

```
% file SIOC 221A HW 4
%
% author Julia Dohner, with help from Annie Adelson, Jacob Morgan and Luke
% Kachelein
%
% due date October 26, 2017
clear all; close all;

%% gathering 2015 Pier pressure record

% create empty arrays to hold time and temp data
time = [];
pressure = [];

time = [time; ncread(strcat('http://sccoos.org/thredds/dodsC/autoss/scripps_pier-2015.nc'), 'time')];
pressure = [pressure; ncread(strcat('http://sccoos.org/thredds/dodsC/autoss/scripps_pier-2015.nc'), 'pressure')];

% remove bad data using the flagged data from .nc file
pressure_flagPrimary = [];
pressure_flagPrimary = [pressure_flagPrimary; ncread(strcat('http://sccoos.org/thredds/dodsC/autoss/scripps_pier-2015.nc'), 'pressure_flagPrimary')];

% looping through to remove bad data from pressure record
for i = 1:length(pressure)
    if pressure_flagPrimary(i) ~= 1
        pressure(i) = nan;
    end
end

%% consider a period with equal increments
% this for the first 34 days of the 2015 record

% time differences
time = double(time);
t_diff = diff(time);

% finding segment of data with even spacing
cutoff = find(t_diff(1:82236)>t_diff(1),1);
pressure_sub = pressure(1:cutoff-1); % subsampled pressure
time_sub = time(1:cutoff-1); % subsampled times
date0=datenum(1970,1,1); % give reference date (first date)
time_sub = double(time_sub)/86400+date0; % in units of days (conversion: seconds/day)

figure
hold on

plot(time_sub, pressure_sub, 'LineWidth',1)

ax = gca;
xlabel('\fontsize{12}2015')
ylabel('\fontsize{12}Pressure (dbar)')
```

```
title('fontsize{14}2015 Scripps Pier Pressure')
datetick('x','mmm dd','keepticks')

% Fourier transforming my data

f = fft(pressure_sub); % Fourier transform of my data
data = f(1:3590); % only using first half of vector
N = length(pressure_sub);
frequency = (0:7180)/(7181*361)*(24*3600);
frequency = frequency(1:3590)'; % only using first half of vector
data = data/N; % Normalizing the data

figure
semilogy(frequency,abs(real(data)), '-b')
hold on
semilogy(frequency,abs(imag(data)), '-m');
title('fontsize{14}Real and imaginary parts of Fourier transform of 2015 Scripps Pier pressure');
legend('fontsize{12}real parts','imaginary parts');
xlabel('fontsize{12}cycles per day (cpd)');
ylabel('fontsize{12}Pressure (dbar)');

figure
semilogy(frequency,abs(data), '-b')
title('fontsize{14}Fourier transform of 2015 Scripps Pier pressure');
xlabel('C\fontsize{12}ycles per day (cpd)');
ylabel('fontsize{12}Pressure (dbar)');
xlim([0 125]);

%% mean pressure and amplitudes of major peaks
% using Fourier coefficients to find mean pressure, amplitudes of peaks

% mean pressure is the first value in the Fourier coefficient vector
meanPressure = data(1);

absData = abs(data);

% sorting peaks into descending order
[amp ind] = sort(absData, 'descend');
amplitudes(:,2) = amp(1:5);
amplitudes(:,1) = ind(1:5); % stores the indices (col 1) and amplitudes (col 2) of major peaks

% multiply amplitudes by 2 to account for both positive and negative
% frequencies
amplitude_a = 2*amplitudes(1,2); % this is the mean
amplitude_b = 2*amplitudes(2,2); % largest non-mean peak
amplitude_c = 2*amplitudes(3,2); % second-largest non-mean peak
amplitude_d = 2*amplitudes(4,2); % third-largest non-mean peak

% indices of peaks
index_cpd_a = amplitudes(1,1);
index_cpd_b = amplitudes(2,1);
index_cpd_c = amplitudes(3,1);
index_cpd_d = amplitudes(4,1);

% frequencies of peaks
cpd_a = frequency(index_cpd_a);
cpd_b = frequency(index_cpd_b);
```

```
cpd_c = frequency(index_cpd_c);
cpd_d = frequency(index_cpd_d);

% periods of peaks
%cpd_a_period = 24/cpd_a;
cpd_b_period = 24/cpd_b;
cpd_c_period = 24/cpd_c;
cpd_d_period = 24/cpd_d;

%% spectral energy vs. frequency

spectralE = abs(data).^2;

figure
semilogy(frequency,spectralE,'-b')
title('\fontsize{14}Spectral Energy of Scripps Pier pressure in August 2015');
xlabel('\fontsize{12}Cycles per day (cpd)');
ylabel('\fontsize{12}dbar^2 / cpd');

% taking derivative of my data
diff_data = diff(pressure_sub);
f_diff = fft(diff_data); % Fourier transform of my data
data_diff = f_diff(1:3590);
data_diff = data_diff/N; % Normalizing the data

spectralE_diff = abs(data_diff).^2;

figure
semilogy(frequency,spectralE_diff,'-b')
title('\fontsize{14}Spectral Energy of time derivative of 2015 Scripps Pier pressure');
xlabel('\fontsize{12}Cycles per day (cpd)');
ylabel('\fontsize{12}(dbar/time)^2 / cpd');
```