SIO221A Homework #2 (Eric Gallimore)

Output follows program listing.

```
% The ocean is stratified.  The ~1% difference between surface and deep waters inh
%  A density measuring instrument called a CTD was suspended from a boom on the Re
%
%  The PATCHEX_ISO.mat data file that you have gives the variations in depth of a
%    ? = ? (?i, tj ) = ?i j
%    {? i} represents a set of densities which are separated by 1m in the vertical
%
%
% Task 1:    Check out the data:
% time is in the first row of the data matrix, the depth of isopycnal surfaces in
%  type “time=piso(1,:);eta=-piso(2:end,:);”
% (time is in decimal Yeardays, depth is in meters. The minus sign above makes dep
% type:
%  “plot(time,eta(1:10:end,:)’)”
%
function hw2()
piso_file = load('patchex_iso.mat');

% Make a structure so we can use nice names.
piso.time = piso_file.piso(1,:);
piso.eta = -(piso_file.piso(2:end,:));

figure(1);
clf();
plot(piso.time, piso.eta(1:10:end, :), '.');
title('Some isopycnals measured by a CTD')
ylabel('Depth of isopycnal (m)')
xlabel('Time (days)')


%       Note the regions where the data are clearly noisy
%  Form a new array of isopycnal displacements by taking each isopycnal time serie

% First, let's find the means.
for i=1:size(piso.eta,1)
    piso.eta_tmean(i) = mean(piso.eta(i,:));
end

% Now, find the displacements
% The fastest way to to this in matlab is another for loop.  I miss list
% comprehensions.
for i=1:size(piso.eta,1)
    piso.eta_disp(i,:) = piso.eta(i,:) - piso.eta_tmean(i);
end

%  dispij= ?(i,:) - mean (? (i, :));
%  Produce a plot of dispij using Matlab routines “mesh” or “surf”.
```

```
%

figure(2);
clf;
mesh(piso.eta_disp);
title('Isopycnal depth displacement from mean')
ylabel('Depth of isopycnal (m)')
xlabel('Sample')
zlabel('Depth deviation from mean (m)')

%
% Task 2:  Produce a single estimate of the probability density function Pd of iso
%  Normalize such that
%  Plot, labeling all axes!!!
%  Write your own algorithm to estimate the pdf, do not use “hist”
%
% Combine all the data

i = (100:150);

data_a = reshape(piso.eta_disp(i,:), [],1);
figure(3);
clf();
plot_pdf(data_a, 0.75, 'b');
title('PDF of (selected) displacement data');
ylabel('Probability density (1/m)');
xlabel('Isopycnal Displacement from Mean (m)');

% Task 3: Repeat Task 2 using all good data from depths i= 400-500.Use ?disp of 0.
%
hold on;
data_b = reshape(piso.eta_disp(400:500,:), [],1);
plot_pdf(data_b, 0.4, 'r');

disp('Task 3: The displacement variance is greater for the range 400:500, by visua


% Task 4: Using a ?displacement of 0.4 m, form joint pdfs
%  P(disp1 , disp2) for
%   A)
%       disp1 = dispi,j    i=100-150
%        disp2 = dispi+10,j          i.e. two isopycnals separated by 10m.
%

% Find the index of two isopycnals separated by 10m.
% First, pick a starting isopycnal depth, say, -100m.
% Find it in the first matrix.
% This was apparently not the right thing to do.
%i_top = find(piso.eta(:,1) <= -100, 1);
%i_bot = find(piso.eta(:,1) <= -110, 1);

% Now, just reshape that data:
data_i10j = reshape(piso.eta_disp(i+10,:), [],1);
```

```
figure(4);
clf();
plot_joint_pdf(data_a, 0.4, data_i10j, 0.4);
ylabel('disp1, Displacement (m)');
xlabel('disp2, Displacement (m)');
zlabel('Probability Desnsity (1/m^2)');
title('Joint PDF for displacement from mean: disp1(i=100:150) and disp2(i=110:160)

% Plot the same thing with view(0,90);
% Note that this is not the best way to generate a plot of this style,
% because the z-axis label is not visible.  We could fix that by using
% heatmap.
% Additionally, it's bad, since we are regenerating data we just generated.
% If this took longer than 100ms to run, we might care.
figure(5);
clf();
[joint_pdf, bins_a, bins_b] = plot_joint_pdf(data_a, 0.4, data_i10j, 0.4,1);
ylabel('disp1, Displacement (m)');
xlabel('disp2, Displacement (m)');
zlabel('Probability Desnsity (1/m^2)');
title('Joint PDF for displacement from mean: disp1(i=100:150) and disp2(i=110:160)

% Task 5: Using the joint PDF estimate from Task 4, A, calculate the
%   A) mean of disp2       < disp2>
%   B) variance of disp2
%   C) covariance of < disp1 disp2>
% Do the statistics.  The hard way.
% Mean
% Multiply each value by its corresponding probabilty density and then sum
% those values.
disp2_mean_frompdf = sum(reshape(bsxfun(@times, bins_b, joint_pdf), 1, []));
% Also do it the sane way, to check the code above.  It won't match, but
% should be close.
disp2_mean = mean(data_i10j(:));

% Variance.  Variance is the mean of the squared differences from the mean.
disp2_var_frompdf = sum(reshape(bsxfun(@times, (bins_b - disp2_mean_frompdf).^2, j
% Check our work...
disp2_var = var(data_i10j(:));

% Covariance
% Get the mean
disp1_mean = mean(data_a);
% I love bsxfun.
disp12_cov_frompdf = sum(reshape( bsxfun(@times,(bins_b - disp2_mean_frompdf)' * (
% Check it.
disp12_cov = cov(data_a(:), data_i10j(:));
disp12_cov = disp12_cov(1,2); % Non diagonal elements are covariance.

% Now, print the results:

fprintf('\nTask 5 (using 4A data)\n');
fprintf('\tMean(disp2) = %d\n', disp2_mean_frompdf);
fprintf('\tVariance(disp2) = %d\n', disp2_var_frompdf);
```

```
fprintf('\tCovariance(disp1,disp2) = %d\n', disp12_cov_frompdf);


disp('Check against built-in functions:')
fprintf('\tMean(disp2) = %d\n', disp2_mean);
fprintf('\tVariance(disp2) = %d\n', disp2_var);
fprintf('\tCovariance(disp1,disp2) = %d\n', disp12_cov);



%   B)
%                          disp1 = dispi,j       i=100-150
