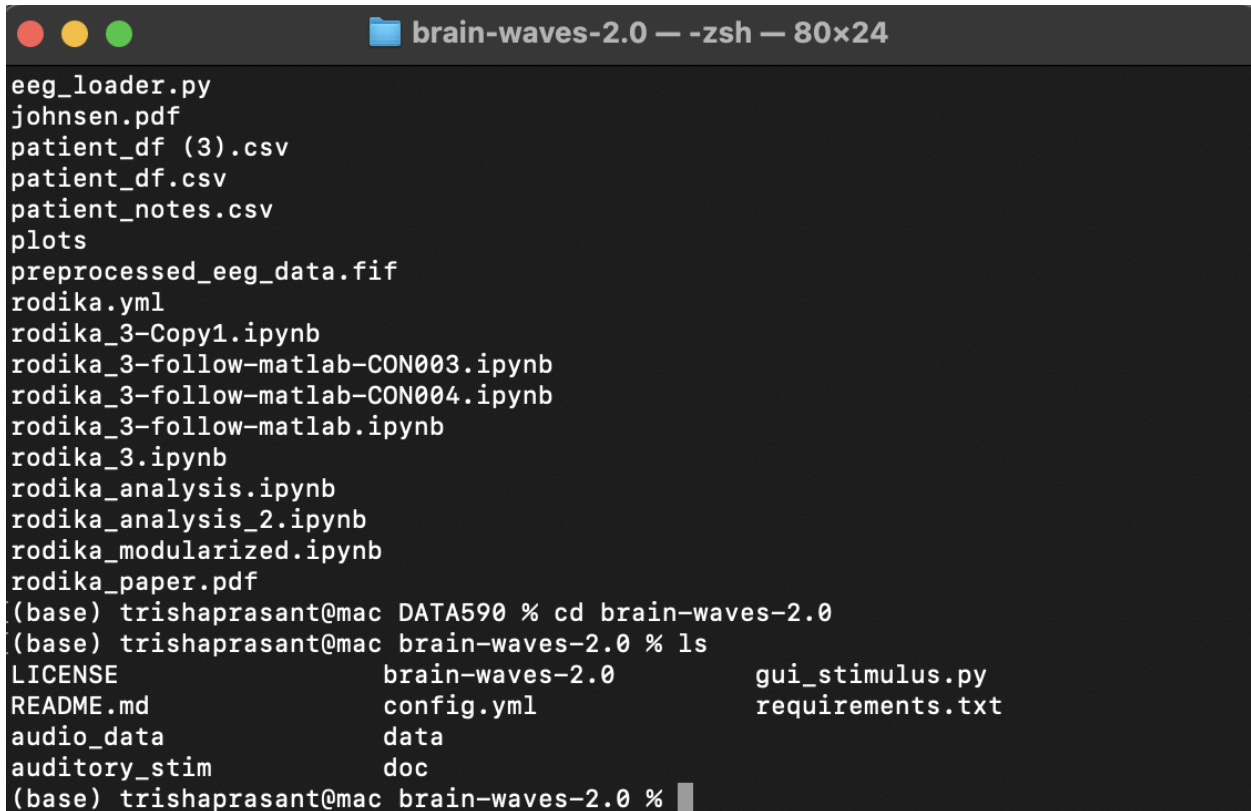


Brain Waves *Application* Step-by-Step Guide

Below is a **step-by-step** guide on how to use the Streamlit application. These instructions follow general workflow of our application,, so someone can replicate the process closely.

1. Launching the Application

1. **Open a terminal** (or command prompt).
2. **Navigate to your project folder** containing `gui_stimulus.py`.

