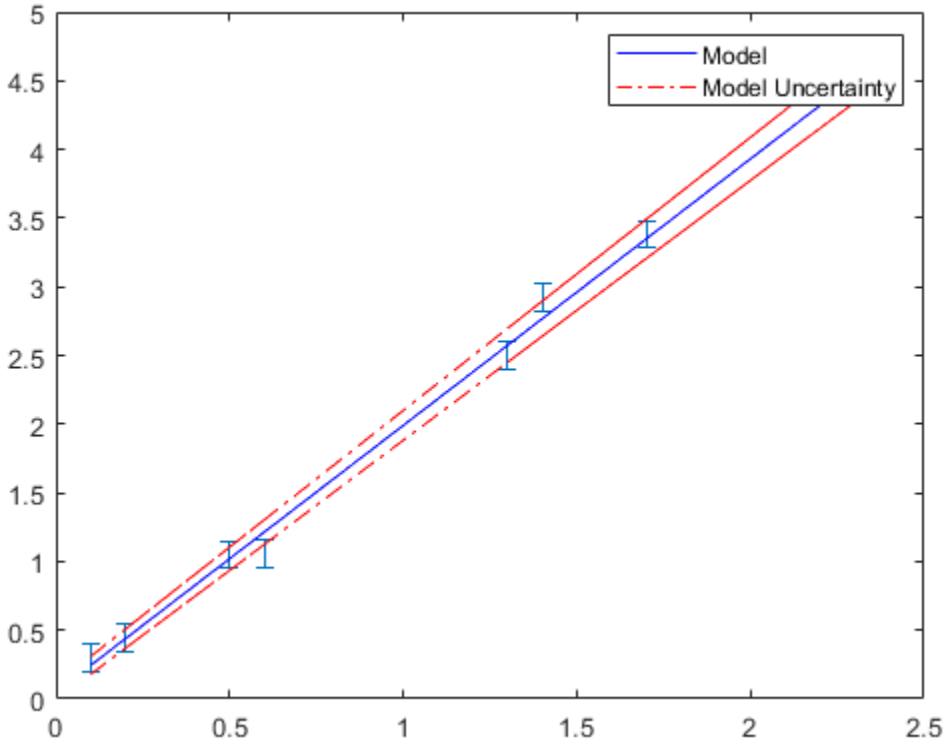
```
% Matlab Practical 10
% Zach Vig

clear;
close all;

x = [0.1; 0.6; 0.2; 0.5; 1.4; 2.3; 1.7; 1.3];
y = [0.3; 1.06; 0.44; 1.05; 2.92; 4.47; 3.38; 2.5];

%q1.1
figure(1);
scatter(x,y,'k+');

%q1.2
A = cat(2,x,ones(length(x),1));

%q1.3
Cd = diag([0.01,0.01,0.01,0.01,0.01,0.01,0.01,0.01]);
errorbar(x,y,sqrt(diag(Cd)),"LineStyle","none")

%q1.4
Cm_pri = 0;
m = (((A' * (Cd\A)) + Cm_pri) \ A') * (Cd \ y);
%{
    slope: 1.9405
    y-intercept: 0.0502
%}

Cm_post = inv(A' * (Cd\A) + Cm_pri);
%{
    Cm_post =
    0.0233   -0.0236
   -0.0236    0.036
%}

%q1.5
figure(1); hold on;
plot(x,A*m,"Color",'b');
m_sd = sqrt(diag(Cm_post));
plot(x,A*(m+m_sd),"LineStyle","-.","Color","r");
plot(x,A*(m-m_sd),"LineStyle","-.","Color","r");
legend('','Model','','Model Uncertainty');
hold off;

m_corr = Cm_post(1,2)/prod(sqrt(diag(Cm_post)));
%{
    The uncertainty in the slope and y-intercept are 0.0483 and 0.0603,
respectively. The off diagonal terms of the posterior model covariance
matrix are nonzero, so these parameters are correlated (i.e. they trade-off
with each other). The correlation coefficient here is -0.8276, which means
that the slope and y-intercept are strongly anti-correlated. This makes
sense because if you were to increase the y-intercept by a small ammount,
```