%        disp2 = dispi+50,j        i.e. two isopycnals separated by 50m.
%

% Find the index of two isopycnals separated by 50m.
% First, pick a starting isopycnal depth, say, -100m.
% Find it in the first matrix.
% Again, not what we wanted.
%i_top = find(piso.eta(:,1) <= -100, 1);
%i_bot = find(piso.eta(:,1) <= -150, 1);

% Now, just reshape that data:
data_i50j = reshape(piso.eta_disp(i+50,:), [],1);

figure(6);
clf();
plot_joint_pdf(data_a, 0.4, data_i50j, 0.4);
ylabel('disp1, Displacement (m)');
xlabel('disp2, Displacement (m)');
zlabel('Probability Desnsity (1/m^2)');
title('Joint PDF for displacement from mean: disp1(i=100:150) and disp2(i=150:200)

figure(7);
clf();
[joint_pdf, bins_a, bins_b] = plot_joint_pdf(data_a, 0.4, data_i50j, 0.4, 1);
ylabel('disp1, Displacement (m)');
xlabel('disp2, Displacement (m)');
zlabel('Probability Desnsity (1/m^2)');
title('Joint PDF for displacement from mean: disp1(i=100:150) and disp2(i=150:200)

% Task 5: Using the joint PDF estimate from Task 4, B, calculate the
%   A) mean of disp2       < disp2>
%   B) variance of disp2
%   C) covariance of < disp1 disp2>
% Do the statistics.  The hard way.
% Mean
% Multiply each value by its corresponding probabilty density and then sum
% those values.
disp2_mean_frompdf = sum(reshape(bsxfun(@times, bins_b, joint_pdf), 1, []));
% Also do it the sane way, to check the code above.  It won't match, but
% should be close.
disp2_mean = mean(data_i50j(:));
```

```
% Variance.  Variance is the mean of the squared differences from the mean.
disp2_var_frompdf = sum(reshape(bsxfun(@times, (bins_b - disp2_mean_frompdf).^2, j
% Check our work...
disp2_var = var(data_i50j(:));

% Covariance
% Get the mean
disp1_mean = mean(data_a);
% I love bsxfun.
disp12_cov_frompdf = sum(reshape( bsxfun(@times,(bins_b - disp2_mean_frompdf)' * (
% Check it.
disp12_cov = cov(data_a(:), data_i50j(:));
disp12_cov = disp12_cov(1,2); % Non diagonal elements are covariance.

% Now, print the results:

fprintf('\nTask 5 (using 4B data)\n');
fprintf('\tMean(disp2) = %d\n', disp2_mean_frompdf);
fprintf('\tVariance(disp2) = %d\n', disp2_var_frompdf);
fprintf('\tCovariance(disp1,disp2) = %d\n', disp12_cov_frompdf);


disp('Check against built-in functions:')
fprintf('\tMean(disp2) = %d\n', disp2_mean);
fprintf('\tVariance(disp2) = %d\n', disp2_var);
fprintf('\tCovariance(disp1,disp2) = %d\n', disp12_cov);


% Verify that the joint PDF is doing the right thing.
figure(8);
clf();
plot_joint_pdf(data_a, 0.4, data_a, 0.4, 1);

fprintf('\n');
disp('Why is the variance bigger in one case and the covariance bigger in the othe
fprintf('\n');
disp('Qualitatively, the covariance is bigger in the case where the two ')
disp('data sets are more tightly correlated, and the variance is bigger')
disp('in the case where the data in one set deviates further from the mean.')
disp('Really, there in no reason why we would expect the variance and the')
disp('covariance to be the same.')
disp('Perhaps there is a more profound truth here that escapes me.')

% Please turn in printouts of your programs along with the plots



end


function plot_pdf(data, bin_size, plot_arg)
    [pdf_values, bins, interval] = get_pdf(data, bin_size);

    plot((bins - (interval / 2)), pdf_values, plot_arg);
```

```
    %bar((x_bins - (x_interval / 2)), x_hist, 1);
end


function [pdf_values, bins, interval] = get_pdf(data, bin_size)
    min_value = min(data);
    max_value = max(data);
    interval = bin_size;
    bins = (min_value + interval):interval:max_value;

    x_work = data;
    cumulative = 0;
    in_bin = zeros(1,length(bins));
    below_bin = zeros(1,length(bins));
    for i = (1:1:length(bins))
        in_bin(i) = length(x_work(x_work <= bins(i)));
        cumulative = cumulative + in_bin(i);
        below_bin(i) = cumulative;
        x_work = x_work(x_work > bins(i));
    end

    pdf_values = in_bin / length(data) / interval;
    cdf_values = below_bin / length(data);

end

function [joint_pdf, bin_centers_a, bin_centers_b] = get_joint_pdf(data_a, bin_siz
    min_value_a = min(data_a);
    max_value_a = max(data_a);
    interval_a = bin_size_a;
    bins_a = (min_value_a + interval_a):interval_a:max_value_a;

    min_value_b = min(data_b);
    max_value_b = max(data_b);
    interval_b = bin_size_b;
    bins_b = (min_value_b + interval_b):interval_b:max_value_b;

    in_bin = zeros(length(bins_a),length(bins_b));
    for i_a = (1:1:length(bins_a))
        index_a = find((bins_a(i_a)-interval_a <= data_a) & (data_a <= bins_a(i_a)
        for i_b = (1:1:length(bins_b))
            in_bin(i_a, i_b) = length(...
                data_b((bins_b(i_b)-interval_b <= data_b(index_a)) ...
                & (data_b(index_a) <= bins_b(i_b))));
        end
    end

    % Now, scale it appropriately.
    joint_pdf = in_bin ./ length(data_b) ./ length(interval_b);

    % Also, figure out the bin centers (useful for plotting)
    bin_centers_a = bins_a - (interval_a / 2);
    bin_centers_b = bins_b - (interval_b / 2);
```

```
    % Verify that the integral is 1.
    %sum(joint_pdf(:))

end

function [joint_pdf, bin_centers_a, bin_centers_b] = plot_joint_pdf(data_a, bin_si
    [joint_pdf, bin_centers_a, bin_centers_b] = get_joint_pdf(data_a, bin_size_a,

    if nargin < 5
        surf(bin_centers_b, bin_centers_a, joint_pdf);
    else
        surf(bin_centers_b, bin_centers_a, joint_pdf);
        view(0,90);
        colorbar();
    end

end
```