A terminal window titled "brain-waves-2.0 — -zsh — 80x24" showing a list of files in the current directory, including eeg_loader.py, johnsen.pdf, patient_df (3).csv, patient_df.csv, patient_notes.csv, plots, preprocessed_eeg_data.fif, rodika.yml, rodika_3-Copy1.ipynb, rodika_3-follow-matlab-CON003.ipynb, rodika_3-follow-matlab-CON004.ipynb, rodika_3-follow-matlab.ipynb, rodika_3.ipynb, rodika_analysis.ipynb, rodika_analysis_2.ipynb, rodika_modularized.ipynb, and rodika_paper.pdf. The user then runs 'cd brain-waves-2.0' and 'ls', showing the directory contents: LICENSE, README.md, audio_data, auditory_stim, brain-waves-2.0, config.yml, data, doc, gui_stimulus.py, and requirements.txt.

```
eeg_loader.py
johnsen.pdf
patient_df (3).csv
patient_df.csv
patient_notes.csv
plots
preprocessed_eeg_data.fif
rodika.yml
rodika_3-Copy1.ipynb
rodika_3-follow-matlab-CON003.ipynb
rodika_3-follow-matlab-CON004.ipynb
rodika_3-follow-matlab.ipynb
rodika_3.ipynb
rodika_analysis.ipynb
rodika_analysis_2.ipynb
rodika_modularized.ipynb
rodika_paper.pdf
(base) trishaprasant@mac DATA590 % cd brain-waves-2.0
(base) trishaprasant@mac brain-waves-2.0 % ls
LICENSE                brain-waves-2.0        gui_stimulus.py
README.md              config.yml             requirements.txt
audio_data             data
auditory_stim         doc
(base) trishaprasant@mac brain-waves-2.0 %
```

Navigating to the location of brain-waves-2.0 on local computer

3. **Activate your Python/Conda environment** (e.g., `conda activate eeg`).

Run the Streamlit app by typing:

`streamlit run gui_stimulus.py`

```
brain-waves-2.0 — open ◀ streamlit run gui_stimulus.py — 80x24
rodika.yml
rodika_3-Copy1.ipynb
rodika_3-follow-matlab-CON003.ipynb
rodika_3-follow-matlab-CON004.ipynb
rodika_3-follow-matlab.ipynb
rodika_3.ipynb
rodika_analysis.ipynb
rodika_analysis_2.ipynb
rodika_modularized.ipynb
rodika_paper.pdf
(base) trishaprasant@mac DATA590 % cd brain-waves-2.0
(base) trishaprasant@mac brain-waves-2.0 % ls
LICENSE          brain-waves-2.0      gui_stimulus.py
README.md        config.yml           requirements.txt
audio_data       data
auditory_stim    doc
(base) trishaprasant@mac brain-waves-2.0 % streamlit run gui_stimulus.py

You can now view your Streamlit app in your browser.

Local URL: http://localhost:8501
Network URL: http://192.168.86.37:8501
```

Running streamlit app using `streamlit run gui_stimulus.py`

4. **Open your web browser** to the URL displayed in your terminal (often <http://localhost:8501>).

2. Administer Stimuli (Tab 1)

A. Patient ID & Trial Types

1. **Look for the field** labeled **"Enter Patient/EEG ID"**.
 - **For example:** typing in the patient ID "CON004", ensures that any stimuli you administer will be recorded under this ID.
2. **Click** the **"Prepare Stimulus"** button.
 - This often randomizes or fetches the trial types from your back-end code. You'll see a message or have trials assigned in memory if successful.

