Homework I

Question 1:

(a) we use the function in Matlab rand() to randomly assign masses, `m = rand(1,100)`

(b) The problem can be restated as a set of equations,

$$\begin{cases} m_1 = d_1 \\ m_1 + m_2 = d_2 \\ m_1 + m_2 + m_3 = d_3 \\ m_2 + m_3 + m_4 = d_4 \\ \cdots \\ m_{98} + m_{99} + m_{100} = d_{100} \end{cases}$$

Which can be rewritten as matrix forms, and the size is 100×100.

$$\begin{bmatrix} 1 & 0 & 0 & 0 & \cdots & 0 & 0 \\ 1 & 1 & 0 & \cdots & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & 1 & 1 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & 0 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} m_1 \\ m_2 \\ m_3 \\ m_4 \\ \cdots \\ m_{99} \\ m_{100} \end{bmatrix} = \begin{bmatrix} d_1 \\ d_2 \\ d_3 \\ d_4 \\ \cdots \\ d_{99} \\ d_{100} \end{bmatrix}$$

And G is,

$$\begin{bmatrix} 1 & 0 & 0 & 0 & \cdots & 0 & 0 \\ 1 & 1 & 0 & \cdots & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & 1 & 1 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & 0 & 1 & 1 & 1 \end{bmatrix}$$

(c) we use the function in Matlab to create observed data with Gaussian random numbers, `dobs = G1*m' + normrnd(0,sigmad,100,1)`

(d) Solve the inverse problem by simple least squares,

$$m_{est} = (G^T G)^{-1}(G^T d)$$

(e) The variance of the estimated model parameters can use the command, `var = std2(mest),` and different calculation will return different values.

(f) The counted number is 100 after calculating.

Question 2:

This method can be restated as follows,

$$\begin{cases} m_1 = d_1 \\ m_1 + m_2 = d_2 \\ m_1 + m_2 + m_3 = d_3 \\ m_1 + m_2 + m_3 + m_4 = d_4 \\ \dots \\ m_1 + \dots + m_{98} + m_{99} + m_{100} = d_{100} \end{cases}$$

The second weighing method will construct a different kernel G, which is quite easy, a lower triangular matrix.

$$G = \begin{bmatrix} 1 & 0 & \cdots & 0 & 0 \\ 1 & 1 & & 0 & 0 \\ \vdots & & \ddots & & \vdots \\ 1 & 1 & \cdots & 1 & 0 \\ 1 & 1 & & 1 & 1 \end{bmatrix}$$

The same as the Method 1, After calculating the variance of the estimated model parameters, we get the answer for (f), counted number is 100 .

(g) In fig.1 and fig.2 we can see the difference.

According to the two figures below, comparing the two ways to weigh a set of boxes, we can find the second way can be better, for the inverse value fits the initial value well.

