1928
K. N. Toosi University
of Technology

Faculty of Electrical Engineering

Course: Functional Brain Imaging Systems

Professor: Dr. Ali Khadem

Final Project

Topic: *Temporal Dynamics of Action Prediction in a Social Context Using EEG and SVM*

Spring 2025

Prepared By:

Mohammadreza Shahsavari, Ramin Tavakoli, Mahdi Molaie

Project Overview

This project aims to identify the precise moment the brain decodes and predicts another person's actions during a social interaction. We will use a machine learning approach, specifically a Support Vector Machine (SVM) classifier, to analyze Electroencephalography (EEG) data.

The dataset is the same we used for Computer Assignment 3 (CA3). It is from an experiment involving an "attacker" and a "blocker." Our goal is to decode the blocker's brain activity to determine which action the attacker chose (selecting a left or right button).

To achieve this, we will implement a sliding window analysis. An SVM model will be trained and tested on small, sequential time windows of the blocker's EEG data. By systematically shifting this window across the trial period, we can track the model's performance over time.

We hypothesize that the SVM's classification accuracy will initially be at chance level and then rise significantly. This increase in accuracy should pinpoint the earliest time point at which the blocker's brain processes the information about the attacker's impending move.

Data

The dataset for this project is an expanded version of the one used in CA3, featuring data from 10 subjects instead of one.

The data is organized into folders for each subject. Within each subject's folder, you will find the following four files:

1. `*.eeg`
2. `*.vhdr`
3. `*.vmrk`
4. `TrialInfo.csv`

The main dataset directory also contains a crucial file, `EEG Event Markers.xlsx`, which provides definitions for all event IDs used in the data.
For a comprehensive background on the experimental paradigm and data collection methods, please refer back to the CA3 documentation.

You can download this data set from here. You send an access request before you can download the data.

1. Preprocess EEGs

To ensure reliable results, the raw EEG data for all 10 subjects must be cleaned. Apply the complete preprocessing pipeline you developed in CA3 to each subject's data.

This process should replicate the methodology used previously, including filtering and the individual inspection and interpolation of any bad channels. The objective is to create a clean and consistent dataset across all participants before proceeding to the analysis phase.

2. Creating the Accuracy-Over-Time Plot

The goal of this analysis is to track how well the SVM classifier can predict the attacker's action at different moments in time. To do this, you will use a **time-resolved decoding** approach (also known as a "sliding window" analysis).

The process involves three main stages:

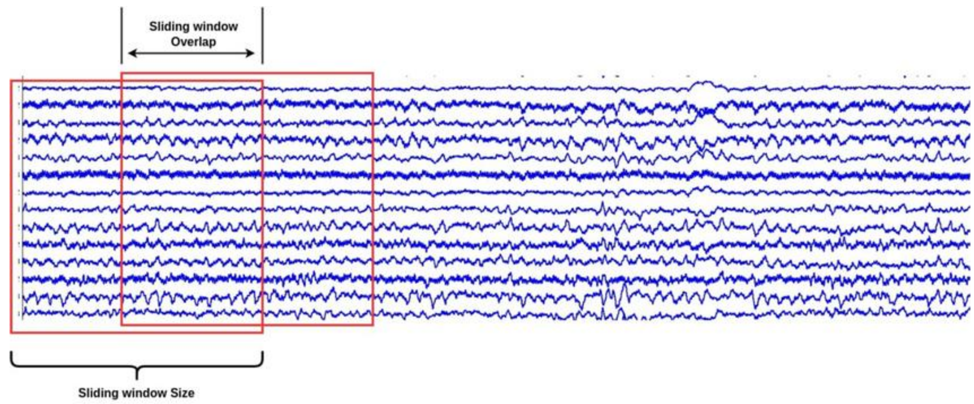
1. Running the core sliding window analysis on a single subject.
2. Repeating and averaging the results to create a robust, stable curve.
3. Visualizing the final group data with statistical information.

Part 1: The Sliding Window Loop (for a Single Subject)

First, let's define the core procedure for one subject's data.

1. **Prepare Data and Variables:**
    - Select a single subject's preprocessed data.
    - Randomly split the trials into a **70% training set** and a **30% testing set**.
    - Initialize an empty list to store your results, for example: prediction_accuracies = [].
2. **Define Window Parameters:**
    - **Window Length:** The duration of the data segment to analyze at each step. A good starting point is **50 ms**.

- **Stride:** The amount to shift the window forward in each iteration. A smaller stride gives higher temporal resolution but is more computationally expensive. A good starting point is **10 ms**.



