考下圖，我的檔案欄位是 X 跟 Y）

	A	B	C
1	Time	X	Y
2	0	1,731	892
3	1	1,730.39	885.96
4	2	1,729.79	879.93
5	3	1,729.18	873.89
6	4	1,728.57	867.86
7	5	1,727.96	861.82
8	6	1,727.36	855.79

須改紅色字的地方

```
df['X'] = df['X'].round().astype(int)
df['Y'] = df['Y'].round().astype(int)

entropy_value_X = calculate_entropy(df['X'])
entropy_value_Y = calculate_entropy(df['Y'])

prob_X = calculate_probabilities(df['X'])
prob_Y = calculate_probabilities(df['Y'])
```

4.5. Feature parameter extraction

```
# 設定資料夾路徑
folder_path = r"C:\Users\User\Desktop\post\1"
```

改成你的資料夾路徑（例如：E:\Biomedical-Signal-

Feature-Extraction-main\data)

```
# 計算 X 欄位的統計量
x_stats = compute_statistics(df, 'X')

# 計算 Y 欄位的統計量
y_stats = compute_statistics(df, 'Y')
```

請換為你的檔案欄位（參考下圖，我的檔案欄位是 X 跟 Y）

	A	B	C
	Y	Y	Y
1	Time	X	Y
2	0	1,731	892
3	1	1,730.39	885.96
4	2	1,729.79	879.93
5	3	1,729.18	873.89
6	4	1,728.57	867.86
7	5	1,727.96	861.82
8	6	1,727.36	855.79

須改紅色字的地方

```
# 計算 X 欄位的統計量
x_stats = compute_statistics(df, 'X')

# 計算 Y 欄位的統計量
y_stats = compute_statistics(df, 'Y')
```

4. 6. HHT_NLID

```
# 設定資料夾路徑
folder_path = r"F:\ts\T\t" # 請替換成你的資料夾路徑
```

改成你的資料夾路徑(例如：E:\Biomedical-Signal-Feature-Extraction-main\data)

```

# 確保 X 和 Y 欄位存在
if "X" not in df.columns or "Y" not in df.columns:
    print(f"檔案 {file_name} 內缺少 'X' 或 'Y' 欄位，跳過此檔案。")
    continue

# 取得 X 和 Y 信號
signal_x = df["X"].values
signal_y = df["Y"].values

```

請換為你的檔案欄位（參考下圖，我的檔案欄位是 X 跟 Y）

	A	B	C
1	Time	X	Y
2	0	1,731	892
3	1	1,730.39	885.96
4	2	1,729.79	879.93
5	3	1,729.18	873.89
6	4	1,728.57	867.86
7	5	1,727.96	861.82
8	6	1,727.36	855.79

須改紅色字的地方

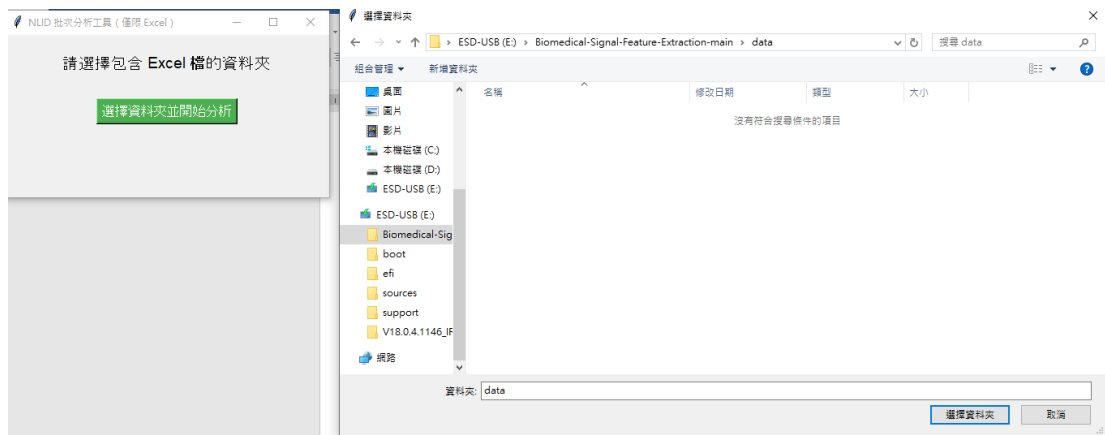
```

# 確保 X 和 Y 欄位存在
if "X" not in df.columns or "Y" not in df.columns:
    print(f"檔案 {file_name} 內缺少 'X' 或 'Y' 欄位，跳過此檔案。")
    continue

# 取得 X 和 Y 信號
signal_x = df["X"].values
signal_y = df["Y"].values

```

4.7. NLID_V5_ui



選好後會自動計算

```
if not {'X', 'Y'}.issubset(df.columns):
    print(f"跳過檔案 (缺少 X 或 Y 欄位): {filename}")
    continue

x = df['X'].dropna().values
y = df['Y'].dropna().values
```