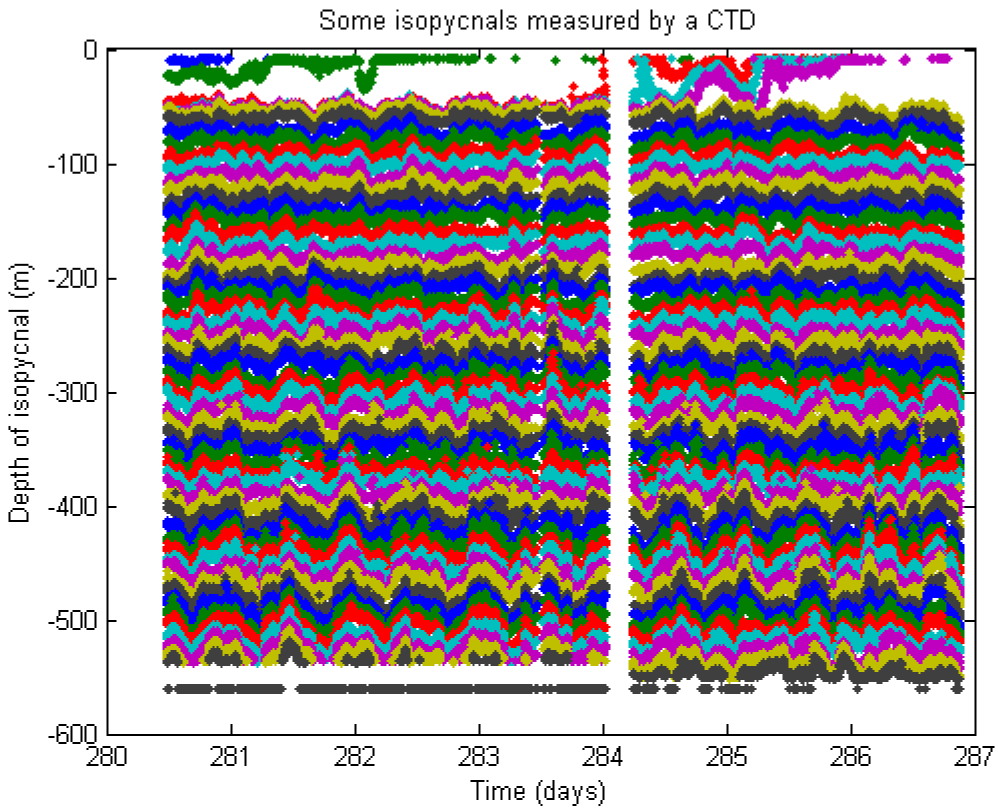
Task 3: The displacement variance is greater for the range 400:500, by vis

