Table of Contents

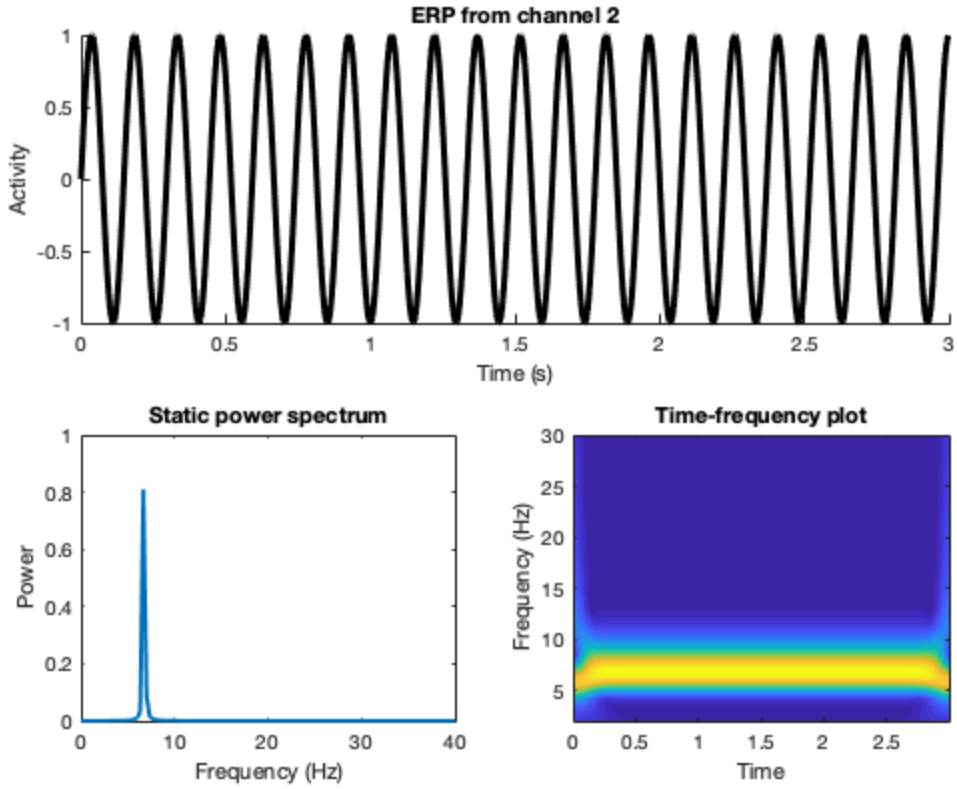
Please note that this project is from a course by Mike x Cohen

INSTRUCTIONS:

The goal of this assignment is to simulate time series data that can be used to test time-series analysis methods. For each section below: 1) Complete the MATLAB code 2) Put the data into the EEG structure - Make sure all relevant fields are accurate (EEG.data, EEG.pnts, EEG.trials, EEG.srate, EEG.nbchan, EEG.times) 3) Use function plot_simEEG to plot some data

