SIOC 221A Homework 3 **Julia Dohner**
Prof. Sarah Gille Due Date: October 19th, 2017

1. Visual evaluation.

The data are for the most part uniformly spaced by 361 seconds, aside from later in the record where there seems to be a huge break of 17 days (max(X) = 1476982 seconds), and some places where the data is taken at closer intervals (min(X) = 220 seconds). The plots of the time series of pressure data from 2015 and from the first month of 2015 are displayed below.

2. Least-squares fit.

The mean value from the least-squares fit of the 2015 pressure record is 3.49 dbar (first element in the x matrix). The total amplitudes of the O1, M2 and K1 tidal constituents are 0.17, 0.52, and 0.38 dbar, respectively. (See MATLAB code lines 72-95 for method.) A plot of the least squares fit overlaid on the pressure data of the first month of 2015 is displayed below.

3. Stationarity of the tide.

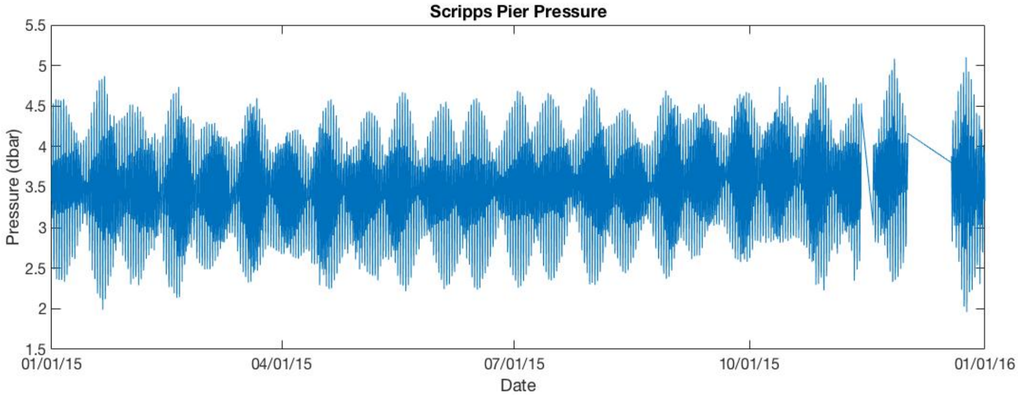
The tidal amplitudes for the K1 tide decreases by 0.07. The amplitude of M2 decreases by 0.01, and the amplitude of O1 does not change. The tidal component that changes the most is the K1 tide. (See MATLAB code lines 98-103 for method.) Because the earth rotates on a tilted axis relative to our axis of rotation around the sun, the distance of the pier from the sun changes on a seasonal scale, changing the influence of solar forcing on local tides. The K1 tide is the only of the three tides that contains a component reflecting the influence of the sun on tides, thus it makes sense that K1 shows the biggest change in amplitude between January and August. A plot of the least squares fit overlaid on the pressure data of a 30-day summer subset is displayed below.

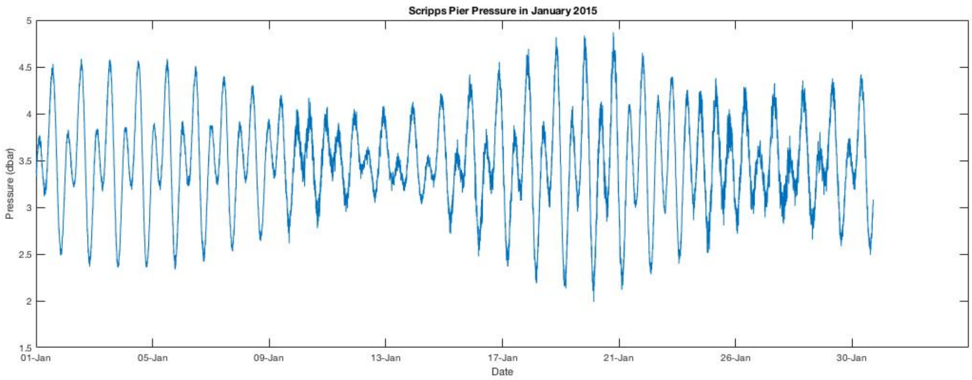
4. χ^2 and the misfit.