請換為你的檔案欄位（參考下圖，我的檔案欄位是 X 跟 Y）

	A	B	C
1	Time	X	Y
2	0	1,731	892
3	1	1,730.39	885.96
4	2	1,729.79	879.93
5	3	1,729.18	873.89
6	4	1,728.57	867.86
7	5	1,727.96	861.82
8	6	1,727.36	855.79

須改紅色字的地方

```
if not {'X', 'Y'}.issubset(df.columns):
    print(f"跳過檔案 (缺少 X 或 Y 欄位): {filename}")
    continue
```

```
x = df['X'].dropna().values  
y = df['Y'].dropna().values
```

4.8. sample_entropy_V3

```
# 設定參數  
# 設置資料夾路徑  
folder_path = r'C:\Users\User\Desktop\Teye\內差法後excel' # 替換為你的資料夾路徑  
output_file = r'C:\Users\User\Desktop\Teye\Entropy\post-entropy_results.xlsx' # 結果輸出文件  
m = 1 # 嵌入維度
```

folder_path 改成你的資料夾路徑(例如：E:\Biomedical-Signal-Feature-Extraction-main\data)

Output_file 改成你要存放檔案的位置(例如：

E:\Biomedical-Signal-Feature-Extraction-main\data\post 結果.xlsx)一定要路徑+?. Xlsx

M(維度)可以根據分析檔案給定

4.9. Regression Analysis_ui_V5

Regression Model Evaluation

CSV File: Select

Select Input Columns (Multiple): Confirm

Select Target Columns: Confirm

Train Model

給 CSV 檔案格式如下

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Fixation_c	Fixation_c	Observatio	Y-Sample	IX-Sample	IX_SD	X_CV	Y_SD	Y_CV	D	NLID	correlation	R50	Observatio	SSQ-TS	SSQNLabe	SSQOLabe	SSQDLabe	SSQTSLab	mySSQTS	SSQN	SSQO	SSQD
987	129	27.1	0.1	0.13	47.66	0.03	128.3	0.12	192.8065	0.769272	0.011935	56.45678	3.57	490	1	2	1	1	1	28.6	60.6	41.8
1360	12	19.8	0.07	0.03	193.23	0.1	224.84	0.22	361.8846	0.65	0.0114	122.3899	0.168	56.7	1	1	1	1	1	0	15.2	C
173	19	11.4	0.01	0.01	582.06	0.28	226.49	0.3	508.3675	0.343853	0.062047	455.5298	1.22	449.7	1	2	1	1	2	19.1	45.5	55.7
118	4	1.98	0.12	0.06	212.03	0.12	147.96	0.16	340.6	0.353842	0.062047	223.6485	0.059	165.5	1	1	1	1	2	0	30.3	13.9
31	3	2.1	0.01	0.03	107.21	0.06	339.85	0.37	298.5716	0.616497	-0.05907	161.9047	0.172	345.6	1	2	1	1	1	19.1	45.5	27.8
87	1	4.24	0.06	0.06	101	0.05	266.85	0.2	337.7961	0.755593	-0.18725	160.088	0.027	281.5	1	2	1	1	2	9.5	37.9	27.8
142	16	5.01	0.12	0.08	139.45	0.07	300.6	0.19	384.5055	0.686501	-0.04311	212.6016	0.458	35.7	1	1	1	0	2	9.5	0	C
46	5	2.9	0.02	0.01	256.22	0.13	428.6	0.26	503.0965	0.425462	-0.06552	420.551	0.635	189.2	1	1	1	1	2	0	22.7	27.8
187	13	3.18	0.12	0.13	198.35	0.09	258.29	0.2	363.6102	0.506756	-0.07113	194.4496	0.186	771.3	1	2	2	1	2	19.1	75.8	111.4
416	94	9.11	0.08	0.16	103.73	0.06	571.64	0.64	389.9646	0.520255	0.011935	434.2476	2.1	293.5	1	2	1	1	2	19.1	45.5	13.9
1712	378	28.8	0.13	0.11	69.7	0.04	100.6	0.07	131.7963	0.89438	0.069126	62.75018	6.25	373.7	1	1	2	1	1	0	30.3	69.6
903	91	18.2	0.14	0.19	71.73	0.04	146.92	0.15	272.6888	0.697066	-0.36574	95.53474	2.07	201.1	1	1	1	1	1	9.5	30.3	13.9
1788	1074	27.3	0.14	0.06	126.24	0.07	168.2	0.13	55.64967	0.68	0.336179	95.52742	16	345.6	1	2	1	1	1	19.1	45.5	27.8
112	3	3.86	0.1	0.09	88.24	0.05	198.64	0.24	470.3622	0.785959	-0.12436	127.6645	0.067	390.3	1	2	1	1	2	9.5	53.1	41.8
1755	142	26.8	0.1	0.2	63.36	0.04	148.09	0.14	172.0178	0.980967	0.04	56.88091	2.1	120.7	1	1	1	1	1	9.5	22.7	C

4.10. 多元分類 v2

選擇CSV檔案:

選擇文件

選擇輸入參數:

