Logistic Regression Toolbox – How to use

By: Jonny Giordano

Introduction: In this project, there will be 3 folders: “Data”, “EEGLAB14\_1\_2b”, and “Results”. There will also be many scripts and functions that will be used in order to carry out the analysis.

Step 1: Data Preparation (if Test.set and Train.set are present, skip Step 1!)

1. First the data must be prepared. Currently, the data is separated in Correct Rejection and Miss cases for each subject. We will divide each of these trials into “test” and “train” data sets. The classifier will learn on the training set, and be validated on the testing set. ~10% of the data is for testing, and ~90% is for training

run the script, “split\_into\_test\_and\_train.m” to do this.

1. The next step is to concatenate all the trials, and create labels from the EEG event types that the classifier will recognize. We will end up with two data files named “Train” and “Test”.

run the script, “recode\_subject.m”

Step 2: Basic Analysis

1. Now we can do a basic analysis to make sure the classifier works. An easy signal to decode in our EEG data is the saccadic direction, or more simply, the direction the eyes moved. To make the analysis simple, the script we will use averages the value of the horizontal electroocular electrodes (EOG) over the time course of each trial.

run the script “Classify\_Basic.m”

This script will call compute\_cost\_basic, which computes the cost function of our logistic classifier, extract\_data\_basic which extracts our data to classify from the appropriate data files. The script will also call plot\_boundary\_test, and plot\_boundary\_train which will plot the test and train data, along with the boundary that was learned during training. The training file by default is ’01cr.set’ and the test file is ’01miss.set’. We train on subject 01 in the correct rejection condition and test on subject 01 in the miss condition.

The results should be the same as those in Classify\_Basic\_Results.pdf

Step 3: Complex Saccadic Analysis

1. Instead of trying to classify an entire trial, we can classify each time point in a trial. The classifier will then tell us at what point the information is different between the two conditions. In EEG, this can give us a better understanding of the time frame of neural signals. We can test our classifier by running it every time point, first in the condition where we detect saccadic direction again as a sanity check. We should expect our best classification to happen shortly after the saccade occurs.

run Classify.m now

This script calls compute\_cost, extract\_data, balance\_classes, and classifier\_score which checks how well the classifier did at each time point. The results for the accuracy are then plotted. For computation time reasons, we analyze every 2nd time point.

The results should be those found in Classify\_Complex\_Results.pdf

Step 4: Answering our question

1. Now that we know our classifier works, we can test it on the question we want to answer. Can the classifier detect a difference between the conditions correct rejection and miss? If we can, this indicates that there is information about the target movement occurring during a saccade.

Run Classify.m, but change “experiment\_type” to “jump”

The results should be those found in Classify\_Movement\_Results.pdf