Administer Stimuli Patient Information Results

EEG Stimulus Package

Administer Auditory Stimuli

Enter Patient/EEG ID

CON004

Prepare Stimulus

Done

Play Stimulus

First tab of application with patient ID entered, and stimulus prepared

B. Playing the Stimulus

1. **Check** that your patient ID is still set.
2. **Click "Play Stimulus"**.
 - The system shows a progress bar, updating as each trial is administered.
 - In the background, the software logs the start and end times of each stimulus in a CSV (the `patient_df.csv` or a similar file).
3. **Watch** for any console output.
 - The terminal shows messages/ status updates in the Streamlit interface confirming how many stimuli were played.
 - This will take time!

```
brain-waves-2.0 — ffplay ◀ streamlit run gui_stimulus.py — 80x24
lang_48', 'lang_11', 'lang_69', 'lang_37', 'lang_21', 'lang_0', 'lang_39', 'lang_50', 'lang_40', 'beep', 'lang_52', 'lang_46', 'lang_36', 'lang_7', 'lang_38', 'lang_62', 'lang_9', 'beep', 'lang_27', 'lang_47', 'lang_63', 'lang_19', 'lang_22', 'beep', 'lang_64', 'lcmd', 'lang_53', 'lang_26', 'lang_4', 'rcmd', 'lang_23', 'lang_29', 'lang_51', 'beep', 'lang_67', 'lang_55', 'rcmd', 'lang_35', 'lang_61', 'lang_32', 'lang_56', 'lang_45', 'lang_49', 'beep', 'lang_28', 'lang_30', 'lang_59', 'lang_12', 'lang_65', 'lang_43', 'lang_60', 'lang_42', 'lang_15', 'lang_3', 'rcmd']
Trial 0: lang_57
Playing audio_data/lang_trials/lang_57.mp3...
Input #0, wav, from '/var/folders/kp/42tvxzv924xdtzydmjjp380000gn/T/tmpffgjfpvi.wav':
  Duration: 00:00:15.38, bitrate: 352 kb/s
  Stream #0:0: Audio: pcm_s16le ([1][0][0][0] / 0x0001), 22050 Hz, 1 channels, s16, 352 kb/s
  15.26 M-A: 0.000 fd= 0 aq= 0KB vq= 0KB sq= 0B
Trial 1: lang_25
Playing audio_data/lang_trials/lang_25.mp3...
Input #0, wav, from '/var/folders/kp/42tvxzv924xdtzydmjjp380000gn/T/tmpup8kazvw.wav':
  Duration: 00:00:15.38, bitrate: 352 kb/s
  Stream #0:0: Audio: pcm_s16le ([1][0][0][0] / 0x0001), 22050 Hz, 1 channels, s16, 352 kb/s
  4.28 M-A: 0.000 fd= 0 aq= 104KB vq= 0KB sq= 0B
```

Terminal output when audio is being played

C. Adding Notes

1. **Scroll down to "Add Notes to your Selected Patient and Date."**
2. **Type** any relevant note in the text box (e.g., "Patient responded slowly after the second beep.").
3. **Click "Add Note".**

Add Notes to your Selected Patient and Date

Write your note here

Example Note!

Add Note

Your note was successfully added to patient_notes.csv

Find Patient Notes

The following notes have been written for the selected patient and date:

Example note added on Tab 1 of application

D. Reviewing Existing Notes

1. **Look** under "Find Patient Notes".
 2. **Select** a patient ID and date from the drop-down menus (if available).
 3. **Read** any existing notes that are displayed.
-

