LING646 FALL 2015
LAB 1 - ERP

In this lab, you will explore ERP data from a recent experiment examining the effect of predictability in adjective-noun sequences (Lau, Namyst, Fogel & Delgado, unpublished manuscript). In this experiment, we used a text corpus to derive estimates of the probability of a given noun following an adjective, and we used this instead of human-normed Cloze probability to create materials that varied in the predictability of the noun. All of the items were semantically congruous. In the current lab, you will focus on the simple contrast between high probability (>50%) and low probability (< 2%) items. The task participants performed was a memory recall test; at regular intervals throughout the session, participants were given a break and then asked to indicate whether they had seen particular adjective-noun sequences in the previous set of trials. Items were counterbalanced such that the same noun appeared in both conditions across lists but no words were repeated within a session for a given participant.

High Predictability	Low Predictability
runny nose	runny yogurt
curly hair	soaked hair

To do this lab, you will use two Matlab toolboxes, EEGLAB and ERPLAB. Therefore, you can do this lab on any computer that has Matlab installed and on which you are able to install these toolboxes. You can download EEGLAB here: http://sccn.ucsd.edu/eeglab/ and ERPLAB here: http://erpinfo.org/erplab.

Installation of these two packages should be easy. First you unzip the downloads. The EEGLAB directory can go anywhere, but you need to put the ERPLAB folder inside the EEGLAB 'plugins' subdirectory. Finally, open Matlab, go to File -> Set Path, 'Add with subfolders', select the EEGLAB folder, and save the path. You can test whether it worked by typing 'eeglab' at the Matlab command line. It should open up a window, and one of the menus within that sub-window should read 'ERPLAB'. That means everything is working.

The data for the lab is available here:
ling.umd.edu/~ellenlau/courses/ling646/Lab1_ERP_data.zip

This document should contain all the information you need to complete the lab. If you get stuck, you are welcome to contact me for help, but you can also try consulting the very nice ERPLAB tutorial located here: http://erpinfo.org/erplab/erplab-documentation/tutorial/. This lab should also provide you with the basic processing steps needed for the group project, although you can check out the tutorial for many more options in preprocessing and data display.

For most of this lab you'll examine EEG data from a single subject who participated in this paradigm. Start EEGLAB in Matlab by typing 'eeglab' at the command line. If you've correctly installed ERPLAB, you should see an *ERPLAB* tab in the EEGLAB menu.

I. DATA CONVERSION

The first step is to convert your raw data from the '.cnt' format that our Neuroscan acquisition software gives us to the '.set' format that EEGLAB uses. Go to File -> Import Data -> Neuroscan .CNT. Choose the MOOSE_S11.cnt file in the lab data directory. Important: check the '16-bit' box on the subsequent screen for data format, then say 'OK'. Name the file something informative like 'S11'.

Now we need to add information about where the channels are located. Go to Edit -> Channel locations and say 'OK' on the screen that pops up. This will load the location information. Now click OK to finish.

You now have a raw data file in EEGLAB format. At this point, it would be a good idea to save it as a '.set' file. You can do this by going to File -> Save current dataset as… and saving as 'S11.set'.

If your computer does not have enough memory, you may get an 'OUT OF MEMORY' error when you try the import step. If this happens, please contact your instructor.

II. CONTINUOUS EEG DATA

If the .set file is not still loaded from the last step, you can load the data with *File -> Load existing dataset*, and select *S1.set*. Now take a first look at the continuous data by selecting *Plot -> Channel data (scroll)*.

1. Scroll across the data using the arrow keys at the bottom of the screen. At what point in time do you first see something that looks like a blink? Illustrate with a screenshot.

2. One of the electrodes is a little noisier than the others, probably due to an imperfect connection. Which one is it?

3. How are the stimulus events indicated in the plot? (Hint: the recording started before some time before the stimulus events began, so you need to scroll a ways down to see these)

Next, we will create a structure that contains the information about all the events that ERPLAB will use, called an event list. Go to *ERPLAB -> EventList -> Create EEG EventList (Basic)*. Click on 'Export EventList to text file' and enter a filename (e.g. S1_eventlist.txt) before clicking 'CREATE'. When the next window pops up, say 'OK'. You now have two datasets loaded in EEGLAB, the original one and the one with the event list attached (S1_elist). You can toggle back and forth between them using the *Datasets* tab, but so far the basic parameters should be the same. Take a moment to look at the basic parameters listed under the 'S1_elist' dataset.