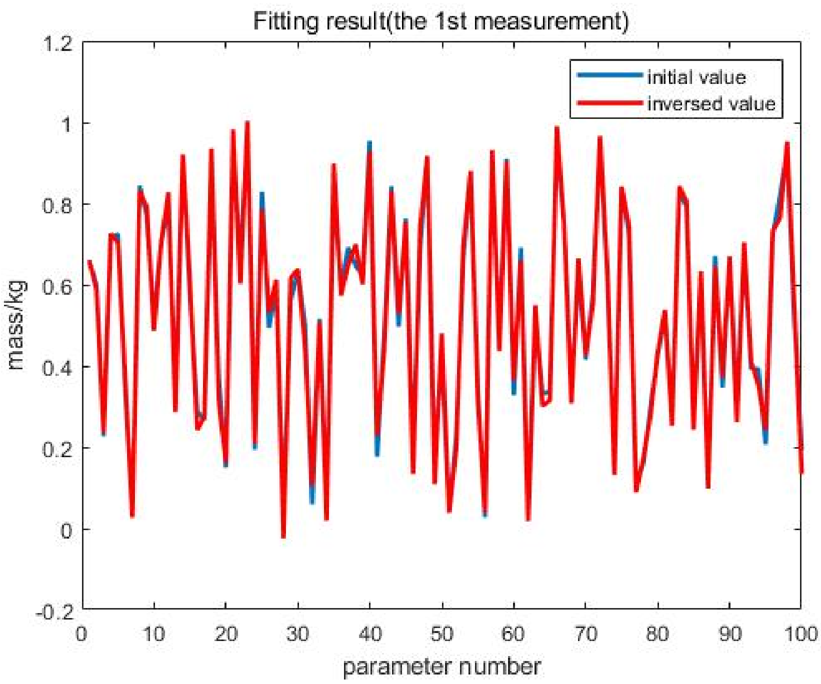

fig.1 first measurement result

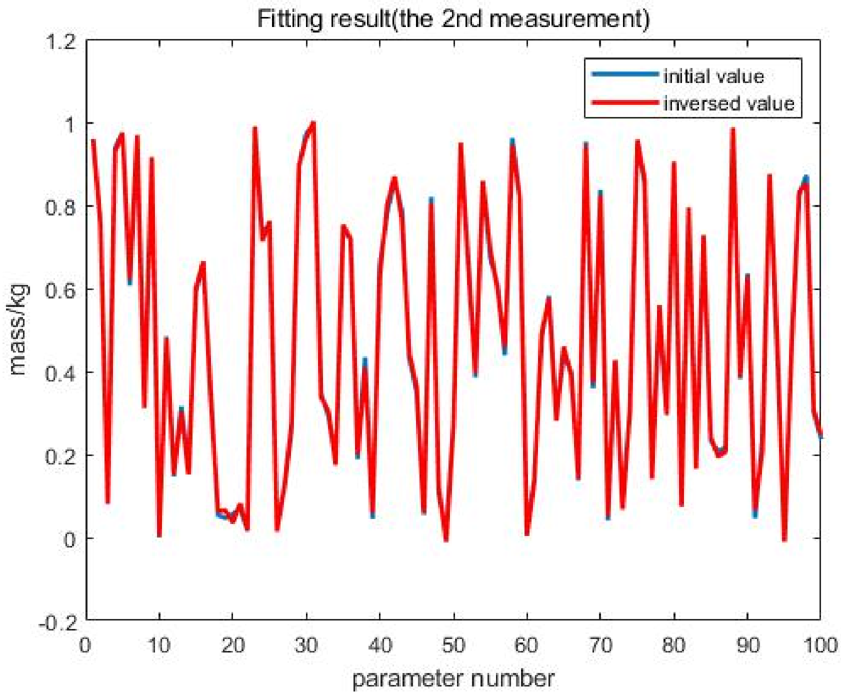

fig.2 second measurement result

Question 3:

Consider the cubic equation and answer the questions.

The problem can be restated as a set of equations as follows,

$$d_j = m_1 + m_2 z_j + m_3 {z_j}^2 + m_4 {z_j}^3, j = 1,2,3,\dots,11$$

which can be rewritten as the matrix form,

$$\begin{bmatrix} 1 & z_1 & {z_1}^2 & {z_1}^3 \\ 1 & z_2 & {z_2}^2 & {z_2}^3 \\ \cdots & \cdots & \cdots & \cdots \\ 1 & z_{11} & {z_{11}}^2 & {z_{11}}^3 \end{bmatrix} \begin{bmatrix} m_1 \\ m_2 \\ m_3 \\ m_4 \end{bmatrix} = \begin{bmatrix} d_1 \\ d_2 \\ \cdots \\ d_{11} \end{bmatrix}$$

$$\mathrm{G} = \begin{bmatrix} 1 & z_1 & {z_1}^2 & {z_1}^3 \\ 1 & z_2 & {z_2}^2 & {z_2}^3 \\ \cdots & \cdots & \cdots & \cdots \\ 1 & z_{11} & {z_{11}}^2 & {z_{11}}^3 \end{bmatrix}$$

We use the function in Matlab: `normrnd(0,0.05,11,1)` to create synthetic data with Gaussian random numbers with zero means and $\sigma_d = 0.05$.