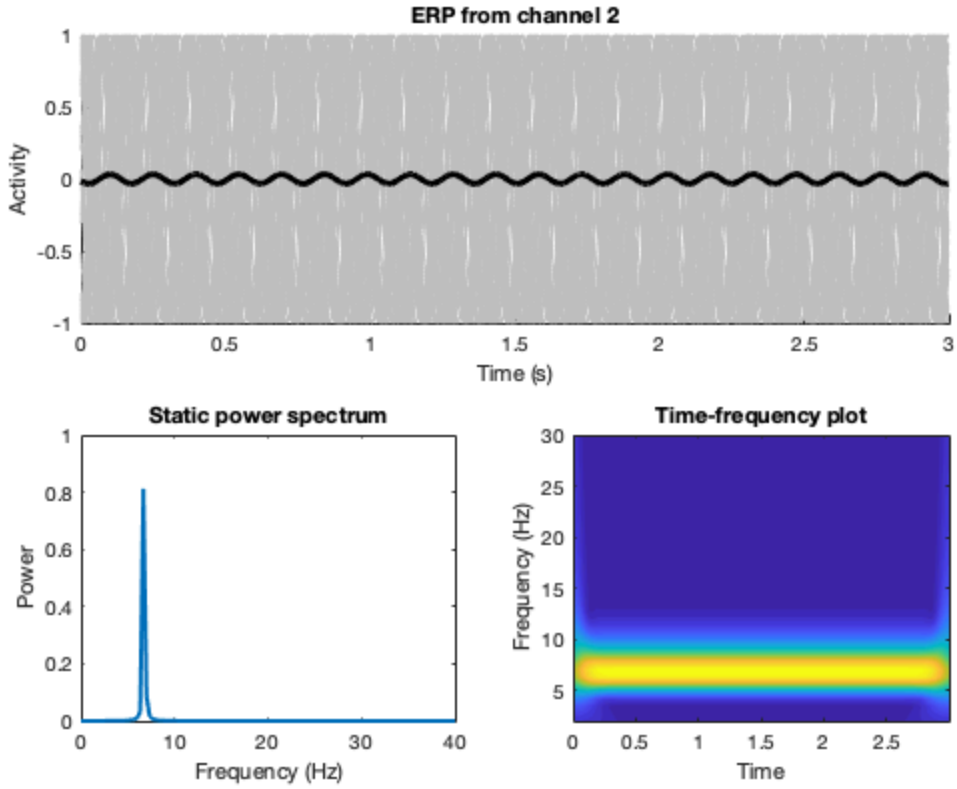
1) pure phase-locked sine wave

```
% parameters
EEG.srate  = 500; % sampling rate in Hz
EEG.pnts   = 1500;
EEG.trials = 30;
EEG.nbchan = 23;

sinefreq = 6.75; % in Hz

% time vector
EEG.times = (0:EEG.pnts-1)/EEG.srate;


% loop over channels and create data
for chani=1:EEG.nbchan
    for triali=1:EEG.trials
        % data as a sine wave
        EEG.data(chani,:,triali) = sin(2*pi*sinefreq*EEG.times);
    end
end

% the function below takes at least one argument (EEG),
% and optionally a second argument (channel number),
% and optionally a third argument (figure number)
plot_simEEG(EEG,2,1)
```



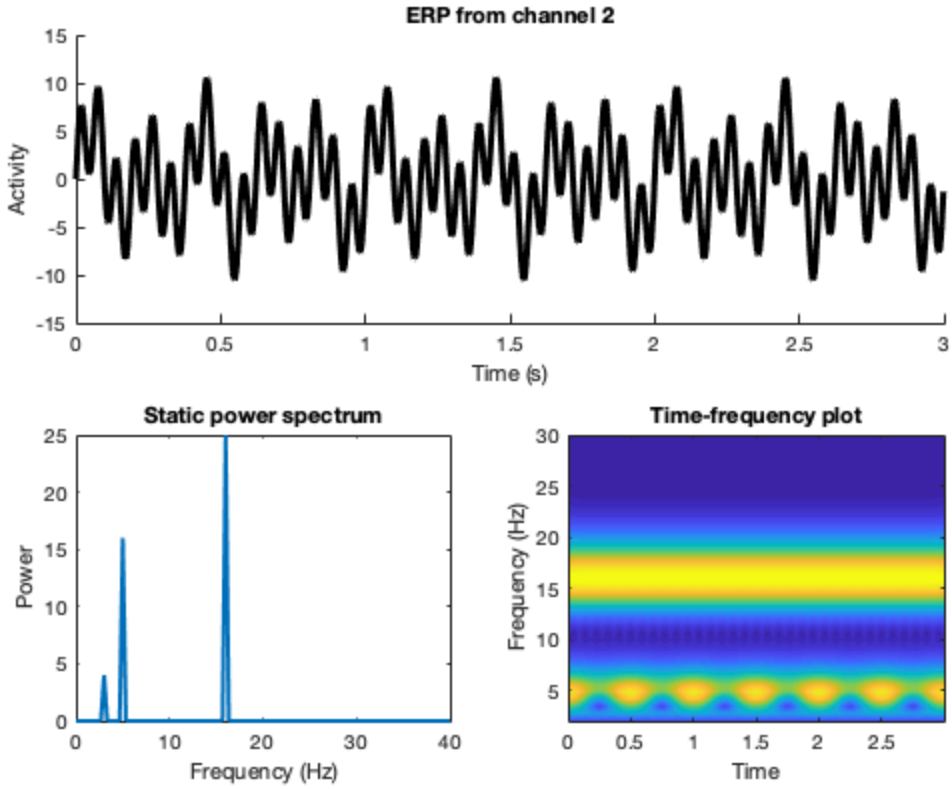
2) pure non-phase-locked sine wave

```
% loop over channels and create data
for chani=1:EEG.nbchan
    for triali=1:EEG.trials
        % data as a sine wave
        EEG.data(chani,:,triali) = sin(2*pi*sinefreq*EEG.times +
 2*pi*rand);
    end
end

% the function below takes at least one argument (EEG),
% and optionally a second argument (channel number),
% and optionally a third argument (figure number)
plot_simEEG(EEG,2,2)
```