4. How many channels of data were recorded in this experiment?

5. What was the sampling rate for this dataset?

6. Open up the eventlist.txt file in a text editor. Notice that the first set of rows have event codes 224 and 225, but these never appear again in the experiment. Any guesses about what these might be?

III. BIN CREATION AND EPOCHING

The next step is to sort the event codes into different bins, or what we might think of as conditions. The idea here is that you may have sent a lot of different event codes during your experiment, not all of which are relevant for your analysis. You may also want to exclude certain sequences of trials from analysis. For example, if you send different codes for correct and incorrect responses, you can exclude trials that contained an incorrect response.

The file that tells ERPLAB which events you want to bin together is called a 'bin descriptor file' or bdf. The original bdf that we actually used for analyzing this data is included as 'MOOSE_bdf_full.txt' and you can look at it if you are curious about all the event codes in the experiment. However, for this lab we will focus on the comparison between high probability and low probability items, for which you can use the simpler bdf file 'MOOSE_bdf_646.txt'. Take a moment to open it up and take a look. A good practice before starting a new EEG experiment is to create your bdf file ahead of time. This can help you confirm that you have coded your materials such that you can actually extract all your events of interest.

Now with the S11_elist set selected in the EEGLAB window, select *ERPLAB -> Assign Bins (BINLISTER)*. Load the MOOSE_646_bdf.txt file as the Bin Descriptor File, specify that you will Write resulting EVENTLIST to text file called 'S11_eventlist_bins.txt', and click 'Run'. You should see the progress bar in the Matlab command window, and when it finishes you should see a record of 'Successful trials per bin:' at the bottom.

7. How many trials per condition are in the 2 critical conditions (HiProb and LoProb)?

You should have a new dataset called S11_elist_bins. Now we are finally going to extract the 'epochs'. This means we will grab a window of time around each of our events of interest, and concatenate all these windows into one file. This will give us a much smaller file that will make the next step of artifact rejection faster and more efficient. Go to *ERPLAB -> Extract bin-based epochs*. Pick the time-range from -200:800ms. Make sure that the Baseline Correction is set to 'Pre'; this is the step that will shift the data such that the activity prior to the stimulus is considered the 'zero' point. Now click 'RUN' and say 'OK' to create a new dataset, called S11_elist_bins_be. The info in the EEGLAB window for this dataset should be different from the previous ones.

8. How many frames per epoch are listed? Why do you think this is?

9. How many epochs were found?
Go to *Plot->Channel data (scroll)* again and take a look at the new dataset. As you can see, the data is now composed of concatenated events, each from -200:800ms.

10. Which is the first event to contain an artifact? Illustrate with a screenshot.

11. Which is the first event to contain a blink? Illustrate with a screenshot.

If you click on one of the events in this view, you can see it turns yellow. One way to exclude trials containing artifact would be to click on an event and press the 'REJECT' button. However, ERPLAB contains very nice routines for automatically excluding artifact. A good routine for getting rid of blinks is a step function, which is sensitive to very broad peaks like eyeblinks. We will need to specify which channel we expect to be sensitive to eyeblinks. For this lab we will pick the VEOG (vertical electro-oculogram), channel, which is channel 32. Go to *ERPLAB -> Artifact detection -> Step-like artifacts*. Now enter the following parameters and click 'Accept'. Say 'OK' to create a new dataset, *S11_elist_bins_be_ad*.

After the dataset has been created, a channel plot should pop up at the same time. Note that the epoch with an eyeblink has now been marked yellow, indicating that the algorithm correctly identified it as an artifact.

12. Scroll across the file. Does it appear that most of the eyeblinks have been identified? Did it also catch other kinds of artifacts? Do you think the default threshold of 40 microVolts is appropriate or too conservative?