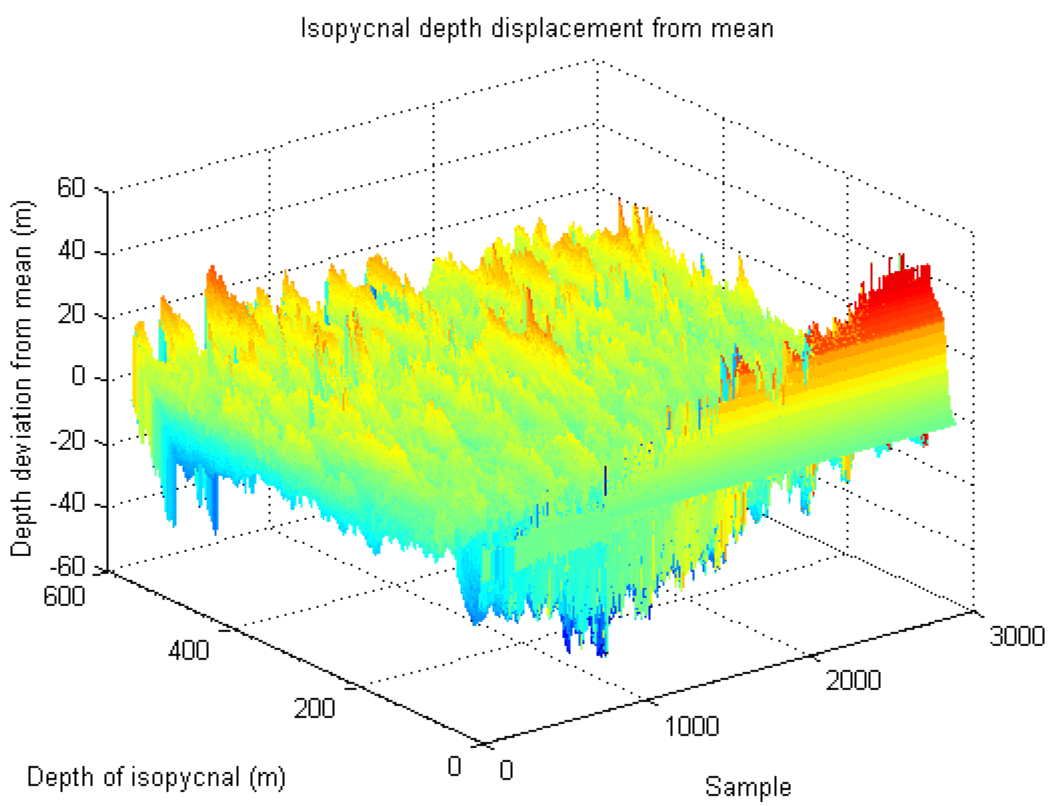
Task 5 (using 4A data)
Mean(disp2) = -8.968205e-004
Variance(disp2) = 1.767833e+001
Covariance(disp1,disp2) = 1.516827e+001
Check against built-in functions:
Mean(disp2) = 7.950648e-015
Variance(disp2) = 1.767728e+001
Covariance(disp1,disp2) = 1.517896e+001