3. Upload EEG Files (Tab 2)

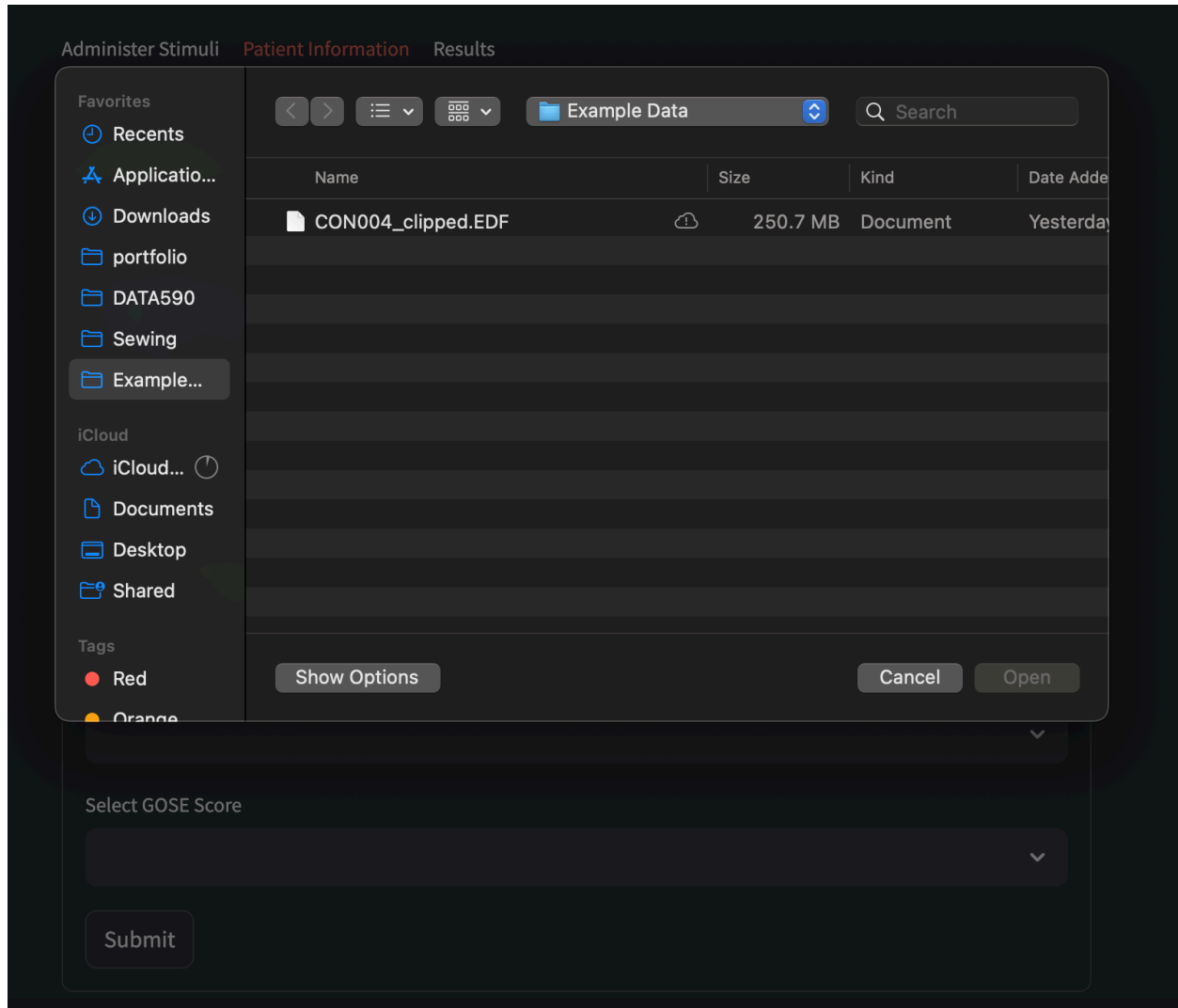
A. Patient ID, Date, & EDF Upload

1. **Type** the patient ID (e.g., "CON004") in the "Patient ID" field.

The screenshot shows a web application interface for uploading EEG files. At the top, there are three tabs: 'Administer Stimuli', 'Patient Information' (which is active and highlighted in red), and 'Results'. Below the tabs, the main heading is 'Upload EEG Files'. The form contains several input fields: 'Patient ID' with the value 'CON004', 'Recording Date' with the value '2025/03/18', 'Upload EDF File' which includes a cloud icon, the text 'Drag and drop file here', 'Limit 500MB per file • EDF', and a 'Browse files' button, 'Select CPC Score' with a dropdown arrow, and 'Select GOSE Score' with a dropdown arrow. At the bottom left of the form is a 'Submit' button.

Tab 2 of Application with patient ID and Recording Date inputted

2. **Select** a date from the calendar widget under "Recording Date."
3. **Upload** the EDF file by clicking on "Browse files" (or dragging the file) into the "Upload EDF File" widget.
 - Ensure your file is named in a way that the system can use it or that you keep track of the name for future reference.



Selected a new EEG reading (in the form of .edf file) to upload on to application

B. CPC / GOSE Scales

1. **Pick** the relevant CPC (Cerebral Performance Category) from the drop-down.
2. **Pick** a GOSE (Glasgow Outcome Scale Extended) level from the second drop-down.