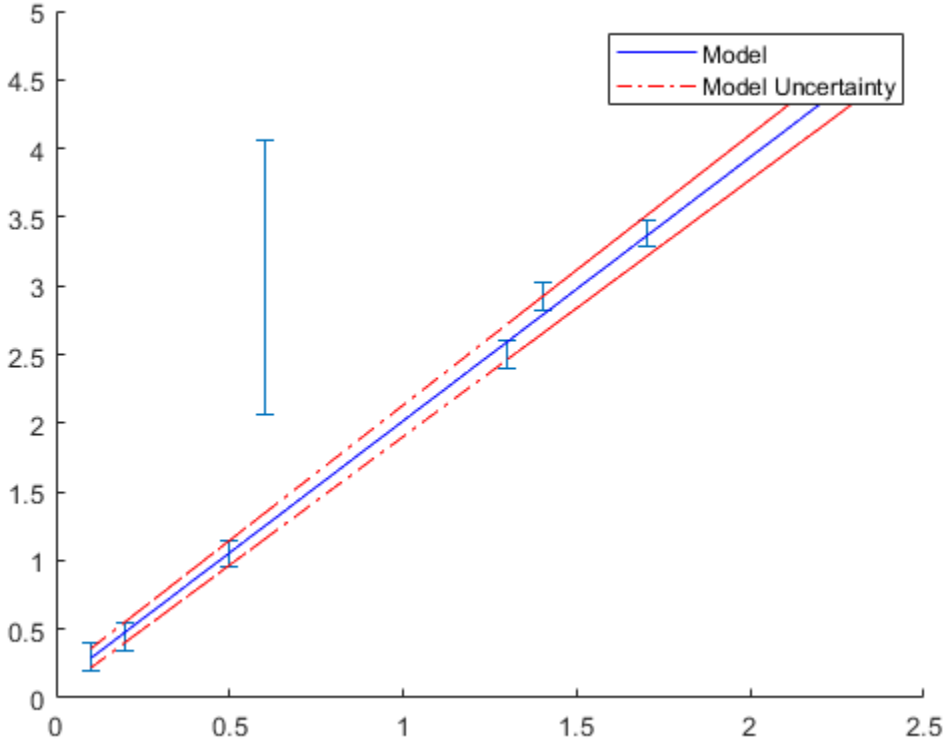
```
the slope would have to decrease drastically to stay fitted to the data.
%}

%q1.6
y = [0.3; 3.06; 0.44; 1.05; 2.92; 4.47; 3.38; 2.5];
Cd2 = diag([0.01,1,0.01,0.01,0.01,0.01,0.01,0.01]);
m2 = (((A' * (Cd2\A)) + Cm_pri) \ A') * (Cd2 \ y);
Cm_post2 = inv(A' * (Cd2\A) + Cm_pri);

figure(2); hold on;
errorbar(x,y,sqrt(diag(Cd2)),"LineStyle","none");
plot(x,A*m2,"Color",'b');
m_sd2 = sqrt(diag(Cm_post2));
plot(x,A*(m2+m_sd2),"LineStyle","-.","Color","r");
plot(x,A*(m2-m_sd2),"LineStyle","-.","Color","r");
legend('','Model','','Model Uncertainty');
hold off;
%{
    slope: 1.9207
    y-intercept: 0.0962
%}

%q1.7
m_corr2 = Cm_post2(1,2)/prod(sqrt(diag(Cm_post2)));
%{
    Cm_post2 =
    0.0024   -0.0026
   -0.0026    0.0042
   The uncertainty on the slope and the y-intercept are 0.049 and 0.0648,
respectively. The correlation coefficient this time is again very negative,
indicating an anti-correlation between the slope and y-intercept, with a
value of -0.8103. Overall, adding in an outlier with a large uncertainty
did not change the model by a significant amount. Through my own analysis,
however, I tested what the effect of an outlier with a small uncertainty
would be, and this changed the model significantly more.
%}

%q2.1
load('Data_for_Practical_10.mat');

figure(3);
imagesc(X(1,:),Y(:,1),Z-1,"CDataMapping",'scaled');
clim([-0.1,0.3]);
colorbar();

hold on;
plot(sta_x,sta_y,"rx");
for i = 1:size(putanje_x,1)
    p = plot(putanje_x(i,:),putanje_y(i,:),'r');
end
hold off;

%q2.2
Cd = 0.01*eye(length(d));
```

