```
%run this function to connect and plot raw EEG data
%choose which waves and mindwave params to show
%make sure to change portnum1 to the appropriate COM port
%based on readRAW.m

clear all
close all
test_time = 1; %predicted time (in minutes) of

test. overestimate
%preallocate storage for all samples in a structure
[data_att,...
    data_med,...
    data_raw,...
    data_delta,...
    data_theta,...
    data_alpha1,...
    data_alpha2,...
    data_beta1,...
    data_beta2,...
    data_gamma1,...
    data_gamma2,...
    data_blink] = deal(zeros(1,512*60*test_time));

portnum1 = 4;   %COM Port #
comPortName1 = sprintf('\\\\.\\COM%d', portnum1);


% Baud rate for use with TG_Connect() and TG_SetBaudrate().
TG_BAUD_57600 =      57600;


% Data format for use with TG_Connect() and TG_SetDataFormat().
TG_STREAM_PACKETS =     0;


% Data type that can be requested from TG_GetValue().
TG_DATA_POOR_SIGNAL =     1;
TG_DATA_ATTENTION =       2;
TG_DATA_MEDITATION =      3;
TG_DATA_RAW =             4;
TG_DATA_DELTA =           5;
TG_DATA_THETA =           6;
TG_DATA_ALPHA1 =          7;
TG_DATA_ALPHA2 =          8;
TG_DATA_BETA1 =           9;
TG_DATA_BETA2 =          10;
TG_DATA_GAMMA1 =         11;
TG_DATA_GAMMA2 =         12;
TG_DATA_BLINK_STRENGTH = 37;

%load thinkgear dll
if not(libisloaded('Thinkgear.dll'))
    loadlibrary('Thinkgear.dll')
end
```

```
fprintf('Thinkgear.dll loaded\n');

%get dll version
dllVersion = calllib('Thinkgear', 'TG_GetDriverVersion');
fprintf('ThinkGear DLL version: %d\n', dllVersion );
```

```
Undefined variable "test" or class "test.overestimate".

Error in real_time_plots2 (line 11)
test. overestimate
```

Get a connection ID handle to ThinkGear

```
connectionId1 = calllib('Thinkgear', 'TG_GetNewConnectionId');
if ( connectionId1 < 0 )
    error( sprintf( 'ERROR: TG_GetNewConnectionId() returned %d.\n', connectionId1 ) );
end;

% Set/open stream (raw bytes) log file for connection
errCode = calllib('Thinkgear', 'TG_SetStreamLog', connectionId1, 'streamLog.txt' );
if( errCode < 0 )
    error( sprintf( 'ERROR: TG_SetStreamLog() returned %d.\n', errCode ) );
end;

% Set/open data (ThinkGear values) log file for connection
errCode = calllib('Thinkgear', 'TG_SetDataLog', connectionId1, 'dataLog.txt' );
if( errCode < 0 )
    error( sprintf( 'ERROR: TG_SetDataLog() returned %d.\n', errCode ) );
end;

% Attempt to connect the connection ID handle to serial port "COM3"
errCode = calllib('Thinkgear', 'TG_Connect',  connectionId1,comPortName1,TG_BAUD_57600,TG_STR
EAM_PACKETS );
if ( errCode < 0 )
    error( sprintf( 'ERROR: TG_Connect() returned %d.\n', errCode ) );
end

fprintf( 'Connected.  Reading Packets...\n' );
%This means connected to TGConnect, not necessarily to headset.
```

```
%record and plot data
%I'm worried that reading all of the available values may be slow. If it
%   seems like data points are getting skipped, try commenting out any
%   unnecessary calllib lines below.

%Set up figure layout
%Just the power spectrum plot for now
hfig1 = figure('Name', 'SSVEP Frequency Matching Test');
hax = axes('Parent', hfig1);

l = 1024; %determines length of signal to fft and plot
j = -255; %keeps track of every half second (256 samples), doesn't plot first half second
```

```
i = 0; %keeps track of full length of test
k = 1; %keeps track of current plot indices
while (i < 512*60*test_time)   %stop when record is full
    if (calllib('Thinkgear','TG_ReadPackets',connectionId1,1) == 1)   %if a packet was read..
.

        if (calllib('Thinkgear','TG_GetValueStatus',connectionId1,TG_DATA_RAW) ~= 0)   %if RA
W has been updated
            j = j + 1;
            i = i + 1;
            data_att(i) = calllib('Thinkgear','TG_GetValue',connectionId1,TG_DATA_ATTENTION);
 % attention data
            data_med(i) = calllib('Thinkgear','TG_GetValue',connectionId1,TG_DATA_MEDITATION)
; % meditation data
            data_raw(i) = calllib('Thinkgear','TG_GetValue',connectionId1,TG_DATA_RAW); % raw
 data
            data_delta(i) = calllib('Thinkgear','TG_GetValue',connectionId1,TG_DATA_DELTA); %
 delta data
            data_theta(i) = calllib('Thinkgear','TG_GetValue',connectionId1,TG_DATA_THETA); %
 theta data
            data_alpha1(i) = calllib('Thinkgear','TG_GetValue',connectionId1,TG_DATA_ALPHA1);
 % alpha1 data
            data_alpha2(i) = calllib('Thinkgear','TG_GetValue',connectionId1,TG_DATA_ALPHA2);
 % alpha2 data
            data_beta1(i) = calllib('Thinkgear','TG_GetValue',connectionId1,TG_DATA_BETA1); %
 beta1 data
            data_beta2(i) = calllib('Thinkgear','TG_GetValue',connectionId1,TG_DATA_BETA2); %
 beta2 data
            data_gamma1(i) = calllib('Thinkgear','TG_GetValue',connectionId1,TG_DATA_GAMMA1);
 % gamma1 data
            data_gamma2(i) = calllib('Thinkgear','TG_GetValue',connectionId1,TG_DATA_GAMMA2);
 % gamma2 data
            data_blink(i) = calllib('Thinkgear','TG_GetValue',connectionId1,TG_DATA_BLINK_STR
ENGTH); % blink strength data
        end
    end

    if (j == 256) && (i<512*60*test_time-512)
        %figure out indices to plot using k
        %plots results for last l measurements, update every .5 seconds
%         PowerSpectrum(hax, data_raw(k:k+l-1), 512);
        figure(1)
        pwelch(data_raw(k:k+l-1), [], [], 1:40, 512)
        ylim([0 50])
        figure(2)
        pyulear(data_raw(k:k+l-1), 4, 1:40, 512)
        ylim([0 50])
        drawnow
        k = k + 256;
        j = 0;
    end

end


%disconnect
calllib('Thinkgear', 'TG_FreeConnection', connectionId1 );
```

```
% unloadlibrary('Thinkgear.dll')
save('data_raw_15_a.mat', data_raw)
```

---

*Published with MATLAB® R2015b*