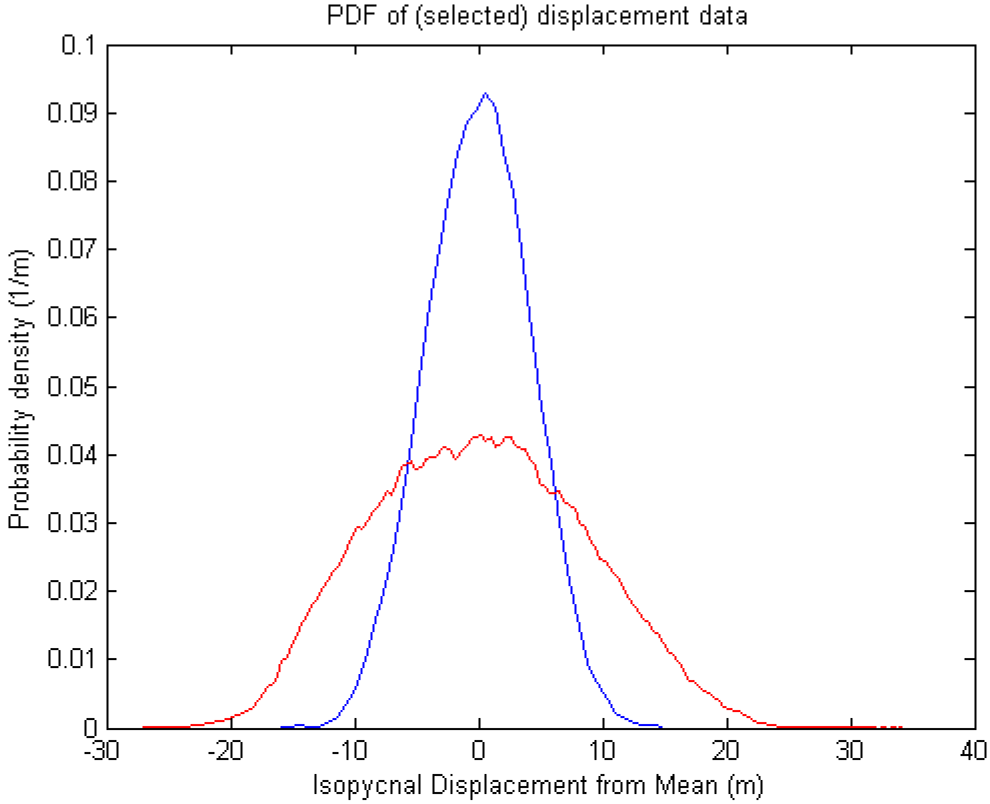
Task 5 (using 4B data)
Mean(disp2) = -5.316420e-004
Variance(disp2) = 2.182372e+001
Covariance(disp1,disp2) = 9.743972e+000
Check against built-in functions:
Mean(disp2) = -1.295386e-014
Variance(disp2) = 2.181217e+001
Covariance(disp1,disp2) = 9.749020e+000

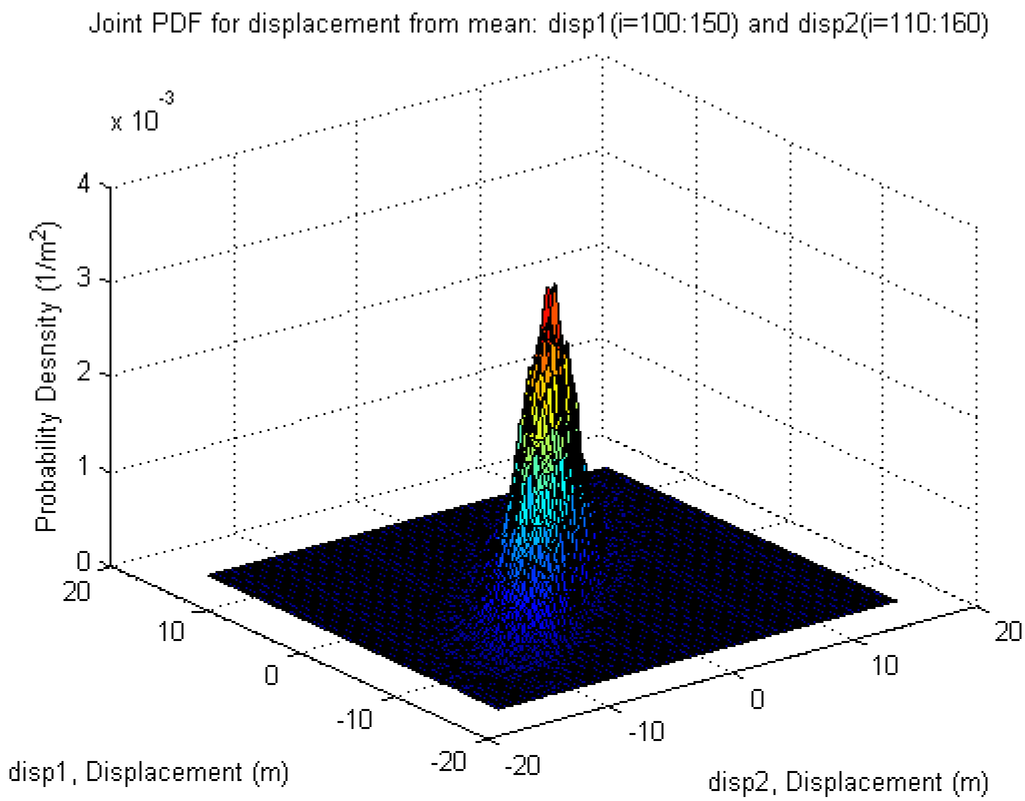
Why is the variance bigger in one case and the covariance bigger in the ot

