Details on your MATLAB Function Assignment for PSY696B

The aim of this assignment is to encourage you to use your new Neural Time Series Analysis skills to create a well-documented function in Matlab that does something useful with EEG data. Ideally, the functions written in this class can form a library of resources that other class members can use in processing their own data.

The specific operations of your function are up to you, but ideally they incorporate aspects of the Neural Time Series processing you learned about this semester. Because Mike Cohen, the author of your text has provided you sample code, this can help you see how to write good code. If your function uses any of the code, or adapts any of the code from Mike's code, you must indicate this in a comment before and after the relevant lines. So although you are not prohibited from using or adapting some of Mike's code, keep in mind: 1) You must acknowledge your use or alteration of his code; 2) At least 50% of your function code must be your own original code. Moreover, to the extent that you use or alter (and of course acknowledge) Mike's code, you should add comments to demonstrate that you fully understand what each line does.

The function could implement one of the many procedures we have discussed this semester. As class has proceeded, I have mentioned several examples as they occurred to me. I won't relist those that I recall here, for I'd like each of you to write a function about what really interests you. But if you are having troubles to think of something, please talk to me and we can discuss ideas.

Keep in mind that this is a function, not simply a script. Variables must be passed into and out of the function. The most versatile function will be able to pass portions of the EEG data structure into and out of the function, as this will allow others who work with other data formats to use the function. For example, it might be easy to write a function that passes in the entire structure like this:

```
EEG = myfunc(EEG, parameter1, parameter2);
```

But if someone does not have the entire EEG data structure (e.g., using intra-cranial local field potential data), I would be more useful to write a function that separates the parts of the EEG structure, like this:

```
New_EEG_data = myfunc(EEG.data, EEG.srate, EEG.times, parameter1, parameter2, parameter3);
```

The function should work (no crashes) and be something you can demonstrate to the class in 15-minute demonstration in the final two periods. In that 15 minute presentation, you should explain why your function is needed (what makes it useful), demonstrate its use, and show highlights of your code.

You should email your completed function (m file) and any necessary data (mat file) other than the sample data (if your function can use that) to the instructor by the last day of class (May 1). Your grade on this function will be determined by: 1) the instructor's evaluation of the code's accuracy and readability (50%); 2) Class ratings of your presentation and function by class members using a grading rubric (50%) that is presented below. I have also included a sample function (FIR_design.m) to give you an idea of how you might organize and document a good function.

Grading Rubric

- 10 points: Function has clear description of purpose (e.g., to design filter coefficients using method x)
- 10 points: Function usage clearly documented (i.e., what parameters must be passed into the function, and what are passed back out)
- 10 points: Code is adequately commented
  - Minimum of one comment per 3 lines of code
  - Each section is described in sufficient detail to understand what it does
- 10 points: Code is accurate! You can demonstrate this to yourself by testing obvious cases
- 10 points: Code runs! No errors returned if user entered all parameters required

By April 10, please send me a short description of what your function will do, along with your idea of what variables will be passed into and out of the function in a format like:

```
New_EEG_data = myfunc(EEG.data, EEG.srate, EEG.times, parameter1, parameter2, parameter3);
```