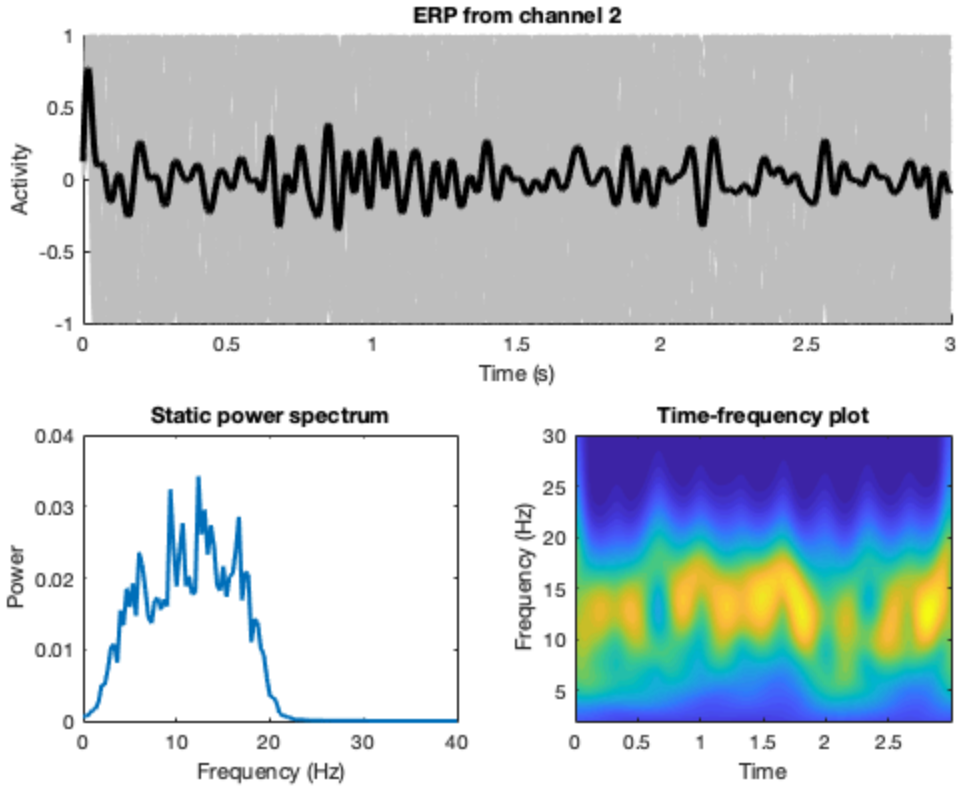
3) multisine waves

```
% list of frequencies and corresponding amplitudes
frex = [ 3 5 16 ];
amps = [ 2 4 5  ];


% loop over channels and trials
for chani=1:EEG.nbchan
    for triali=1:EEG.trials

        sinewave = zeros(1,EEG.pnts);
        for si=1:numel(frex)
            sinewave = sinewave +
 amps(si)*sin(2*pi*frex(si)*EEG.times);
        end


        % data as a sine wave
        EEG.data(chani,:,triali) = sinewave;
    end
end

plot_simEEG(EEG,2,3)
```