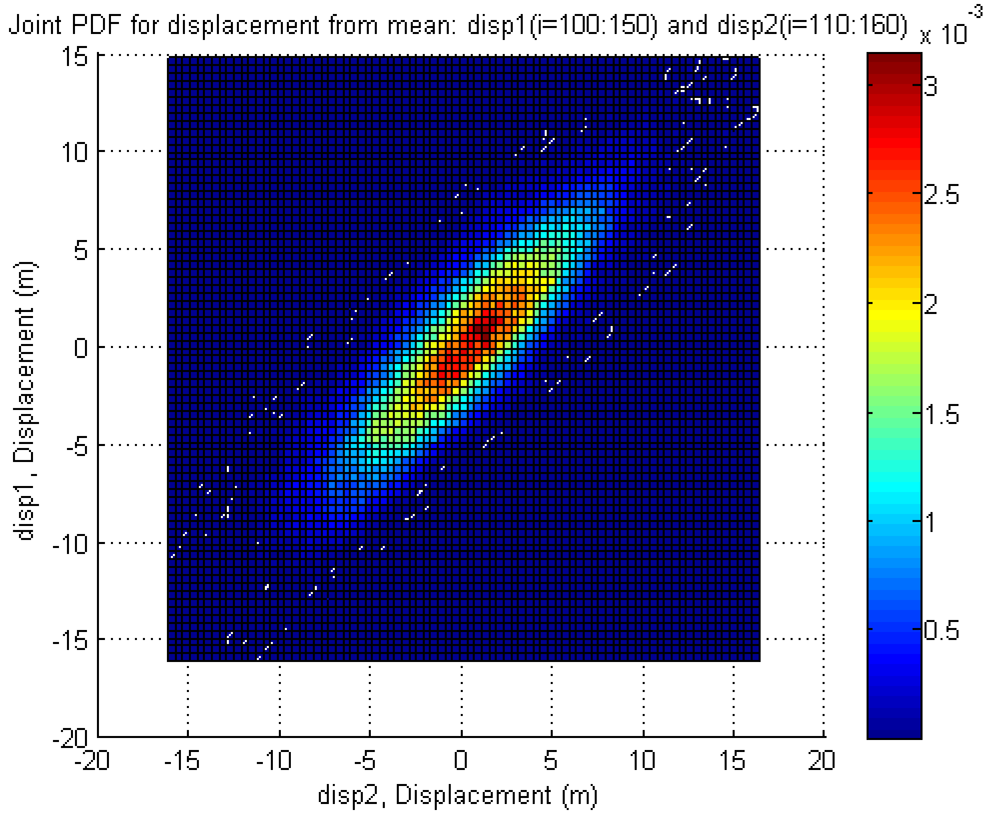
Qualitatively, the covariance is bigger in the case where the two data sets are more tightly correlated, and the variance is bigger in the case where the data in one set deviates further from the mean. Really, there in no reason why we would expect the variance and the covariance to be the same. Perhaps there is a more profound truth here that escapes me.