The formula of simple least squares method is,

$$m_{est} = (G^T G)^{-1} (G^T d)$$

And the predicted data can be calculated by,

$$d_{pre} = G m_{est}$$

Finally, we get the figure below.

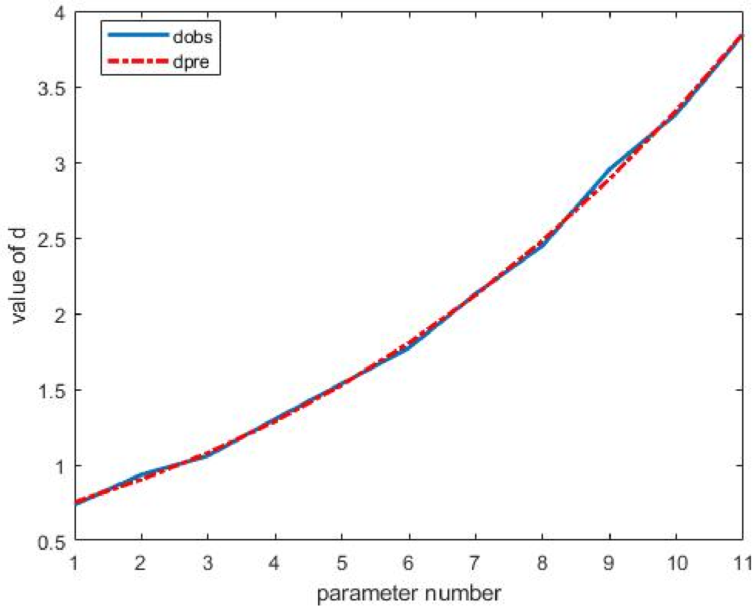

fig.3 plot in question 3(no constraint added)

Question 4:

Add a constraint that the predicted data pass through a fixed point, and go through the steps above in Question 3.

We can solve this problem only to reconstruct the kernel G, let the row 5 in G be zero. In this way, the fifth element in vector d can always be the value we want.

And the result(fig.4) is as follows,

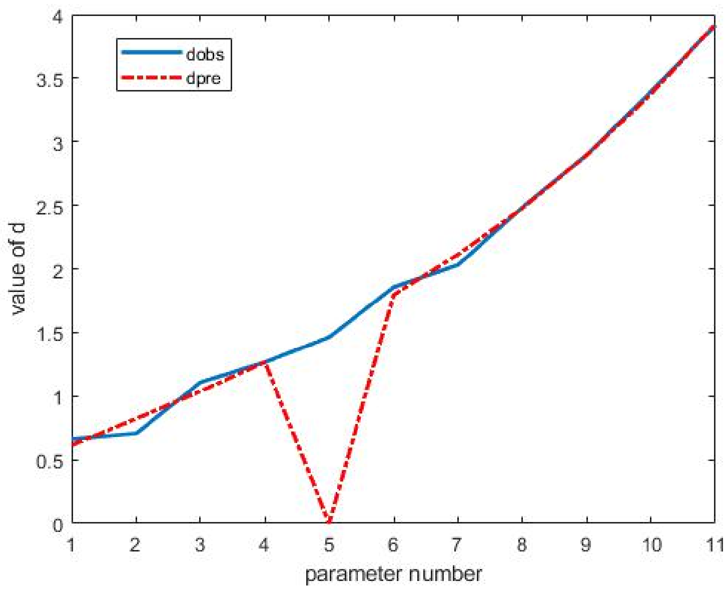

fig.4 plot in question 4(constraint added)