Patient ID

CON004

Recording Date

2025/03/18

Upload EDF File

Drag and drop file here
Limit 500MB per file • EDF

Browse files

CON004_clipped.EDF 250.7MB

Select CPC Score

CPC 1: No neurological deficit

CPC 2: Mild to moderate dysfunction

CPC 3: Severe dysfunction

CPC 4: Coma

CPC 5: Brain death

Selecting CPC Scores for Patient CON004

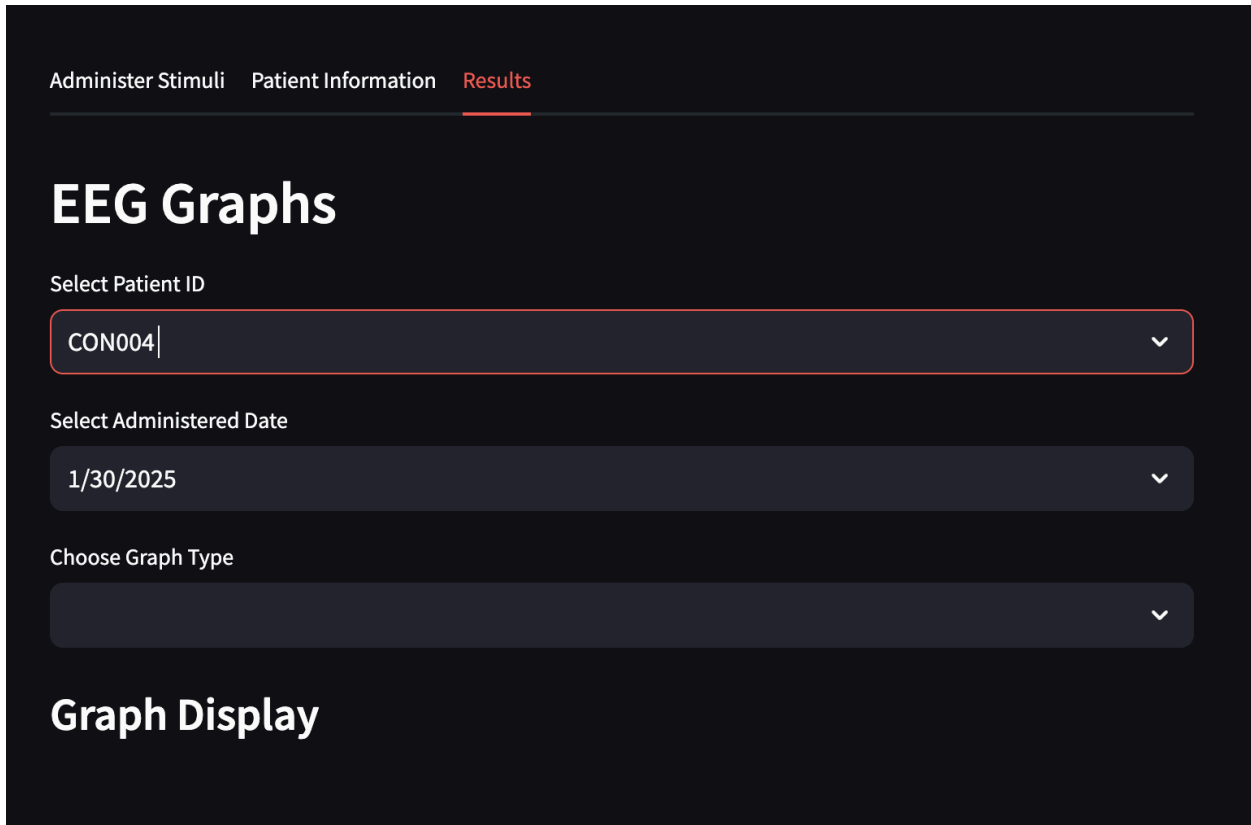
C. Submitting

1. Click "Submit".
 - The app saves your uploaded EDF file to the designated folder (`edf_dir`) and logs the CPC/GOSE labels in a CSV (like `patient_label_path`).

4. Viewing Results / Graphs (Tab 3)

A. Selecting Patient & Date

1. **Select** a patient from the drop-down labeled "**Choose Patient**".
2. **Pick** a relevant date from "**Recording Date**" (or another date selector).
 - The app constructs a filename (e.g., `PatientID_Date.edf`) based on your selection.



