# Appendix: MATLAB Code for Estimating Subjects’ Concentration Level[[1]](#footnote-1)

% Define pipeline

opts = { ...

 'FilterOrdering', { ...

 'flt\_clean\_settings', 'flt\_rescale' ...

 'flt\_fir' 'flt\_fourier\_bandpower'} ...

 'DataCleaning', { ...

 'DataSetting', {'1.1-beta' ...

 'FlatlineRemoval', 'on' ...

 'DriftCutoff', [1 2] ...

 'BadChannelRemoval', 'on' ...

 'ChannelDropoutRepair', 'off' ...

 'BadSubspaceRemoval', { ...

 'Cutoff', 10 ...

 'ReferenceExtraction', { ...

 'MaxBadChannels', 0.3 ...

 'ParameterFitting', { ...

 'StepSizes', [0.001 0.01]}}} ...

 'BadWindowRemoval', 'off'}} ...

 'Rescale', { ...

 'ScaleFactor', 0.001} ...

 'FIRFilter', { ...

 'Frequencies', [40 45] ...

 'Mode', 'lowpass' ...

 'Type', 'linear-phase' ...

 'StopbandRipple', -30 ...

 'NormalizeAmplitude', true} ...

 'FourierBandPower', { ...

 'Filtering', { ...

 'Representation', {'multitaper' ...

 'TimeBandwidth', 5}} ...

 'Bands', {[13 18], [4 12]} ...

 'FreqCollapse', 'mean' ...

 'AverageChannels', true}};

opts = {arg\_guipanel('Function',@flt\_pipeline,'Parameters',opts,'PanelOnly',false)};

% Load some calibration data (e.g. a short artifact-free segment)

calibData = exp\_eval(io\_loadset('data:\calibration.xdf'));

% Calibrate pipeline

cleaned\_data = exp\_eval(flt\_pipeline('signal',calibData,opts{:}));

% Subscribe to MQTT topic and republish as local (named) LSL stream

... we do this by creating a timer object whose callback pulls ...

... new MQTT data and appends into LSL stream data buffer ...

... we create objects mindoPub and mindoSub for MQTT pub/sub ...

% Initialize online pipeline

pipeline = onl\_newpipeline(cleaned\_data,lsl\_streamname);

% Main loop

while true

 % pull data block from LSL stream and process through pipeline

 [eeg\_chunk,pipeline] = onl\_filtered(pipeline,round(5\*cleaned\_data.srate));

 % ... spectral power for [beta] and [alpha+theta] avg'd over channels

 % ... now stored them in eeg\_chunk.data(1) and eeg\_chunk.data(2)

 % compute log beta /(alpha+theta) power ratio

 mqttData = log(eeg\_chunk.data(1)./eeg\_chunk.data(2));

 % ... publish result over MQTT ...

 mindoPub.updateMsg(num2str(mqttData(:)));

end

1. Published with MATLAB® 7.14 [↑](#footnote-ref-1)