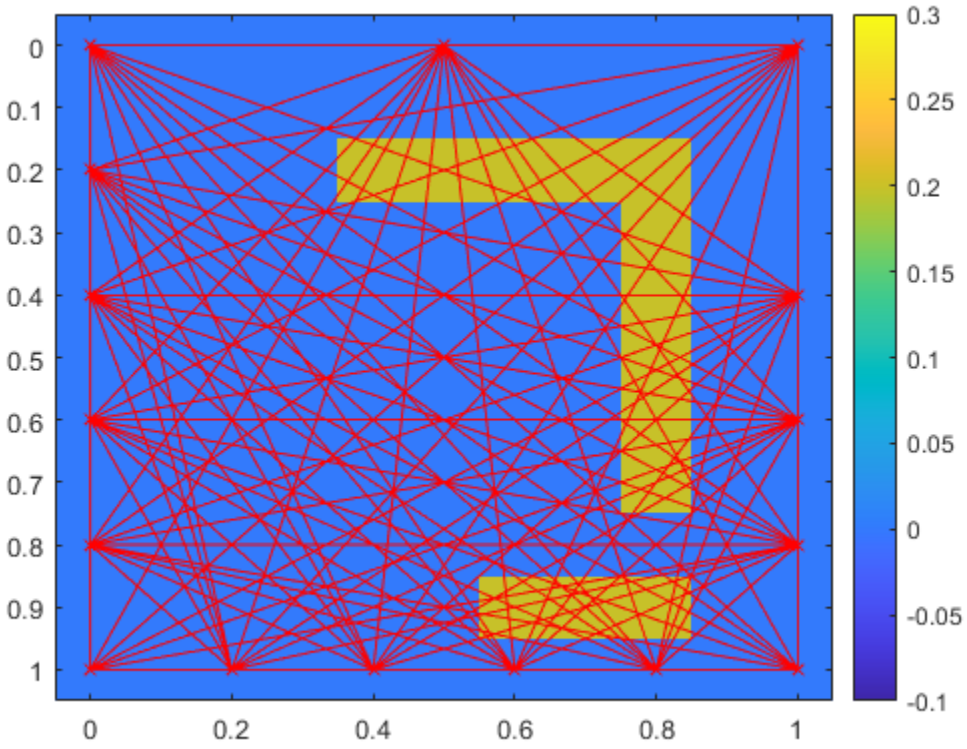
```
%q2.3
mu = 1;
Cm_pri = (mu^2) * eye(size(G,2));

%q2.4
figure(4);subplot(1,2,1);
m_est = (((G' * (Cd\G)) + Cm_pri) \ G') * (Cd \ d);
% heatmap(reshape(m_est,11,11));
image(X(1,:),Y(:,1),reshape(m_est,11,11),"CDataMapping","scaled");
clim([-0.1,0.3]);
colorbar();

hold on;
plot(sta_x,sta_y,"rx");
for i = 1:size(putanje_x,1)
    p = plot(putanje_x(i,:),putanje_y(i,:),'r');
end
hold off;
%{
    The modeled structure is certainly close to the actual structure, but
differs in a few key ways. The first is that on the right angle turn at
(0.8,0.2), the model struggles to reproduce the slowness values. This is
expressed in the model as having a larger spread of slowness values around
this area, and the slowness values that do appear are much lower than the
actual structure. The top piece of the structure, however, that occurs at
y=0.9 is predicted very well. One reason that the right angle turn may be
less resolved is that this area of the grid is relatively undetermined.
That is, there are only paths passing through this location coming from 2
stations.
%}

%q2.5
Cm_pri = (0.001^2)*eye(size(G,2));
subplot(1,2,2);
m_est2 = (((G' * (Cd\G)) + Cm_pri) \ G') * (Cd \ d);
% heatmap(reshape(m_est2,11,11));
image(X(1,:),Y(:,1),reshape(m_est2,11,11),"CDataMapping","scaled");
clim([-0.1,0.3]);
colorbar();
hold on;
plot(sta_x,sta_y,"rx");
for i = 1:size(putanje_x,1)
    p = plot(putanje_x(i,:),putanje_y(i,:),'r');
end
hold off;
%{
    The retrieved structure only differs slightly from the case where mu =
1. If I had to state a general change, it looks like some vaues are now
closer to the actual structure of uniform 0.2 slowness, but this is not
always the case. The two cases are likely so similar because the data is not
changing, only the prior information. Therefore, the model comes to roughly
the same conclusion, since it is seeing the same data.
%}
```