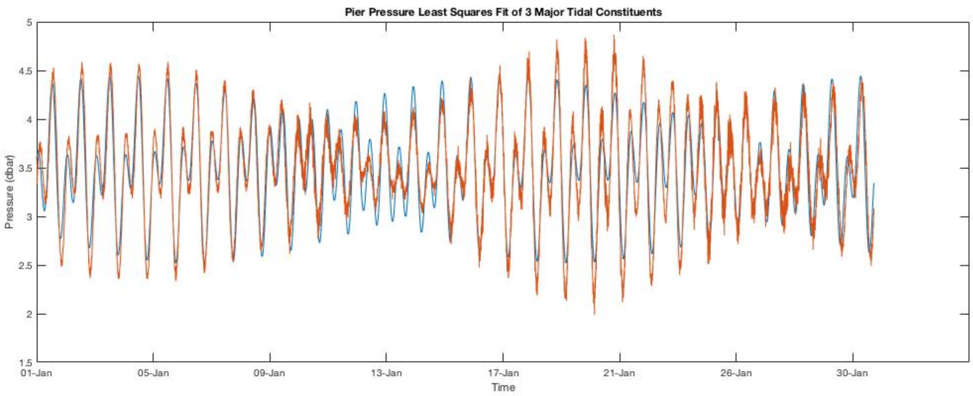
In my understanding, our model A we've chosen represents all of the important variability in our data. Thus any other variability not represented in A will contribute to the uncertainty sigma. If I opt to use the standard deviation of the entire dataset, I will be folding in additional uncertainty due to other tidal forcings and natural variability that can predict the pressure data but that I have not included in my model. Instead, we want to look at high frequency noise, so it makes more sense to look at the standard deviation across a much smaller subset of the data. One option is to look at the standard deviation at the peak (or trough) of a tide so that we can say we're only observing variability when the tide is stationary (not influenced by (potentially under-fitted) tides). What I chose to do was use the standard deviation of 2-point segments, computed as the standard deviation of the difference between adjacent points.

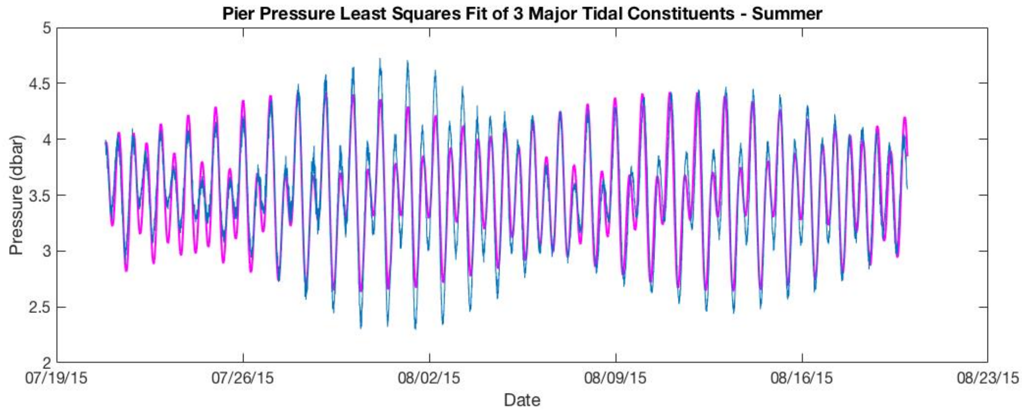
The squared misfit of my least-squares fit is 71,822. (See MATLAB code lines 154-178 for method.) This is one order of magnitude greater than the squared misfit I'd expect from N-M (which, when fitting three tidal frequencies, is 7,174). The misfit decreases by 61,725 to 10,097 when I fit with 5 tidal frequencies instead of 3 (where N-M = 7,170).

To decide if a reduced misfit was sufficient to justify fitting additional frequencies, I'd compare the results of the incomplete gamma functions for χ^2 of each of the fits. If p is close to 1, then the data is likely being over-fit. The text Numerical Recipes in C: the Art of Scientific Computing (the only version of the text available online through Roger) includes a figure (Figure 6.2.1 on page 217) offers a way of evaluating your value of p, but it does not include lines for the values of a used in my calculations.