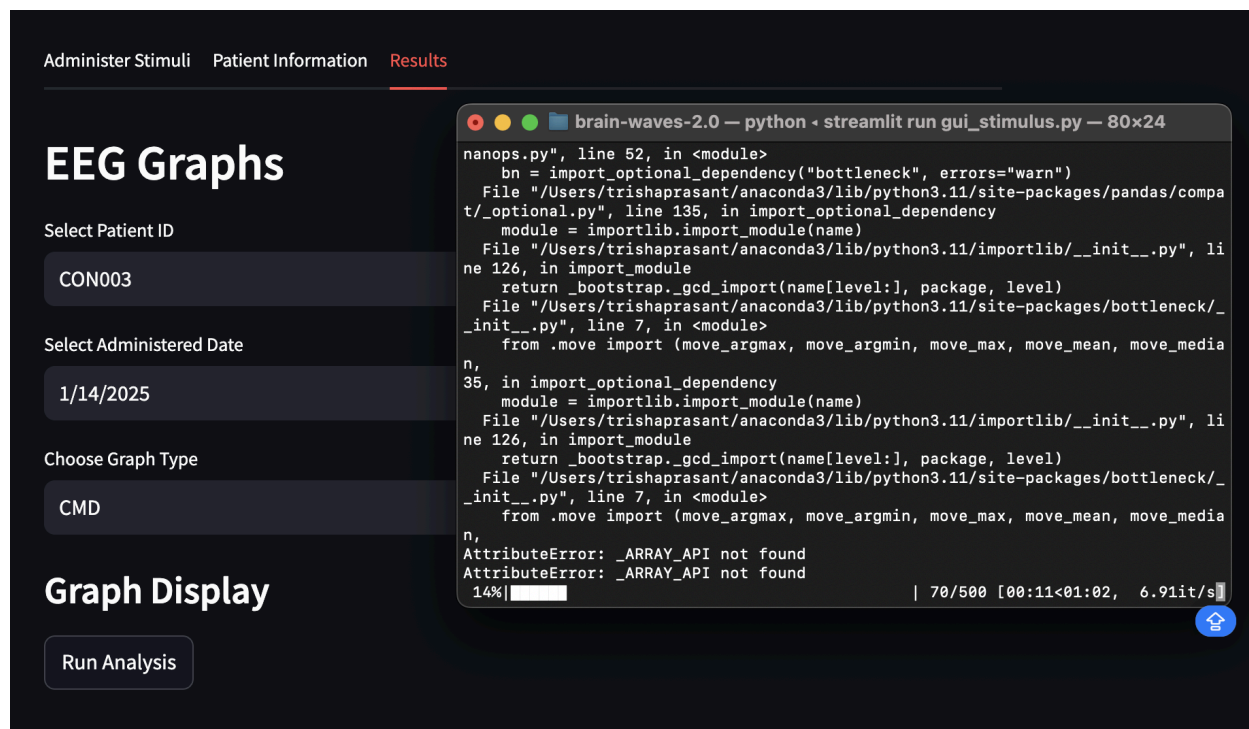
Tab 3 of Application with Patient ID Selected

B. Graph Type

1. **Choose** the **Graph Type** from the drop-down. Typically, you'll see options like "**CMD**," "**Language Tracking**," or "**EEG Reactivity**."
2. **If you pick "CMD"**, the code will load classification plots for motor command analysis.
3. If you pick "Language Tracking," you will see ITPC plots generated for all electrodes or a selected channel.

C. Running the Analysis (If Provided)

1. If there is a button (e.g., "**Run Analysis**" or "**Run Language Tracking Analysis**"), **click** it.
 - The system may call functions from your `eeg_auditory_stimulus` package (like `claassen_analysis.run_analysis(...)` or `rodika_modularized.main(...)`).