Fixation_count_25
Fixation_count_10%
Observation_length_25%
Y-Sample Entropy
X-Sample EntropyEntropy
X_SD
X_CV
Y_SD
Y_CV
D

確認輸入參數

選擇目標參數:

確認目標參數

選擇分類器:

開始執行

給 CSV 檔案格式如下

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Fixation_count_25	Fixation_count_10%	Observation_length_25%	Y-Sample Entropy	X-Sample Entropy	X_SD	X_CV	Y_SD	Y_CV	D	NLID	correlation	R50	Observation_length_25%	SSQ-TS	SSQNLabe	SSQOLabe	SSQDLabe	SSQTSLab	mySSQTS	SSQN	SSQO	SSQD
987	129	27.1	0.1	0.13	47.66	0.03	128.3	0.12	192.8065	0.769272	0.011935	56.45678	3.57	490	1	2	1	1	1	28.6	60.6	41.8
1360	12	19.8	0.07	0.03	193.23	0.1	224.84	0.22	361.8846	0.65	0.0114	122.3899	0.168	56.7	1	1	1	1	1	0	15.2	C
173	19	11.4	0.01	0.01	582.06	0.28	226.49	0.3	508.3675	0.343853	0.062047	455.5298	1.22	449.7	1	2	1	1	2	19.1	45.5	55.7
118	4	1.98	0.12	0.06	212.03	0.12	147.96	0.16	340.6	0.353842	0.062047	223.6485	0.059	165.5	1	1	1	1	2	0	30.3	13.9
31	3	2.1	0.01	0.03	107.21	0.06	339.85	0.37	298.5716	0.616497	-0.05907	161.9047	0.172	345.6	1	2	1	1	1	19.1	45.5	27.8
87	1	4.24	0.06	0.06	101	0.05	266.85	0.2	337.7961	0.755593	-0.18725	160.088	0.027	281.5	1	2	1	1	2	9.5	37.9	27.8
142	16	5.01	0.12	0.08	139.45	0.07	300.6	0.19	384.5055	0.686501	-0.04311	212.6016	0.458	35.7	1	1	1	0	2	9.5	0	C
46	5	2.9	0.02	0.01	256.22	0.13	428.6	0.26	503.0965	0.425462	-0.06552	420.551	0.635	189.2	1	1	1	1	2	0	22.7	27.8
187	13	3.18	0.12	0.13	198.35	0.09	258.29	0.2	363.6102	0.506756	-0.07113	194.4496	0.186	771.3	1	2	2	1	2	19.1	75.8	111.4
416	94	9.11	0.08	0.16	103.73	0.06	571.64	0.64	389.9646	0.520255	0.011935	434.2476	2.1	293.5	1	2	1	1	2	19.1	45.5	13.9
1712	378	28.8	0.13	0.11	69.7	0.04	100.6	0.07	131.7963	0.89438	0.069126	62.75018	6.25	373.7	1	1	2	1	1	0	30.3	69.6
903	91	18.2	0.14	0.19	71.73	0.04	146.92	0.15	272.6888	0.697066	-0.36574	95.53474	2.07	201.1	1	1	1	1	1	9.5	30.3	13.9
1788	1074	27.3	0.14	0.06	126.24	0.07	168.2	0.13	55.64967	0.68	0.336179	95.52742	16	345.6	1	2	1	1	1	19.1	45.5	27.8
112	3	3.86	0.1	0.09	88.24	0.05	198.64	0.24	470.3622	0.785959	-0.12436	127.6645	0.067	390.3	1	2	1	1	2	9.5	53.1	41.8
1755	142	26.8	0.1	0.2	63.36	0.04	148.09	0.14	172.0178	0.980967	0.04	56.88091	2.1	120.7	1	1	1	1	1	9.5	22.7	C

4.11. Hht

```
# 讀取 Excel 檔案
file_path = "T\pre-p3.xlsx" # 請替換成你的 Excel 檔案名稱
df = pd.read_excel(file_path)
```

給定一個檔案相對路近(如 data\post-pl.xlsx)

```

# 確保 X 和 Y 欄位存在
if "X" not in df.columns or "Y" not in df.columns:
    raise ValueError("Excel 檔案內必須包含 'X' 和 'Y' 欄位!")

# 取得 X 和 Y 信號
signal_x = df["X"].values
signal_y = df["Y"].values

```

請換為你的檔案欄位（參考下圖，我的檔案欄位是 X 跟 Y）

	A	B	C
1	Time	X	Y
2	0	1,731	892
3	1	1,730.39	885.96
4	2	1,729.79	879.93
5	3	1,729.18	873.89
6	4	1,728.57	867.86
7	5	1,727.96	861.82
8	6	1,727.36	855.79

須改紅色字的地方

```

# 確保 X 和 Y 欄位存在
if "X" not in df.columns or "Y" not in df.columns:
    raise ValueError("Excel 檔案內必須包含 'X' 和 'Y' 欄位!")

# 取得 X 和 Y 信號
signal_x = df["X"].values
signal_y = df["Y"].values
t = np.arange(len(signal_x)) # 假設時間軸是索引序列

```