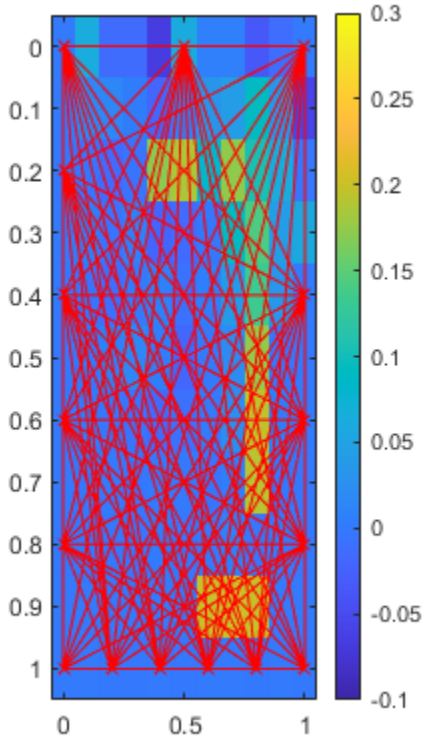
```
%q2.6
noise1 = 0.1*randn(size(d));
noise2 = randn(size(d));
d_noise1 = d+noise1;
d_noise2 = d+noise2;
Cm_pri = eye(size(G,2));

m_est_noise1 = (((G' * (Cd\G)) + Cm_pri) \ G') * (Cd \ d_noise1);
m_est_noise2 = (((G' * (Cd\G)) + Cm_pri) \ G') * (Cd \ d_noise2);

figure(5); subplot(1,2,1);
image(X(1,:),Y(:,1),reshape(m_est_noise1,11,11),"CDataMapping",'scaled');
title('SD = 0.1'); clim([-0.1,0.3]); colorbar();
hold on;
plot(sta_x,sta_y,"rx");
for i = 1:size(putanje_x,1)
    p = plot(putanje_x(i,:),putanje_y(i,:),'r');
end
hold off;

subplot(1,2,2);
image(X(1,:),Y(:,1),reshape(m_est_noise2,11,11),"CDataMapping",'scaled');
title('SD = 1'); clim([-0.1,0.3]); colorbar();
hold on;
plot(sta_x,sta_y,"rx");
for i = 1:size(putanje_x,1)
    p = plot(putanje_x(i,:),putanje_y(i,:),'r');
end
hold off;

%q2.7
%{
    The retrieved structure becomes fuzzier as noise is added to the
travel times. As the standard deviation approaches 1, the structure becomes
entirely hidden by the noise. THe first regions of the structure to become
affected by noise are areas that are relatively underdetermined by crossing
travel time paths. Areas that are more overdetermined, such as the line at
(0.7,0.9), can be retrieved even with a little bit of noise. Too much noise
still destroys the structure in these areas as well.
%}

%q2.8
Cm_pri = eye(size(G,2));
R = (((G' * (Cd\G)) + Cm_pri) \ G') * (Cd \ G);
%{
    Trace = 105.8134. This means that the resolution is not perfect (121),
but it is not terrible.
%}

%2.9
check1 = checkerboard(1,6,6) > 0.5;
check1 = reshape(check1(1:end-1,1:end-1),121,1);
```