Pier Pressure Least Squares Fit of 3 Major Tidal Constituents - Summer
Pressure (dbar)
5
4.5
4
3.5
3
2.5
2
07/19/15
07/26/15
08/02/15
08/09/15
08/16/15
08/23/15
Date

```
% file SIOC 221A HW 3
%
% author Julia Dohner, with help from Luke Kachelein, Dillon Amaya, and
% Annie Adelson
%
% due date October 19, 2017

clear all; close all;

numYears = 2017-2005 + 1;

%% plotting the Scripps Pier 2015 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

% examining the time increments between adjacent measurements
X = diff(time);
%figure
%plot(X);
%plot(X(1:4000))

% plot the time series
date0=datenum(1970,1,1);% give reference date (first date)
time2 = double(time)/24/3600+date0;% divide the time by 24*3600 to convert seconds into↙
days since 1970
figure('name','Scripps_Pier_Pressure_2015');
plot(time2, pressure,'LineWidth',1);

% label the x-axis in months
set(gca,'FontSize',16);
title('Scripps Pier Pressure');
xlabel('Date');
datetick('x','mm/dd/yy')
ylabel('Pressure (dbar)');


%% plotting just the first month of 2015
```

```
% 2592000 seconds in 30 days
% If a measurement is taken every 361 seconds, then 30 days into the record
% should be roughly the first 7180 measurements (2592000/361) in the 2015
% series. My record is 30 days long.
time3 = time(1:7180);
pressure2 = pressure(1:7180);

% plot the time series
date0=datenum(1970,1,1);% give reference date (first date)
time4 = double(time3)/24/3600+date0;% divide the time by 24*3600 to convert seconds into
days since 1970
time4_string = datestr(time4);
figure('name','Scripps_Pier_Pressure_January_2015');
plot(1:length(pressure2),pressure2,'LineWidth',1);
title('Scripps Pier Pressure in January 2015');
xlabel('Date');
set(gca, 'xtick', 1:1000:length(pressure2),'xticklabel', time4_string(1:1000:length
(pressure2),1:6));
ylabel('Pressure (dbar)');

%% Least squares fit

% defining sine and cosine components of major tidal constituents

% convert period to days (to match x axis time units)
O1_sin = sin(2*pi*time4/(25.83/24));%O1: principal lunar diurnal
O1_cos = cos(2*pi*time4/(25.83/24));
K1_sin = sin(2*pi*time4/(23.93/24));%K1: luni-solar diurnal
K1_cos = cos(2*pi*time4/(23.93/24));
M2_sin = sin(2*pi*time4/(12.42/24));%M2: principal lunar
M2_cos = cos(2*pi*time4/(12.42/24));


A2=[ones(length(time4),1) O1_sin O1_cos K1_sin K1_cos M2_sin M2_cos];
x2=inv(A2'*A2)*A2'*pressure2;
figure('name','Pier_Pressure_Tidal_LSF');
matrixProd = A2*x2;
plot(1:length(matrixProd),matrixProd,'LineWidth',1);
hold on
plot(1:length(pressure2),pressure2,'LineWidth',1);
set(gca, 'xtick', 1:1000:length(pressure2),'xticklabel', time4_string(1:1000:length
(pressure2),1:6));
title('Pier Pressure Least Squares Fit of 3 Major Tidal Constituents');
xlabel('Time');
ylabel('Pressure (dbar)');


% Total amplitude = square root of the sum of the squares of the sine and
% cosine amplitudes)
% Units of mean and amplitude are decibars
amplitude_O1_jan = sqrt((x2(2,1))^2 + (x2(3,1))^2);
amplitude_K1_jan = sqrt((x2(4,1))^2 + (x2(5,1))^2);
amplitude_M2_jan = sqrt((x2(6,1))^2 + (x2(7,1))^2);

%% Stationarity of the tide

% Repeating the least squares fit for 30 days roughly near August 2015
```

```
% Starting 7/12 of the way through the time record (82237 measurements).
% If a measurement is taken every 361 seconds, then 30 days into the record
% should be roughly the next 7180 measurements (2592000/361). My record is
% 30 days long.
kAugustStart = floor((7/12)*82237);
kAugustEnd = kAugustStart + 7180;
timeAugust = time(kAugustStart:kAugustEnd, 1);