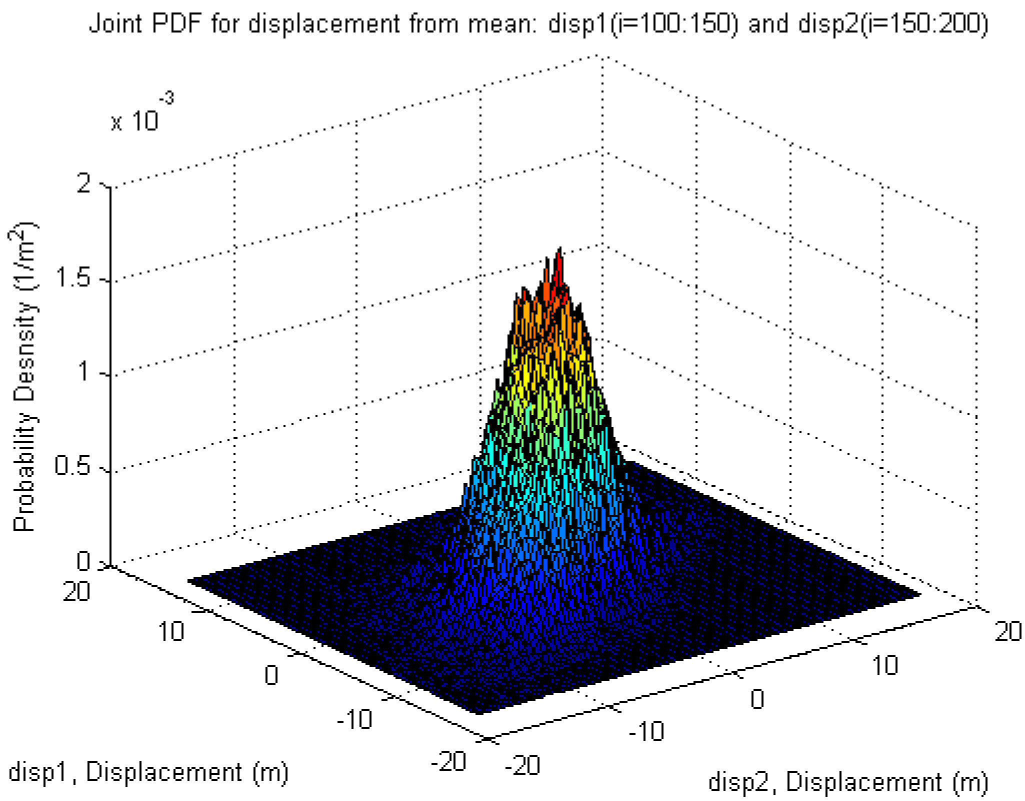

Some isopycnals measured by a CTD
Depth of isopycnal (m)
Time (days)
0
-100
-200
-300
-400
-500
-600
280
281
282
283
284
285
286
287

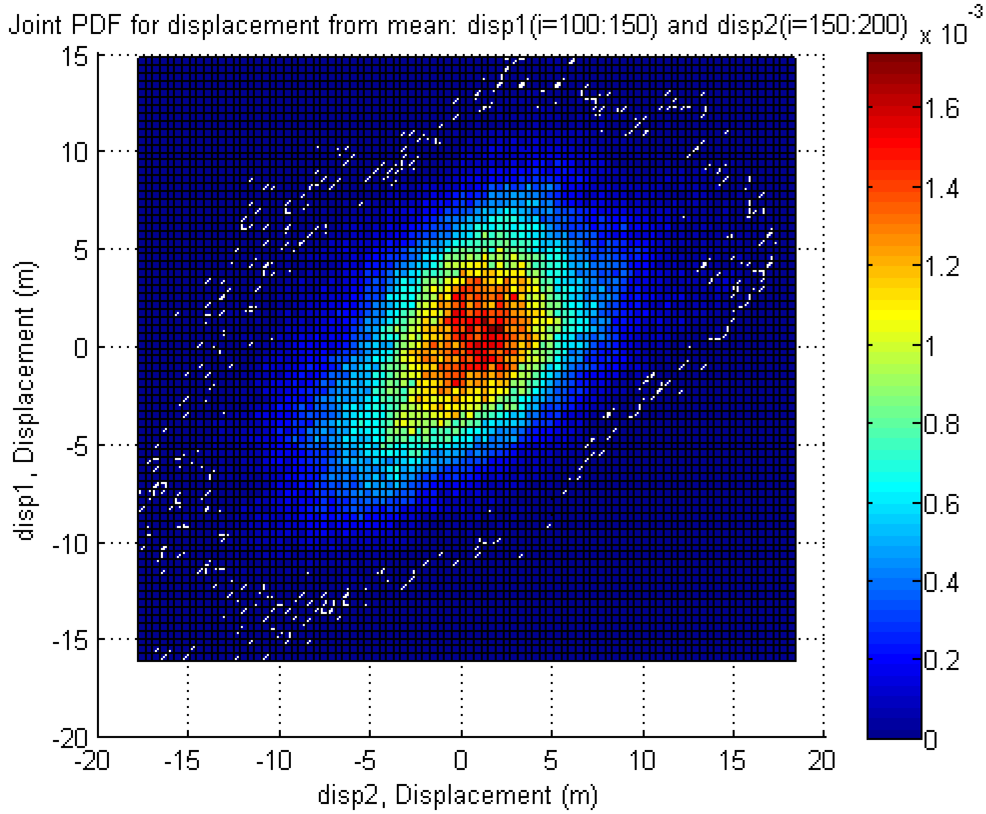

Isopycnal depth displacement from mean
Depth deviation from mean (m)
Depth of isopycnal (m)
Sample
60
40
20
0
-20
-40
-60
600
400
200
0
0
1000
2000
3000