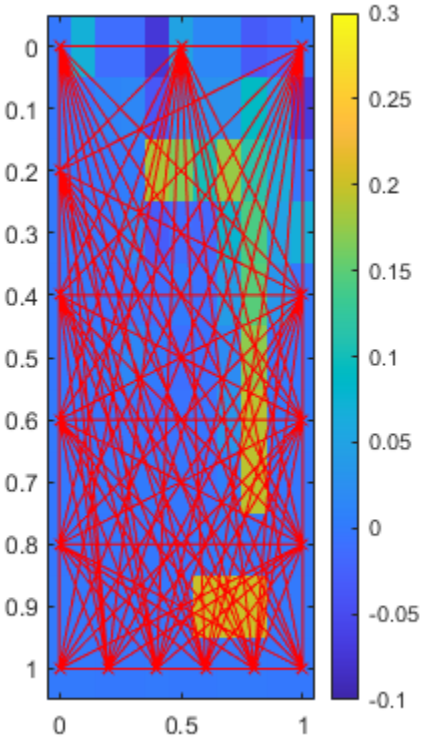
```
check2 = checkerboard(2,3,3) > 0.5;
check2 = reshape(check2(1:end-1,1:end-1),121,1);

test1 = R * check1;
test2 = R * check2;

figure(6);
subplot(2,2,1);imagesc(X(1,:),Y(:,1),reshape(check1,11,11));
subplot(2,2,2);imagesc(X(1,:),Y(:,1),reshape(check2,11,11));
subplot(2,2,3);imagesc(X(1,:),Y(:,1),reshape(test1,11,11));
hold on;
plot(sta_x,sta_y,"rx");
for i = 1:size(putanje_x,1)
    p = plot(putanje_x(i,:),putanje_y(i,:),'r');
end
hold off;
subplot(2,2,4);imagesc(X(1,:),Y(:,1),reshape(test2,11,11));
hold on;
plot(sta_x,sta_y,"rx");
for i = 1:size(putanje_x,1)
    p = plot(putanje_x(i,:),putanje_y(i,:),'r');
end
hold off;

%{
    Once again, the pattern of resolution follows the density of path
intersections. At the top of the image, where time delay paths are less
dense, the structure is poorly resolved, but toward the bottom of the image,
the structure is well resolved because there are many intersecting time
delay paths. This pattern is more apparent for the smaller checkerboard
size, but it also appears in the large checkerboard size.
%}
```