3. Iterate Through Time:

Create a loop that moves the analysis window from the beginning of the epoch to the end (e.g., from -500 ms to +2000 ms relative to the stimulus). For each step in the loop:

- **Extract Data:** For the current time position, extract the corresponding window of EEG data (e.g., a 50 ms segment) from **every trial** in both your training and testing sets.
- **Train the Model:** Train an **SVM classifier** using the data from the *training set*. The features (X) will be the EEG data from the extracted windows, and the labels (y) will be the trial types (i.e., the attacker's choice of 'left' or 'right').
- **Test and Save Accuracy:** Use the trained model to predict the labels for the *testing set* windows. Calculate the prediction accuracy for this specific time window and append it to your prediction_accuracies list.

4. End the Loop:

Continue this process, shifting the window by the defined stride (10 ms) each time, until you have covered the entire time range of the trial. At the end, your prediction_accuracies list will contain the SVM's performance across time for a single train/test split of one subject.

Part 2: Achieving a Stable Result Through Averaging

A single accuracy curve is too noisy to be reliable. We will average the results both within and across subjects to get a clear and robust picture.

1. Within-Subject Averaging:

To reduce noise caused by a single random train/test split, you must repeat Part 1 multiple times (e.g., 10 times) for the same subject. Each time, use a new, different random 70/30 split of the trials. This will result in 10 separate prediction_accuracies lists for that one subject. Average these 10 lists together point-by-point to create a single, stable accuracy-over-time curve for that subject.

2. Across-Subject (Group) Averaging:

Now, perform the entire "Within-Subject Averaging" procedure (Step 1 above) for all 10 subjects. This will leave you with one final, averaged accuracy curve for each of the 10 subjects. To get the final group result, calculate the grand average by averaging these 10 curves together.

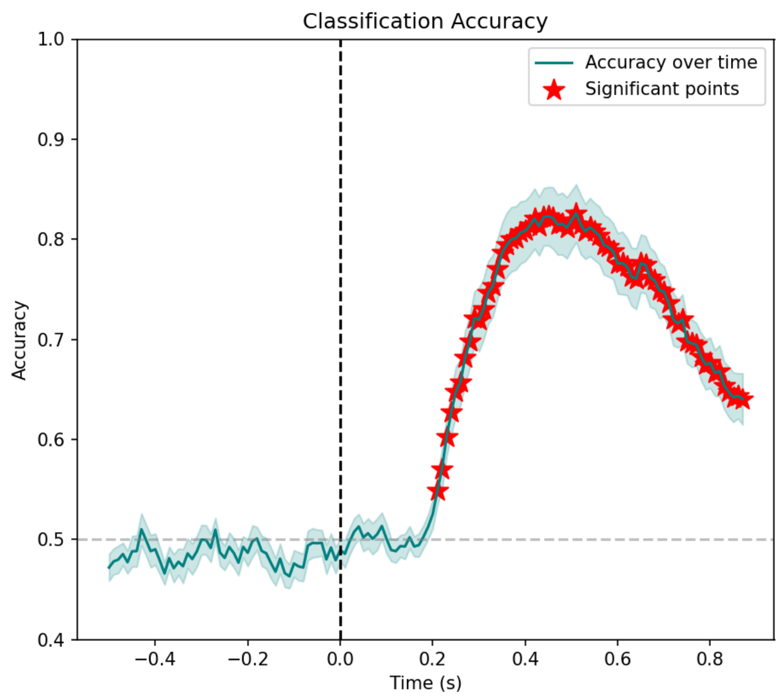
Part 3: Visualization and Statistical Analysis

The final step is to plot the results and identify when the decoding is statistically meaningful.

1. **Plot the Grand Average:** Plot the final grand average accuracy curve, with **Time (ms)** on the x-axis and **Classification Accuracy (%)** on the y-axis.
2. **Add Error Bars:** To visualize the variability across your participants, add an **error bar** to your plot. This is typically the **standard error of the mean (SEM)** calculated at each time point across the 10 subjects' averaged accuracy curves.

3. **Identify Significant Time Points:** We expect accuracy to start around the chance level (**50%** for a two-choice task). The plot should reveal the point where the accuracy curve rises and stays significantly above chance. You can determine this statistically by finding the time points where the lower bound of the error bar is consistently above the 50% chance level. This indicates that the decoding performance is reliable across the group and not just due to random fluctuation.

The ultimate output of this stage is the following figure:



3. Identifying Informative Brain Regions

In the previous section, our analysis of all EEG channels showed that predictive information about the attacker's action is present in the blocker's brain and when it emerges. Now, we will investigate where this information is located.