Side-by-side output of terminal when clicking “Run Analysis” for CMD Analysis

Select Patient ID

CON004

Select Administered Date

1/30/2025

Choose Graph Type

Language Tracking

Graph Display ⇄

Language Tracking Options

Select Bad Channels

T7 × Fp1 × Fp2 ×

Select EOG Channels

Fp1 × Fp2 × T7 ×

Select Graph to Display

Average ITPC

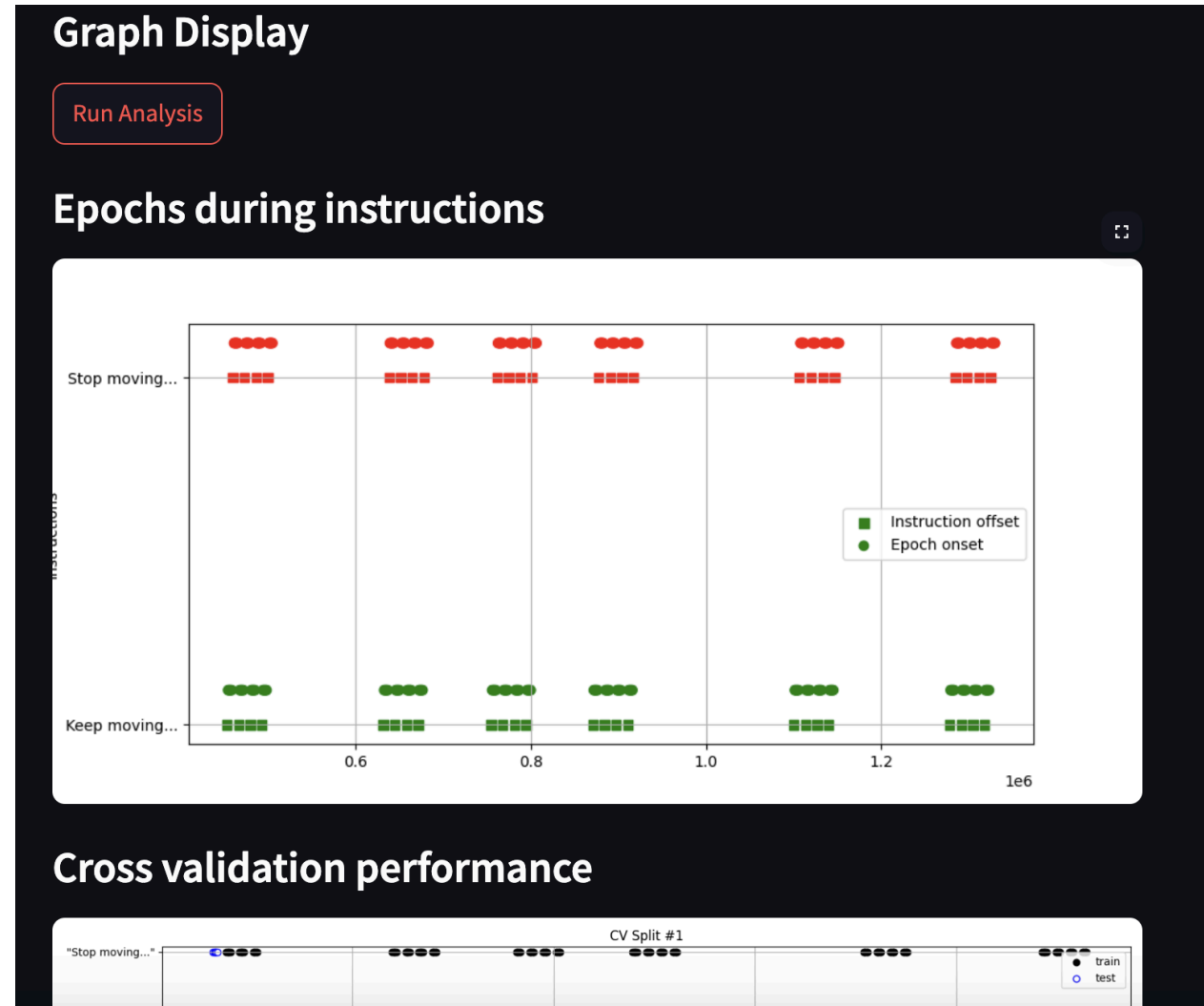
Run Language Tracking Analysis

Preparation for ITPC graph results, before clicking “Run Language Tracking Analysis”

2. **Wait** for the app to generate results or show a success message

D. Displaying Graphs

1. **Scroll down** to see any auto-loaded plots (SVM classification scores, or speech-entrainment ITPC at 3 Hz).



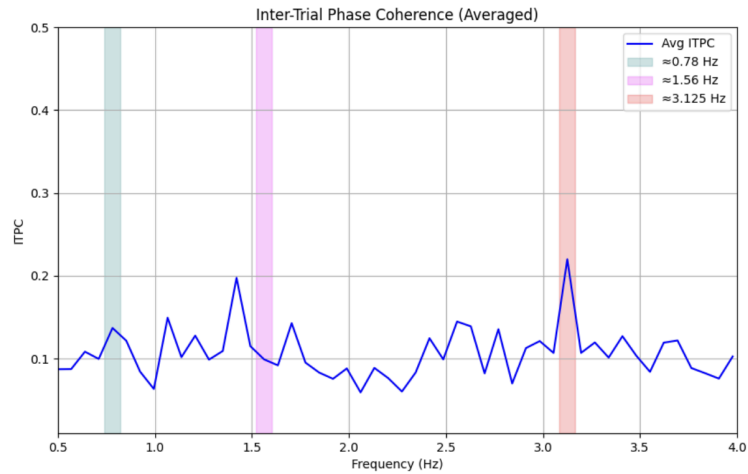
Output of CMD Analysis Charts (should be 6 charts in total)