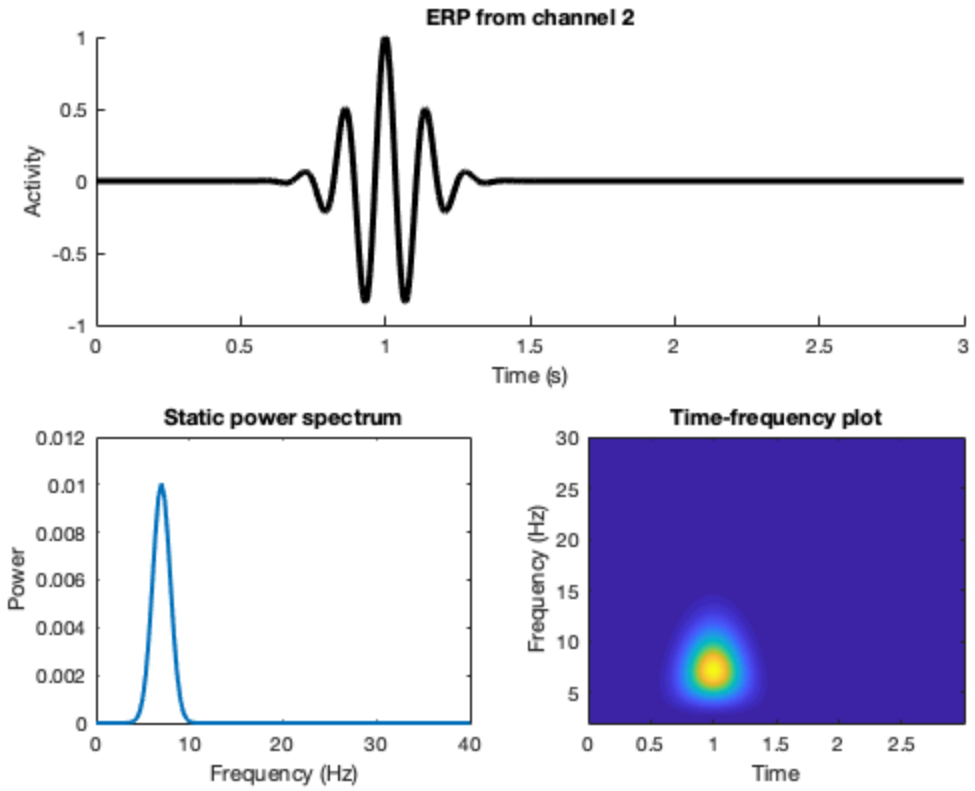
4) nonstationary sine waves

```
% loop over channels and trials
for chani=1:EEG.nbchan
    for triali=1:EEG.trials


        freqmod = 20*interp1(rand(1,10),linspace(1,10,EEG.pnts));
        signal  = sin( 2*pi * (EEG.times + cumsum(freqmod)/
EEG.srate) );

        EEG.data(chani,:,triali) = signal;
    end
end

plot_simEEG(EEG,2,4)
```



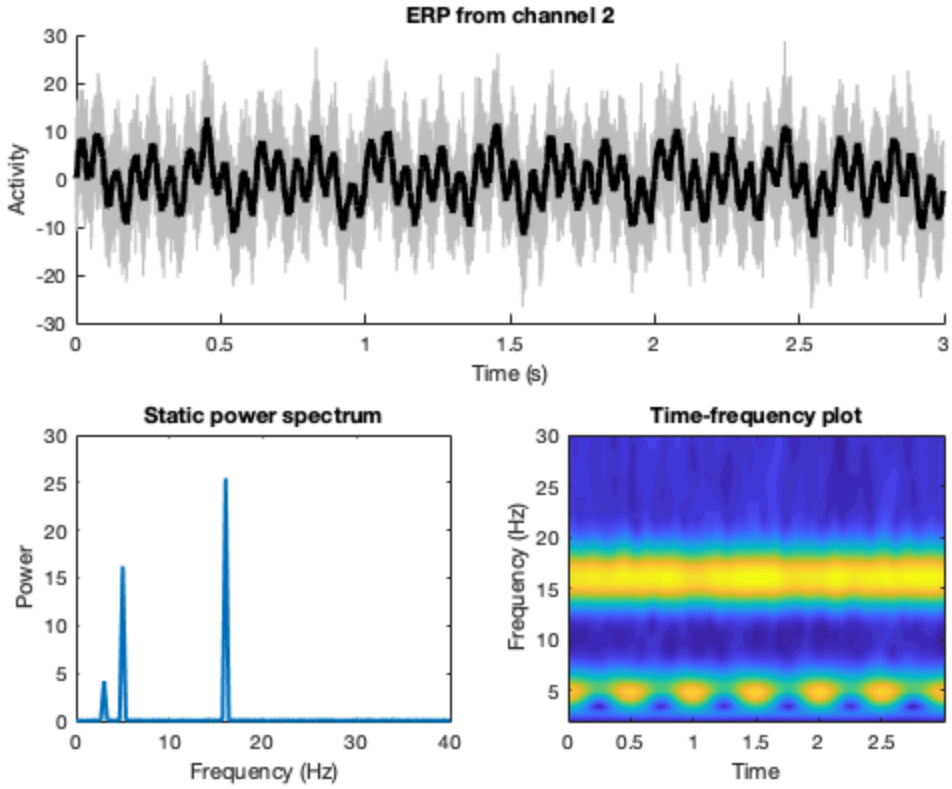
5) transient oscillations w/ Gaussian

```
% Gaussian and sine parameters
peaktime = 1; % seconds
width    = .12;
sinefreq = 7; % for sine wave

% create Gaussian taper
gaus = exp( -(EEG.times-peaktime).^2 / (2*width^2) );


% loop over channels and trials
for chani=1:EEG.nbchan
    for triali=1:EEG.trials

        % trial-unique sine wave
        cosw = cos(2*pi*sinefreq*EEG.times);

        EEG.data(chani,:,triali) = cosw .* gaus;
    end
end

plot_simEEG(EEG,2,5)
```