Programming code

（1）question1:

```
clear;
clc;

m = rand(1,100);%randomly assigns masses mtrue in the range of 0-1kg
% build the appropriate kernel G
% G is a 100*100 matrix
G1 = zeros(100,100);
G1(1,1) = 1;
G1(2,1) = 1;
G1(2,2) = 1;
for i = (3:100)
G1(i,i-2) = 1;
G1(i,i-1) = 1;
G1(i,i) = 1;
end
G = sparse(G1);
% create synthetic observed data by adding Gaussian random vector
sigmad = 0.01;
dobs = G1*m' + normrnd(0,sigmad,100,1);
%solve the inverse problem by simple least squares
mest = (G'*G)\(G'*dobs);
% to calculate the variance of each of the estimated model parameters
var = std2(mest);

sigmam = sqrt(var);
count = 0;
for j = (1:length(mest))
a = mest(j);
b = m(j);
if abs(a-b) <= 2*sigmam
count = count + 1;
end
end
fprintf("the number of estimated model parameters that are within 2σ of their
true value is %d\n",count)

%draw the picture
x = [1:100];
clf;
plot(x,m,x,mest,'r-','linewidth',2),title('Fitting result(the 1st
measurement)'),xlabel('parameter number'),ylabel('mass/kg'),legend('initial
value','inversed value')
```

（2）question 2:

```
clear;
clc;

m = rand(1,100);%randomly assigns masses mtrue in the range of 0-1kg
% build the appropriate kernel G
% G is a 100*100 matrix
G1 = tril(ones(100));
G = sparse(G1);
% create synthetic observed data by adding Gaussian random vector
sigmad = 0.01;
dobs = G1*m' + normrnd(0,sigmad,100,1);
%solve the inverse problem by simple least squares
mest = (G'*G)\(G'*dobs);
% to calculate the variance of each of the estimated model parameters
var = std2(mest);

sigmam = sqrt(var);
count = 0;
for j = (1:length(mest))
a = mest(j);
b = m(j);
if abs(a-b) <= 2*sigmam
count = count + 1;
end
end
fprintf("the number of estimated model parameters that are within 2σ of their true value is %d\n",count)

%draw the picture
figure(2)
x = [1:100];
clf;
plot(x,m,x,mest,'r-','linewidth',2),title('Fitting result(the 2nd measurement)'),xlabel('parameter number'),ylabel('mass/kg'),legend('initial value','inversed value')
```

(3) question 3:

```
clear;
clc;
%construct a vector z with 11 elements equally spaced 0.1
z = [1:0.1:2];
%randomly assigns elements of mtrue
m = [0.1,0.3,-0.2,0.5]';
```

```
%build the appropriate kernel G
G = zeros(11,4);
G(:,1) = ones(1,11)';
G(:,2) = z';
G(:,3) = z'.^2;
G(:,4) = z'.^3;

%create synthetic data with Gaussian random numbers
mean = 0;
sigmad = 0.05;
dobs = G*m + normrnd(mean,sigmad,11,1);

%solve the inverse problem by simple least squares
mest = (G'*G)\(G'*dobs);

dpre = G*mest;
figure(3)
clf;
x = [1:11]';
plot(x,dobs,x,dpre,'r-.','linewidth',2),legend('dobs','dpre')
```

(4) question 4:

```
clear;
clc;
%construct a vector z with 11 elements equally spaced 0.1
z = [1:0.1:2];
%randomly assigns elements of mtrue
m = [0.1,0.3,-0.2,0.5]';

%build the appropriate kernel G
G = zeros(11,4);
G(:,1) = ones(1,11)';
G(:,2) = z';
G(:,3) = z'.^2;
G(:,4) = z'.^3;

%create synthetic data with Gaussian random numbers
mean = 0;
sigmad = 0.05;
dobs = G*m + normrnd(mean,sigmad,11,1);

%solve the inverse problem by simple least squares
%add the prior constriants
```

```
G(5,:) = zeros(1,4);
mest = (G'*G)\(G'*dobs);

dpre = G*mest;
figure(3)
clf;
x = [1:11]';
plot(x,dobs,x,dpre,'r-.','linewidth',2),legend('dobs','dpre')
```