timeAugust = double(timeAugust)/24/3600+date0;% in units of days
pressureAugust = pressure(kAugustStart:kAugustEnd, 1);

% defining sine and cosine components of major tidal constituents

% convert period to days (to match x axis time units)
O1_sin_aug = sin(2*pi*timeAugust/(25.83/24));%O1: principal lunar diurnal
O1_cos_aug = cos(2*pi*timeAugust/(25.83/24));
K1_sin_aug = sin(2*pi*timeAugust/(23.93/24));%K1: luni-solar diurnal
K1_cos_aug = cos(2*pi*timeAugust/(23.93/24));
M2_sin_aug = sin(2*pi*timeAugust/(12.42/24));%M2: principal lunar
M2_cos_aug = cos(2*pi*timeAugust/(12.42/24));


A3=[ones(length(timeAugust),1) O1_sin_aug O1_cos_aug K1_sin_aug K1_cos_aug M2_sin_aug
M2_cos_aug];
x3=inv(A3'*A3)*A3'*pressureAugust;
figure('name','Pier_Pressure_Tidal_LSF_Summer');
plot(timeAugust,A3*x3,'m','LineWidth',2)
hold on
plot(timeAugust, pressureAugust,'LineWidth',1);
set(gca,'FontSize',16);
title('Pier Pressure Least Squares Fit of 3 Major Tidal Constituents - Summer');
xlabel('Date');
datetick('x','mm/dd/yy')
ylabel('Pressure (dbar)');


% The mean is 3.4898 (first row in x2 vector)
%
% Total amplitude = square root of the sum of the squares of the sine and
% cosine amplitudes)
% Units of mean and amplitude are decibars
amplitude_O1_aug = sqrt((x3(2,1))^2 + (x3(3,1))^2);
amplitude_K1_aug = sqrt((x3(4,1))^2 + (x3(5,1))^2);
amplitude_M2_aug = sqrt((x3(6,1))^2 + (x3(7,1))^2);


%% Chi squared and the misfit. (Good name for a short story)

% y is vector containing pressure data (pressureAugust)
% A is the matrix containing ones, sines and cosines (A3)
% x is matrix containing mean and amplitudes (x3)

% looping to create array of differences between two adjacent points
diffArray = [];
for i = 1:(length(pressureAugust)-1);
    diffArray(i) = pressureAugust(i+1) - pressureAugust(i);
end

sigma = std(diffArray);
```

```
% initializing variable before loop
chiSquared = 0;

for i = 1:length(pressureAugust)
    ax_prod = 0;
    for j = 1:length(x3)
        ax_prod = ax_prod + (A3(i,j)*x3(j));
    end
    chiSquared = chiSquared + ((pressureAugust(i) - ax_prod)^2);
end
chiSquared = chiSquared / (sigma^2);% sigma is out of for loop because is
% already a sum of uncertainties of individual points


% How much does the misfit change if you fit with 5 frequencies instead of
% 3?

% convert period to days (to match x axis time units)
S2_sin_aug = sin(2*pi*timeAugust/(12/24));%S2: principal solar semidiurnal
S2_cos_aug = cos(2*pi*timeAugust/(12/24));
N2_sin_aug = sin(2*pi*timeAugust/(12.66/24));%N2: larger lunar elliptic semidiurnal
N2_cos_aug = cos(2*pi*timeAugust/(12.66/24));


A4=[ones(length(timeAugust),1) O1_sin_aug O1_cos_aug K1_sin_aug...
    K1_cos_aug M2_sin_aug M2_cos_aug S2_sin_aug S2_cos_aug N2_sin_aug N2_cos_aug];
x4=inv(A4'*A4)*A4'*pressureAugust;

% recalculate misfit now with the 5-frequency fit (using A4 and x4)

chiSquared_5 = 0;

for i = 1:length(pressureAugust)
    ax_prod_5 = 0;
    for j = 1:length(x4)
        ax_prod_5 = ax_prod_5 + (A4(i,j)*x4(j));
    end
    chiSquared_5 = chiSquared_5 + ((pressureAugust(i) - ax_prod_5)^2);
end
chiSquared_5 = chiSquared_5 / (sigma^2);% sigma is out of for loop because is
% already a sum of uncertainties of individual points

nu = length(pressureAugust)-length(x3);% number of DOF (N-M)
nu_5 = length(pressureAugust)-length(x4);% number of DOF (N-M)

p = gammainc(chiSquared/2, nu/2);
p_5 = gammainc(chiSquared_5/2, nu_5/2);

p_subtraction = gammainc(nu/2, nu/2);
p_subtraction_5 = gammainc(nu_5/2, nu_5/2);
```