Output a summary of the artifact rejection at the command window by going to *ERPLAB -> Artifact detection -> Summarize -> Table* and select *Show in Command Window*.

13. What percent of total trials were rejected? How many trials does this represent?

At this point you could also apply other thresholds, e.g. on HEOG to remove horizontal eye movement artifacts and across all channels to exclude generic kinds of high amplitude noise.

IV. COMPUTING AND PLOTTING ERPS

Finally, let's compute the averaged ERP with *ERPLAB -> Compute averaged ERP*. Leave the default settings and say 'RUN', then name the erp 's11'. Now you can view the averaged ERP data by clicking on *ERPLAB -> Plot ERP waveforms*. You need to set a number of options here to make the figure more interpretable. First, you need to select the ERPs that you would like to plot—here, bins 1 and 2 (high and low probability targets). You should also push the toggle button that says 'positive is up' so that it now reads 'negative is up', since most of the literature in the course plots ERP waveforms with negative up. Under 'style', CLASSIC ERP should be selected. Now create the first plot with 'PLOT'.

14. Which condition should have the more negative peak, HighProb or LowProb?

15. If you look at electrodes that typically show the N400 effect like Cz, Cpz, or Pz, do they show an N400 difference between conditions for this subject? Illustrate with a screenshot.

Now, make a plot in which the waveforms are arranged in a more sensible distribution by selecting 'Topographic' under 'Style' on the plotting window, and click 'Plot'. An 'edit channel info' screen will now pop up, because the right mastoid (which we don't use in this lab) doesn't have a standard location. You can fix this by scrolling to channel number 28 on the 'Edit channel info' window (label 'RM') and unclicking the box that says 'Channel in data array'. Then click 'ok' and 'yes' and 'ok'. Now you should see the topographic plot.

16. Illustrate the topographic plot with a screenshot. What do you notice about the distribution of the N400 effect (which electrodes show a difference between conditions in the N400 time-window of ~300-500ms)?

17. Do you notice differences between conditions in any other time windows?

**Note! Make sure to press 'Cancel' when you are done with the plotting window. Otherwise nothing else in Matlab will run.

Finally, create a topoplot that schematizes the distribution of the difference between the two conditions in the N400 time-window. To do this, we first need to create a new bin that represents the difference between the two conditions. In the ERPLAB menu, go to 'ERP Operations -> ERP Bin operations'. The big white window on the left is the equation editor. Type in the equation b3=b2-b1. This will create a difference ERP that represents the subtraction LoProb-HiProb. It makes most sense to run the difference this way because our hypothesis is that less computations are required for the predictable target than the unpredictable target.

Now go to 'Plot ERP scalp maps'. You should select '3' for the first entry as the bin to plot. The latencies to plot are '300 500', the value to plot is the 'mean amplitude between two latencies', and under 'Options' you should click to include the color bar. Now you can plot.

18. What is the distribution of the reduced negativity for the highly probable items (in blue shades)? Illustrate with a screenshot. Is it consistent with an N400 effect?

V. GRAND AVERAGE ERP

For the last step, we'll take a look at the grand-average ERPs across all 36 subjects that participated in this study.

If you wanted to create this file yourself from the original individual ERPs, you would just need to go to ERPLAB -> Grand average and then choose 'From ERPset files', click on the 'Add Erpset' button, and highlight all the ERPs you want to grand-average. However, rather than give you all 36 individual ERP sets we have included the file 'ga_n36.erp' which contains the output of the grand-average step. You can load it by going to *ERPLAB -> Load existing ERPset*. Note that this file contains ALL of the events of interest from our previous experiment, but you should focus on conditions 1 and 2, which are the same HiProb and LoProb as above but have different labels in the full dataset.

Now plot the waveforms for conditions 1 and 2 (Bins 1 and 2), as you did above.

19. What pattern do you observe in the N400 time-window? Illustrate with a screenshot. Is the pattern across the group consistent with what was observed in subject 1?

Now plot the scalp map for the difference between the low and high probability conditions (this is Bin12 in the grand-average ERPset).

20. Is the pattern across the group relatively consistent with what was observed in subject 1? Illustrate with a screenshot.