

BE 521: Homework 8

p300 Speller

Spring 2015

34 points

Due: 4/09/2015 11:59 PM

Objective: Spell letters using neurosignals

Homework Policy

1. Piazza should be used for peer discussion for all questions related to course material. Please also use Piazza to contact teaching staff for all questions. TA's will be available to help during office hours and occasionally on Piazza.
2. Submit LaTeX write-up (pdf) and Matlab code to Canvas as pennkey_hwx.pdf, .m before listed deadline.
3. Assignments will be returned electronically on Canvas.
4. Collaboration is encouraged but individual write-ups are required. Please list any collaborators. Honor code will be strictly enforced. Note: submitted code is routinely passed through a plagiarism checker.
5. Late Policy: 5% per day. **No homework is accepted after the 5th late day.** (e.g. If originally due Tuesday, 11:59PM, last day to turn in is Sunday, 11:59 PM).

P300 Speller

In this homework, you will work with data from a P300-based brain computer interface called BCI2000 (Schalk et al. 2004) that allows people to spell words by focusing their attention on a particular letter displayed on the screen. In each trial the user focused on a letter, and when that letter's row or column is flashed, the user's brain elicits a P300 evoked response. By analyzing whether a P300 signal was produced, across the flashes of many different rows or columns, the computer can determine the letter that the person is focusing on.

Figure 1 shows the letter matrix from one trial of this task.

Data Organization

The data for this homework is stored in I521_A0008_D001 on the IEEG Portal. The EEG in this dataset were recorded during 85 intended letter spellings. For each letter spelling, 12 row/columns were flashed 15 times in random order ($12 \times 15 = 180$ iterations). The EEG was recorded with a sampling rate of 240 Hz on a 64-channel scalp EEG.



Figure 1: The letter matrix for the P300 speller with the third row illuminated. If the user were focusing on any of the letters in the third row (M, N, O, P, Q, or R), their brain would emit a P300 response. Otherwise it would not.

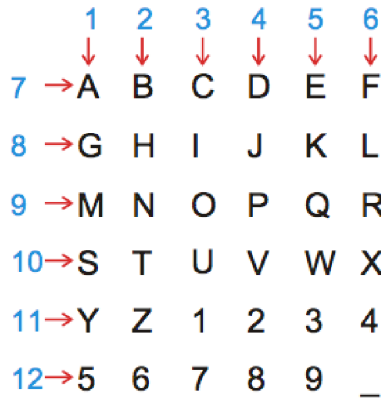


Figure 2: The row/column indices of the letter matrix.

Figure 2 shows the label of each row or column, as encoded in the `Stim` annotation layer (`annotation.description`) matrix.

The annotations for this dataset are organized in two layers as follows:

- **TargetLetter** annotation layer indicates the target letter (`annotation.description`) on which the user was focusing during the recorded EEG segment (`annotation.start/annotation.stop`)
- **Stim** annotation layer indicates the row/column that is being flashed (`annotation.description`) and whether the target letter is contained in that flash (`annotation.type`). The recorded EEG during that flash is (`annotation.start/annotation.stop`)

Hints: There are many annotations in this dataset and getting them all may take 5-10 minutes. Once you retrieve the annotations once, save them for faster loading in the future. Also, use `{ }` to gather variables across structs for easier manipulation (e.g. `strcmp({annotations.type}, '1')`)

Topographic EEG Maps

You can make topographic plots using the provided `topoplotEEG` function. This function needs an “electrode file.” and can be called like

```
topoplotEEG(data, 'eloc64.txt', 'gridscale', 150)
```

where `data` is the value to plot for each channel. This function plots the electrodes according to the map in Figure 3.