To answer this, we'll repeat the time-resolved decoding analysis on specific subsets of electrodes that correspond to different brain regions. If the classification accuracy for a particular region rises significantly above chance, it suggests that this area is involved in processing the attacker's actions and contributes to the prediction. Conversely, regions where accuracy remains at chance level are likely less involved in this specific cognitive task.

Methodology: Region-Specific Analysis

1. **Define Electrode Clusters:** First, we group the electrodes into anatomically-based clusters. You will use the following predefined clusters for this analysis:

```
#all_electrodes = raw.Info.ch_names

electrode_clusters = {
    # All the electrodes.
    'all_electrodes': all_electrodes,

    # Frontal regions
    'left_prefrontal': ['Fp1', 'AF7', 'AF3'],
    'midline_prefrontal': ['AFz'],
    'right_prefrontal': ['Fp2', 'AF8', 'AF4'],

    'left_frontal': ['F7', 'F5', 'F3', 'F1'],
    'midline_frontal': ['Fz'],
    'right_frontal': ['F2', 'F4', 'F6', 'F8'],

    # Fronto-temporal regions
    'left_frontotemporal': ['FT9', 'FT7'],
```

```
    'right_frontotemporal': ['FT8', 'FT10'],

    # Fronto-central regions
    'left_frontocentral': ['FC5', 'FC3', 'FC1'],
    'right_frontocentral': ['FC2', 'FC4', 'FC6'],

    # Central regions
    'left_central': ['C5', 'C3', 'C1'],
    'midline_central': ['Cz'],
    'right_central': ['C2', 'C4', 'C6'],

    # Temporal regions
    'left_temporal': ['T7'],
    'right_temporal': ['T8'],

    # Centro-parietal regions
    'left_centroparietal': ['CP5', 'CP3', 'CP1'],
    'midline_centroparietal': ['CPz'],
    'right_centroparietal': ['CP2', 'CP4', 'CP6'],

    # Temporo-parietal regions
    'left_temporoparietal': ['TP7'],
    'right_temporoparietal': ['TP8'],

    # Parietal regions
    'left_parietal': ['P7', 'P5', 'P3', 'P1'],
    'midline_parietal': ['Pz'],
    'right_parietal': ['P2', 'P4', 'P6', 'P8'],

    # Parieto-occipital regions
    'left_parietooccipital': ['PO7', 'PO3'],
    'midline_parietooccipital': ['POz'],
    'right_parietooccipital': ['PO4', 'PO8'],

    # Occipital regions
    'left_occipital': ['O1'],
    'midline_occipital': ['Oz', 'Iz'],
    'right_occipital': ['O2']
}
```

2. **Iterative Decoding:** You will now loop through each brain region defined above. For each region:
   - Isolate the data from only the electrodes belonging to that specific cluster.
   - Perform the **entire time-resolved decoding analysis** exactly as you did in the previous section (including the within-subject and across-subject averaging).
   - The result of this process will be a final, averaged accuracy-over-time curve for each distinct brain region.

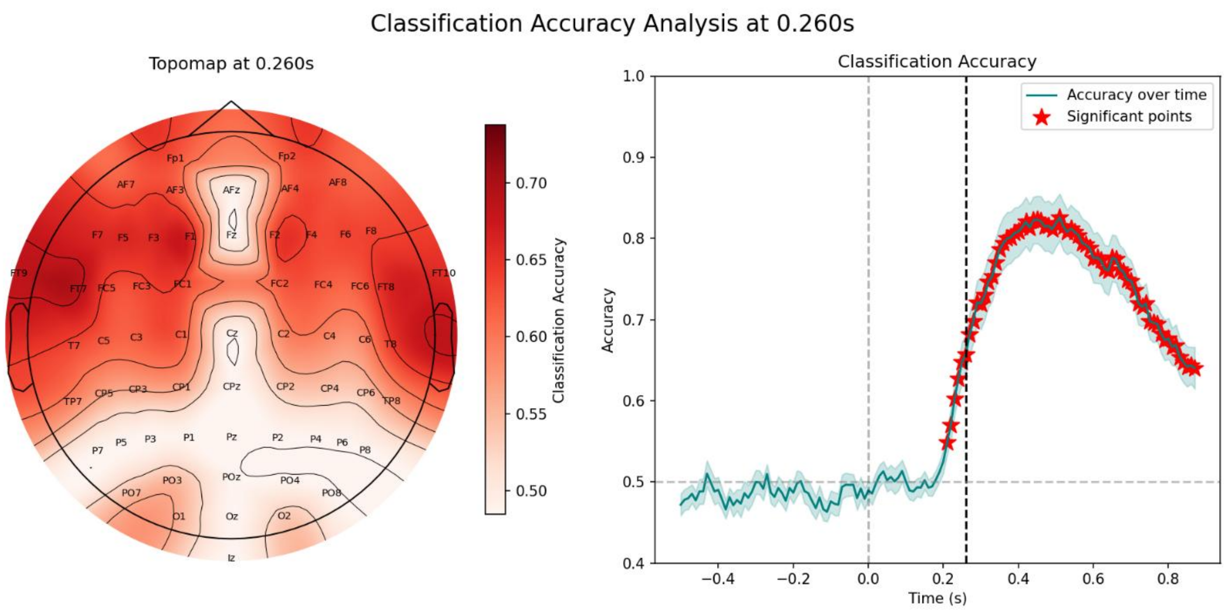
Visualization: Creating a Topographic Accuracy Video