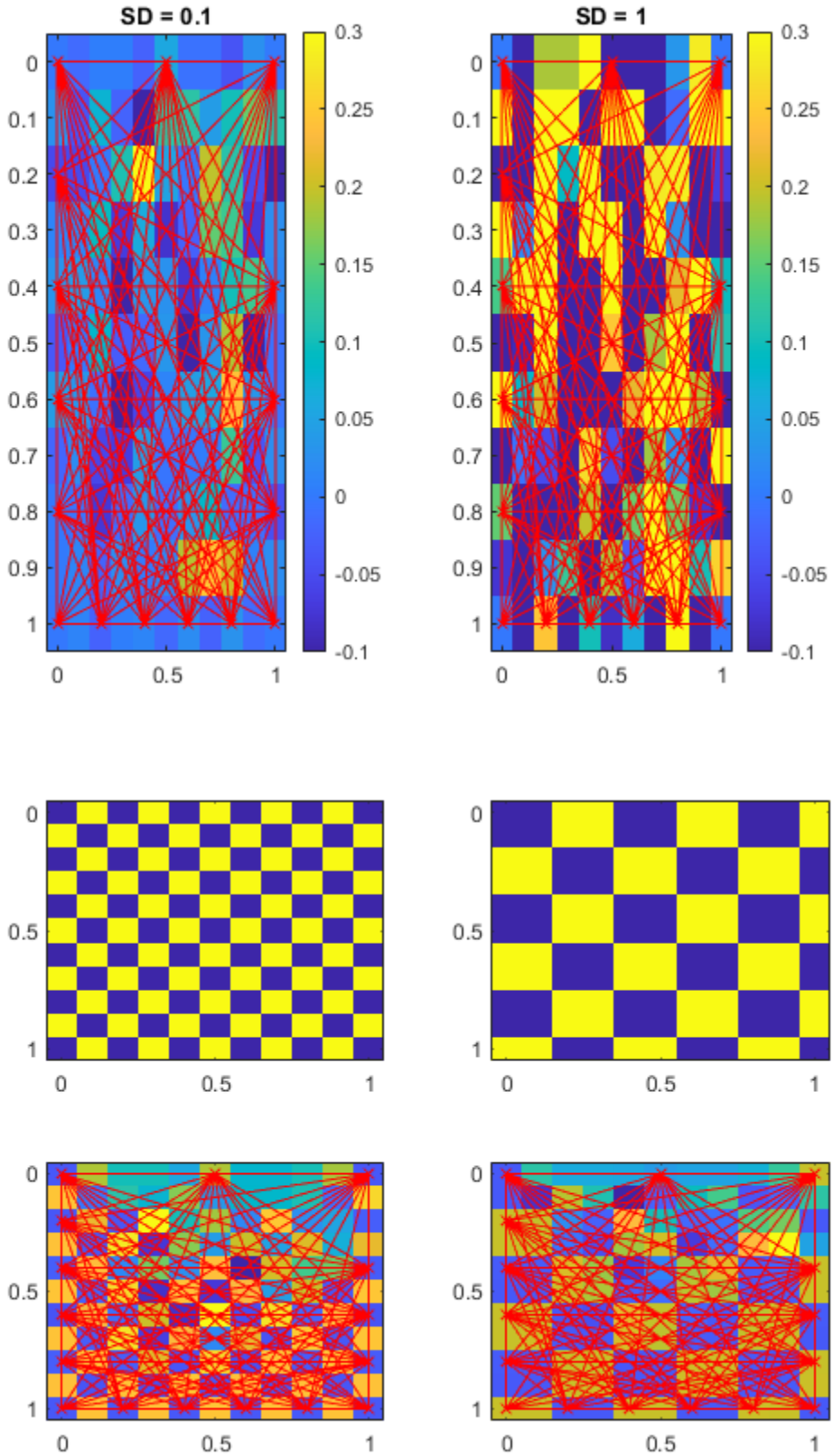

Model
Model Uncertainty
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
5


Model
Model Uncertainty
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
5


0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
0
0.2
0.4
0.6
0.8
1
0.3
0.25
0.2
0.15
0.1
0.05
0
-0.05
-0.1


0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
0
0.5
1
0.3
0.25
0.2
0.15
0.1
0.05
0
-0.05
-0.1


0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
0
0.5
1
0.3
0.25
0.2
0.15
0.1
0.05
0
-0.05
-0.1


SD = 0.1
SD = 1
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
0.5
0.3
0.25
0.2
0.15
0.1
0.05
-0.05
-0.1

Published with MATLAB® R2023b