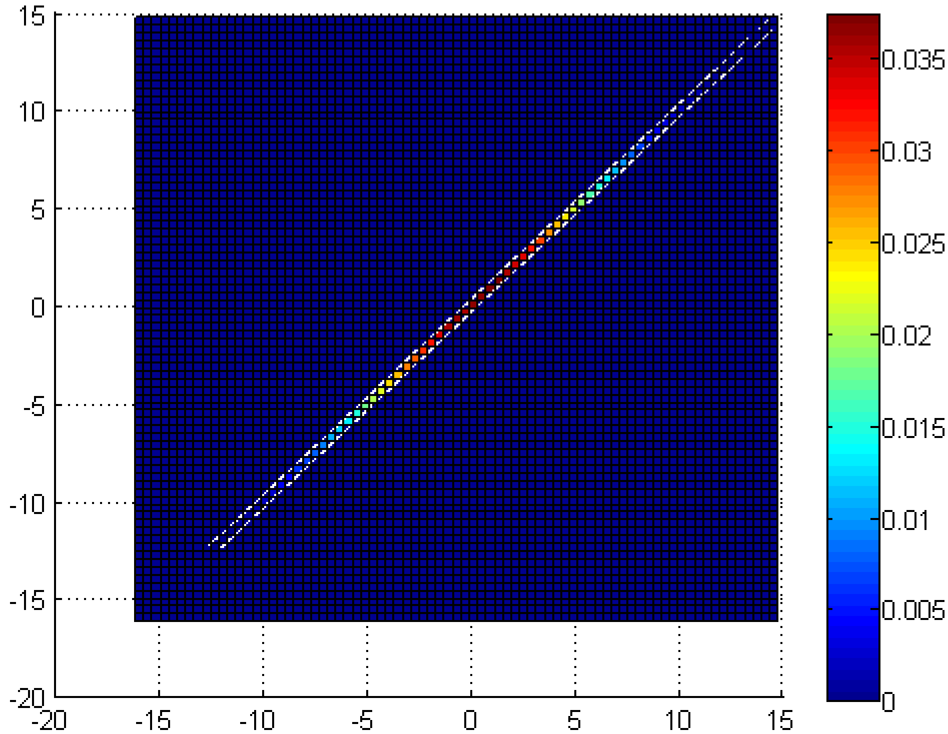

PDF of (selected) displacement data
Probability density (1/m)
Isopycnal Displacement from Mean (m)
0.1
0.09
0.08
0.07
0.06
0.05
0.04
0.03
0.02
0.01
0
-30
-20
-10
0
10
20
30
40


Joint PDF for displacement from mean: disp1(i=100:150) and disp2(i=110:160)
x 10^-3
Probability Desnsity (1/m^2)
disp1, Displacement (m)
disp2, Displacement (m)
4
3
2
1
0
20
10
0
-10
-20
-20
-10
0
10
20


Joint PDF for displacement from mean: disp1(i=100:150) and disp2(i=110:160)
disp1, Displacement (m)
disp2, Displacement (m)
x 10^-3
15
10
5
0
-5
-10
-15
-20
-20
-15
-10
-5
0
5
10
15
20
3
2.5
2
1.5
1
0.5


Joint PDF for displacement from mean: disp1(i=100:150) and disp2(i=150:200)
x 10^-3
Probability Desnsity (1/m^2)
2
1.5
1
0.5
0
20
10
0
-10
-20
-20
-10
0
10
20
disp1, Displacement (m)
disp2, Displacement (m)


Joint PDF for displacement from mean: disp1(i=100:150) and disp2(i=150:200)
x 10^-3
disp1 , Displacement (m)
disp2, Displacement (m)
15
10
5
0
-5
-10
-15
-20
-20
-15
-10
-5
0
5
10
15
20
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0


15
10
5
0
-5
-10
-15
-20
-20
-15
-10
-5
0
5
10
15
0.035
0.03
0.025
0.02
0.015
0.01
0.005
0

Published with MATLAB® 7.12