After completing the analysis for every region, you will have a set of accuracy curves—one for each part of the brain. The next step is to visualize these results as a dynamic topographic map that shows how information is distributed across the scalp over time.

1. **Generate Topographic Maps (Frames):** You will create a video where each frame is a topographic map of decoding accuracy at a single point in time. To create a frame for a given time point `t`:
   - Take the accuracy value you calculated for a specific cluster (e.g., left_prefrontal) at time `t`.
   - Assign that **same accuracy value** to every electrode within that cluster (e.g., 'Fp1', 'AF7', and 'AF3' would all be assigned the same value).
   - Repeat this for all clusters until every electrode on the scalp has an assigned accuracy value for time `t`.

- Plot a single topographic map using these values.

2. **Create the Video:** Generate one such map for each time step in your analysis. By stitching these frames together in chronological order, you will create a video that illustrates how the predictive information evolves across the brain throughout the trial. An example frame from this video is shown below.



As in previous assignments, save each frame and then compile them into a video file. For a reference of the expected output, you can view a sample video here.

You can find a sample accuracy topomap video from here**.** The video indicates that the frontal and temporal regions are primarily responsible for processing information related to an attacker's movements.

4. Source Localization: Identifying Deeper Brain Regions

Our previous analyses on the scalp identified which sensor groups were most informative. However, scalp-level analysis has limited spatial resolution and primarily reflects activity from the cortical surface. To gain deeper insight and pinpoint the origins of this predictive activity with greater precision, we will use **source localization**.

This technique estimates the location of neural activity within the three-dimensional volume of the brain. By applying our decoding analysis to these estimated brain sources rather than to the scalp sensors, we can investigate the role of specific anatomical regions, including those deeper in the brain.

Your Task

1. **Research and Select a Method:** Conduct research to find an appropriate source localization method for this type of EEG analysis (e.g., MNE, dSPM, sLORETA, or Beamformers).
2. **Justify Your Choice:** In your report, clearly state which method you chose and provide a brief rationale explaining why it is suitable for this project.
3. **Implement and Analyze:** Apply the chosen method to estimate brain source activity. Then, repeat the complete time-resolved SVM decoding analysis (from section 2), using the activity of these brain sources (or predefined regions of interest in source space) as input for the classifier.

4. **Present and Interpret:** Present the results of your source-level decoding. Discuss which brain regions show significant decoding accuracy and what these findings imply about the neural networks involved in action prediction.

توجه:

1. حتماً در موعد مقرر به بارگذاری پاسخ تمرین در سامانه VC اقدام کنید (تاخیر قابل قبول نیست)
2. در صورت بروز هرگونه ابهام یا سوال در مورد تمرین حتما با دستیاران آموزشی در ارتباط باشید.
3. لطفاً نام خانوادگی خود را به صورت لاتین به همراه شماره تمرین به عنوان نام فایل بارگذاری شده قرار دهید.

به عنوان مثال :

FIS _FinalProject_YourFullName

4. پاسخ ارسالی شما باید در یک فایل ورد نوشته شده و فایل ورد و PDF آن هردو در یک فایل rar ارسال گردند.
5. زمان کافی جهت تحقیق جهت یافتن پاسخ تمارین به دانشجویان گرامی داده شده. لازم به ذکر است که هر شخص باید برداشت و نتایج تحقیقات خود را ارائه دهد پس به همین دلیل از کپیبرداری پاسخنامه یکدیگر شدیدا خودداری فرمایید و در صورت مشاهده مسئولیت عواقب منفی متوجه تمامی افراد خاطی میگردد. همچنین ممکن است جلساتی تحت عنوان پرسش و پاسخ در خصوص گزارش دوستان گرامی برگزار شود

موفق باشید.