EEG GUI SCRIPT DOCUMENTATION   27/01/2016

       This document describes in detail the MATLAB scripts used to create and run the EEG Processing GUI. The GUI was created using MATLAB's Graphic User Interface Design Environment (GUIDE), which creates .m and .fig files for each GUI. The EEG Processing GUI has two GUIs that interact, the main GUI (eegGUI.m/.fig) and the data setup GUI (datasetup.m/.fig) that is called from the main GUI. In addition, the main GUI calls several different functions depending on the options selected. These will be described in order of the typical EEG processing pipeline used. For quick information about each function, type help (function name) in the MATLAB command window. See the EEG Processing GUI Manual for in-depth instructions on analyzing data using this GUI.

1. runEEG.m
    - Command used to open the EEG GUI (type runEEG in command window)
    - Clears workspace
    - Checks if EEGlab is in MATLAB path
    - Runs eegGUI.m
2. eegGUI.m
    - Main graphic user interface
    - Creates data structures for use in functions and calls functions depending on selected options
3. datasetup.m
    - Data selection graphic user interface (click on Data Setup in EEG GUI)
    - Determines subject/session/task/run selection and directory locations
4. chkactive.m
    - Determine active/inactive status of checkboxes/buttons in main GUI window
    - Based on whether relevant files can be found
    - Separate sections for Windows and Linux OS
5. reexport_log2text.m
    - Calls Linux script "find_missed_codes" if on Linux OS (otherwise skips this section)
    - Subject/Session/Task/Run loops
        o Loads logfile and gets Presentation codes, timing, and uncertainty for each event
        o Converts timing and uncertainty to ms and rounds values (**required for matching missing codes**)
        o Imports missing codes from Linux-created text file (*_MISSING_CODES.txt)
        o Matches missing codes with logfile and gets row #
        o Creates new matrix with converted timing (adding uncertainty) and event codes
        o Creates condition column for each task separately
            ▪ The condition names are based on the (TASK)_eventType.txt files and team discussion
        o Merges "condition" with timing and event codes (called "newtxt")
        o Delete irrelevant events (ITIs, "thanks", "QUIT", etc) and missing codes
        o Writes new text file as "*_fixed.txt" in logfile directory
        o Writes missing codes in importable format as "*_missingcodes.txt" in logfile directory
6. exportEEGevents.m
    - Creates "Triggers" directory in user-defined processing directory if it does not exist

- Loads subjinfo.csv or creates if it does not exist
- Subject/Session/Task loop
    o Determines if .raw or .mat data and gets run list (for .raw, these are filenames and for .mat, these are MATLAB variable names)
    o Run loop
        ▪ Finds subjinfo row and trigger file, skips subject if both exist
        ▪ Populates subjinfo row with relevant subject information
        ▪ Gets data/variable name or prompts to select file if none/multiple found
        ▪ Loads data file/variable
        ▪ Populates subjinfo with data/variable filename
        ▪ Exports trigger file as "*_triggers.txt" in trigger directory
        ▪ Gets logfile name ("*_fixed.txt") or prompts to select logfile if none/multiple found
        ▪ Inputs logfile in subjinfo
    o Saves subjinfo.csv and EEGevent_errors.txt in processing directory
        ▪ The csv file is saved using cell2dataset, which requires statistics toolbox
7. syncEEGtiming.m
    - Creates Processing directory ('ProcessedData') if none exists in user-defined processing directory
    - Loads syncinfo.txt (created from Excel file EEGSyncTiming.xlsm) if exists, otherwise produces an error message and quits the function
    - Subject/Session/Task loop
        o Creates subject-specific processing directory and gets run list depending on .raw/.mat
        o Run loop
            ▪ Finds syncinfo row and gets relevant information
            ▪ Creates EEG structure and adds to ALLEEG
            ▪ Loads Presentation text file defined in syncinfo.txt
            ▪ Adds sync value as well as software and hardware offsets defined in syncinfo.txt to Presentation timing
                • Offsets were manually determined for each task separately
            ▪ If final run
                • Replaces spaces in Presentation event codes with underscores
                    o Necessary when importing Presentation timing to ALLEEG
                • Merges runs in ALLEEG into a single structure
                • Adds software and hardware offsets to EEG DIN latencies
                • Selects whether to use EEG DIN latencies or Presentation timing
                    o This depends on netstation verion and testing location
                    o Netstation 4: using Presentation timing, because EEG timing did not capture all events (USB cable)
                    o Netstation 5: uses EEG timing if the number of events is equal between EEG and Presentation (all events captured - parallel port)
                        ▪ CFRI: removes DIN7s and non-DIN/boundary events
                        ▪ UBC: removes DIN4s and non-DIN/boundary events

- ▪ For a currently unknown reason, these events are not captured by Presentation (they are also not important)
- ▪ Checks that events are equal
- ▪ If equal, uses EEG timing, otherwise uses Presentation timing and outputs error
- Writes final timing (EEG/Presentation) to text file with Presentation conditions and event codes
- Writes a "*_timecomp.txt" file to allow for comparison between EEG and Presentation events in the case of unequal events
- Imports final timing/conditions/event codes to merged EEG structure
- Creates text file with latencies between DINs and imported events (for checking)
- Saves merged EEG structure as "*_raw.set" file
- Writes sync_errors.txt file to save/processing directory
- Writes averaged event_latencies.txt file to save/processing directory (uses cell2dataset)
8. plot_saved_sets.m
9. processEEG.m
10. plot_ICA_comps.m