6) repeat #3 with white noise

```
% list of frequencies and corresponding amplitudes
frex = [ 3 5 16 ];
amps = [ 2 4 5  ];


% loop over channels and trials
for chani=1:EEG.nbchan
    for triali=1:EEG.trials


        sinewave = zeros(1,EEG.pnts);
        for si=1:numel(frex)
            sinewave = sinewave +
 amps(si)*sin(2*pi*frex(si)*EEG.times);
        end


        % data as a sine wave plus noise
        EEG.data(chani,:,triali) = sinewave + 5*randn(size(sinewave));
    end
end

plot_simEEG(EEG,2,6)
```



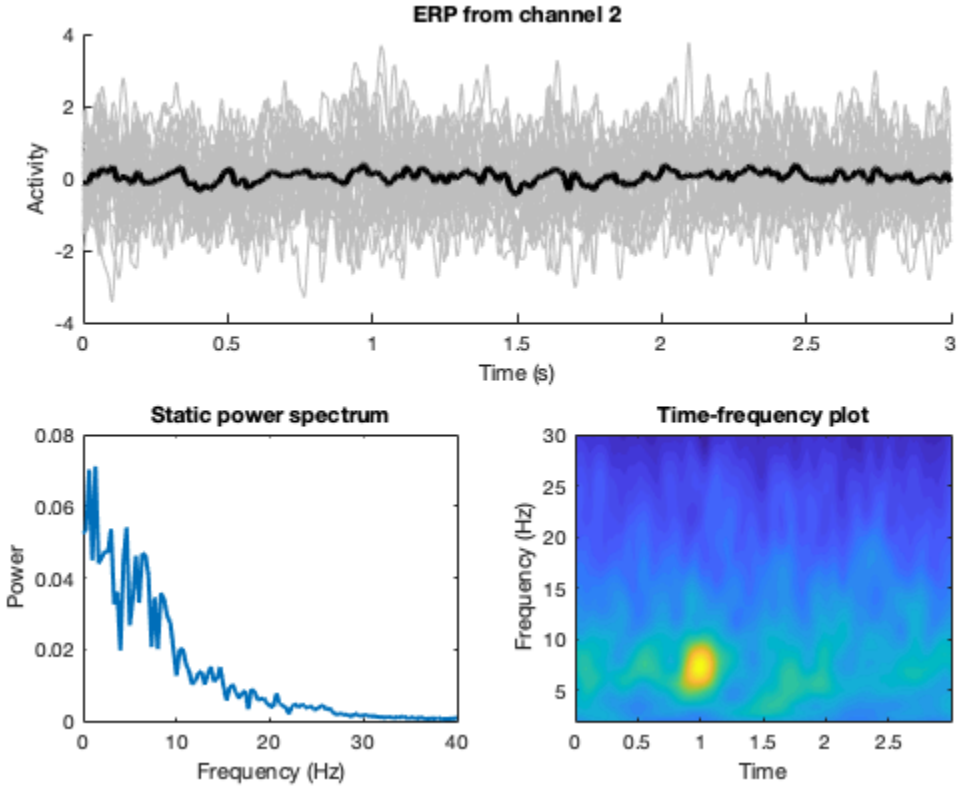
7) repeat #5 with 1/f noise

```
% amount of noise
noiseamp = .3;


peaktime = 1; % seconds
width    = .12;
sinefreq = 7; % for sine wave

% create Gaussian taper
gaus = exp( -(EEG.times-peaktime).^2 / (2*width^2) );


% loop over channels and trials
for chani=1:EEG.nbchan
    for triali=1:EEG.trials

        % trial-unique sine wave
        cosw = cos(2*pi*sinefreq*EEG.times + 2*pi*rand);
```

1/f noise

```
        ed = 50; % exponential decay parameter
```

```
        as = rand(1,floor(EEG.pnts/2)-1) .* exp(-
(1:floor(EEG.pnts/2)-1)/ed);
        as = [as(1) as 0 as(:,end:-1:1)];

        % Fourier coefficients
        fc = as .* exp(1i*2*pi*rand(size(as)));

        % inverse Fourier transform to create the noise
        noise = real(ifft(fc)) * EEG.pnts;


        % data as signal + noise
        EEG.data(chani,:,triali) = cosw .* gaus + noiseamp*noise;
    end
end

plot_simEEG(EEG,2,7)
```



Published with MATLAB® R2020b