Computational Neuroscience Programming Contest: 2010
Sponsored by:
David M. Devilbiss, Ph.D.
Department of Psychology
University of Wisconsin – Madison

Rules

The challenge: Artifact Detection

Electroencephalography (EEG) is technique of recording electrical activity from the surface of the scalp of humans. Similarly, Electrocorticograph (ECoG) recordings are used in animal models where recordings are done from the surface of the brain. The electrical activity is generated by the coordinated discharge of neurons of the underlying cortex as they communicate with one another via action potentials. EEG recording can be obtained with a single recording electrode or with arrays of electrodes or an electrode net.

One of the most important signal processing steps in analyzing EEG data is to pre-process the data and remove contaminating artifacts. This continues to be an active area of research by trying to automatically detect signal drop-outs, muscle artifacts and ocular artifacts, as well as experimentally generated artifacts. However, most artifact detection is designed to detect artifacts that occur simultaneously across many simultaneously recorded channels of an EEG array (e.g. http://eeg.pl/Members/hubert/artefakt, http://sccn.ucsd.edu/wiki/Chapter_01:_Rejecting_Artifacts).

Animal ECoG may only contain a single channel of EEG data making this processes more difficult. Moreover, in some experimental setups artifact is generated by experimental stimulation of the brain. Below is an example output of an artifact detection algorithm we are currently using in our laboratory.

Figure 1: Example single channel ECoG (EEG) trace and artifact identification overlay. This time series illustrates the identification of three types of artifacts found in this trace; Extreme Artifacts that exceed a predefined voltage threshold such as amplifier saturation, dv/dt Artifacts that exceed predefined change in EEG slope caused by static, electrical stimulation, or other artificial electrical noise, and dropout Artifacts occurring from lost signal. The x-axis depicts data samples (Sample Rate = 1kHz), y-axis the recording voltage normalized to one for display purposes only.
Your Charge:

Your job is to write a MATLAB M-file to determine the occurrence of artifacts given a single channel of ECoG recorded from an animal. Given a vector which represents the voltage recorded at each time point your solution as many types of artifacts as it can in the shortest amount of CPU time.

In MATLAB syntax, the function header for your solution should look like this:

```matlab
function output = [programmername_date]_ArtifactDetection(input);
```

where programmer name is your surname and date is the revision date.

Input:

You will be given a \([n \times 1]\) vector. Each row represents the voltage recorded in a time bin (0.001 seconds) across the duration of the recording.

Sample vectors can be found at: https://mywebspace.wisc.edu/ddevilbiss/web/contest

Output:

A struct containing the following results:

- Output.Artifact => logical vector identifying all of the occurrences of an artifact (required).
- Output.xxx => logical vector where “xxx” is a description of the particular artifact subtype (optional).

Judging Criteria:

The allowable functions are those contained in the basic MATLAB package. These are ones available in $MATLAB/toolbox/matlab, where $MATLAB is the root MATLAB directory. Entries will be tested against MATLAB version 2007b 64bit. Functions from other toolboxes will not be available, however you can integrate freely available functions into your own code.

The following are prohibited:

- MEX-files
- eval, feval, etc.
- Shell escape such as !, dos, unix
- Handle Graphics commands
- ActiveX commands
- File I/O commands
- Debugging commands
- Printing commands
- Simulink commands
- Benchmark commands such as tic, toc, and flops

Entries will be judged on a combination factors including:

- number of correct answers
- etime()
- profile()
- McCabe Complexity
The winning entry is the one with the most correct answers and the lowest etime(), profile(), and McCabe Complexity scores.

Submission and Deadline:

The contest will close on April 1st 2010, at 5 PM central time.

At that time, the contest will be closed to new entries. Have all entries FTP'd to https://mywebspace.wisc.edu/ddevilbiss/web/contest/submissions

your function name (i.e. bob_15Feb2009_PSTHcalc) will identify each submission.

The Prizes:

Prizes will be awarded following the completion of the contest.

The author of the final winning entry in the contest will receive their choice of:

- University of Wisconsin swag
- Gift certificate
- Matlab swag

Winning entries must be original or must be substantial improvements over other entries. The contest administrators reserve the right to disqualify trivial changes which happen to result in better scores.