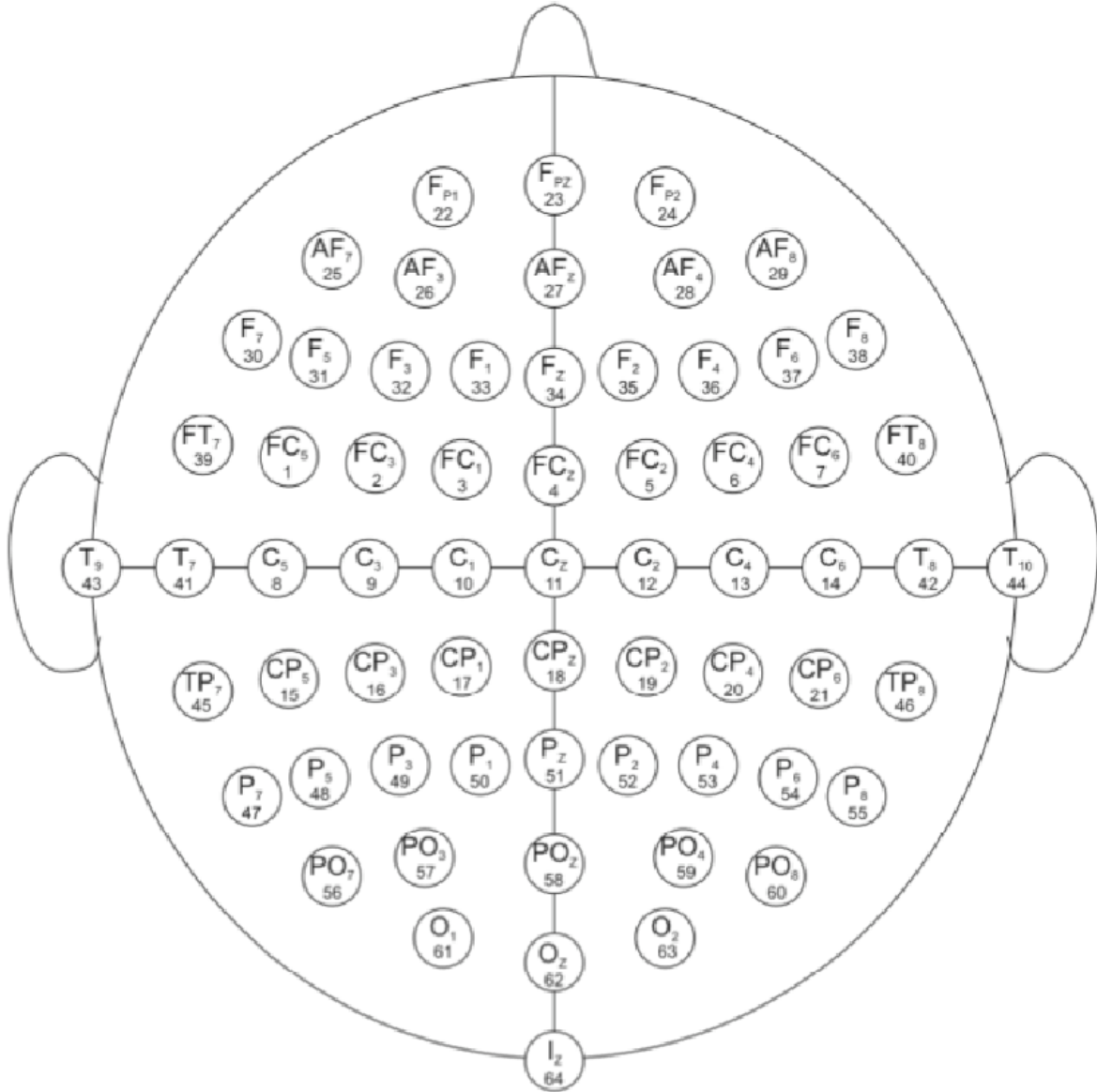


Figure 3: The scalp EEG 64-channel layout.

1 Exploring the data

In this section you will explore some basic properties of the data in I521_A0008_D001.

1. For channel 11 (Cz), plot the mean EEG for the target and non-target stimuli separately, (i.e. rows/columns including and not-including the desired character, respectively), on the same set of axes. Label your x-axis in milliseconds. (3 pts)
2. Repeat the previous questions for channel 23 (Fpz). (1 pts)
3. Which of the two previous channels looks best for distinguishing between target and non-target stimuli? Which time points look best? Explain in a few sentences. (2 pts)
4. Compute the mean difference between the target and non-target stimuli for each channel at timepoint 300 ms averaged across all row/column flashes. Visualize these values using the `topoplotEEG` function. Include a colorbar. (3 pts)
5. How do the red and blue parts of this plot correspond to the plots from above? (2 pts)

2 Using Individual P300s in Prediction

Hopefully the Question 1.4 convinced you that the Cz channel is a reasonably good channel to use in separating target from non-target stimuli in the P300. For the rest of the homework, you will work exclusively with this channel.

1. Explain a potential advantage to using just one channel other than the obvious speed of calculation advantage. Explain one disadvantage. (3 pts)
2. One simple way of identifying a P300 in a single trial (which we'll call the *p300 score*) is to take the mean EEG from 250 to 450 ms and then subtract from it the mean EEG from 600 to 700 ms. What is the *p300 score* for epoch 10, iteration 11 at electrode Cz? (3 pts)
3. Plot the *p300 scores* for each row/column in letter epoch 20 at electrode Cz. (3 pts)
4. Based on your previous answer for letter epoch 20, what letter do you predict the person saw? Is this prediction correct? (2 pts)
5. Using this *p300 score*, predict (and print out) the letter viewed at every epoch. What was your prediction accuracy? (2 pts)

3 Automating the Learning

In Section 2, you used a fairly manual method for predicting the letter. Here, you will have free rein to use put any and all learning techniques to try to improve your testing accuracy.

1. Play around with some ideas for improving/generalizing the prediction paradigm used in the letter prediction. Use the first 50 letter epochs as the training set and the later 35 for validation. Here, you are welcome to hard-code in whatever parameters you like/determine to be optimal. What is the optimal validation accuracy you get? (4 pts)

2. Describe your algorithm in detail. Also describe what you tried that didn't work. (6 pts)