Select Graph to Display

Average ITPC

Run Language Tracking Analysis

Graph Display



ITPC for CON004

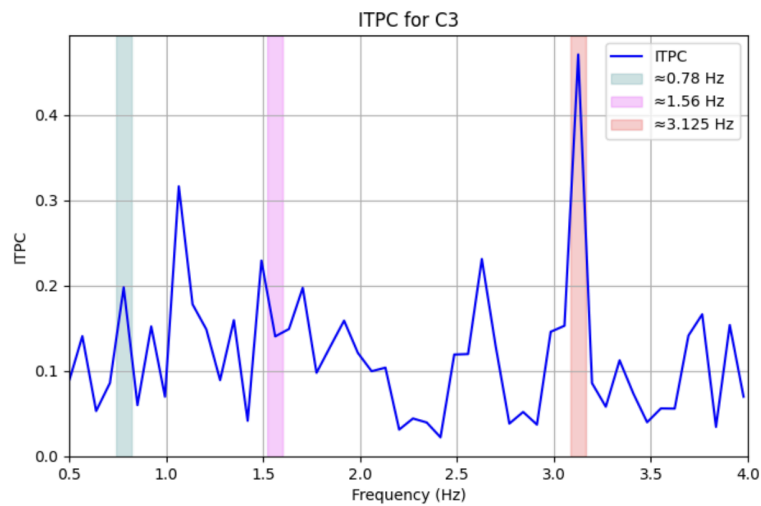
Output of ITPC graph display: averaged ITPC scores over all electrodes

Choose a Channel

C3

Run Language Tracking Analysis

Graph Display



ITPC for CON004

Output of ITPC graph display: ITPC score for one select electrode

5. Troubleshooting Tips

- **File Path Issues:** Make sure your EDF file or stimulus CSV path is correct in the config or user interface.
 - **No Patient or Date Shown:** If a drop-down is empty, it may mean no data for that ID/date combination is in your CSV or the patient never had a stimulus trial in Tab 1.
 - **Progress Bars & Logging:** Tab 1's "Play Stimulus" might show a progress bar. This typically means it's writing each stimulus trial into `patient_df.csv` for future reference.
-

6. Workflow Summary

1. **Tab 1:** Enter a patient ID → prepare trials → click "Play Stimulus" → optionally add or review notes.
 2. **Tab 2:** Upload your EEG data (EDF files) for that patient and date → specify outcome scales.
 3. **Tab 3:** Select the same patient and date → pick an analysis or graph → (optionally) run advanced analysis → review displayed plots or classification metrics.
-

7. Closing or Stopping the App

1. **Return to your terminal** where you started `streamlit run gui_stimulus.py`.
2. **Press Ctrl + C** (or **Cmd + C** on macOS) to stop the Streamlit server.

```
brain-waves-2.0 — -zsh — 80x24

from .move import (move_argmax, move_argmin, move_max, move_mean, move_med
n,
35, in import_optional_dependency
    module = importlib.import_module(name)
    File "/Users/trishaprasant/anaconda3/lib/python3.11/importlib/__init__.py", li
ne 126, in import_module
        return _bootstrap._gcd_import(name[level:], package, level)
    File "/Users/trishaprasant/anaconda3/lib/python3.11/site-packages/bottleneck/_
_init__.py", line 7, in <module>
        from .move import (move_argmax, move_argmin, move_max, move_mean, move_med
n,
AttributeError: _ARRAY_API not found
AttributeError: _ARRAY_API not found
100%|████████████████████████████████████████████████████████████████████████████████| 500/500 [01:20<00:00, 6.21it/s]
Permutation p-value = 0.9320
Empirical AUC = 0.42 +/- 0.19
Shuffle AUC = 0.50
p-value = 0.9321
Test run: False
Test run: False
Test run: False
Test run: False
^C Stopping...
(base) trishaprasant@mac brain-waves-2.0 %
```

Stopping streamlit app, by using Ctrl + C

3. **Deactivate** your Conda environment if you wish (`conda deactivate`).