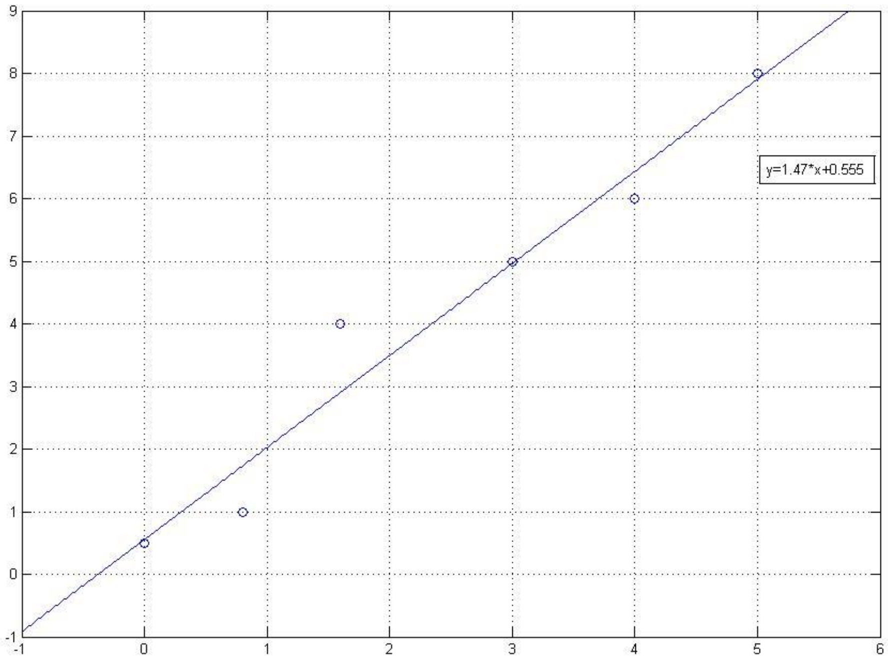
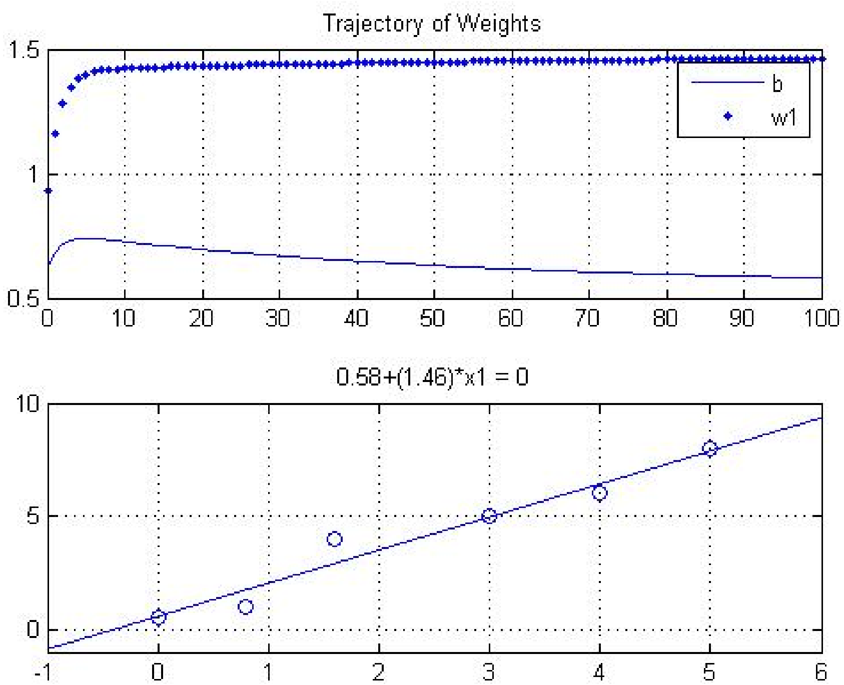
Q4.

1) The fitting result of standard linear least-squares (LLS)

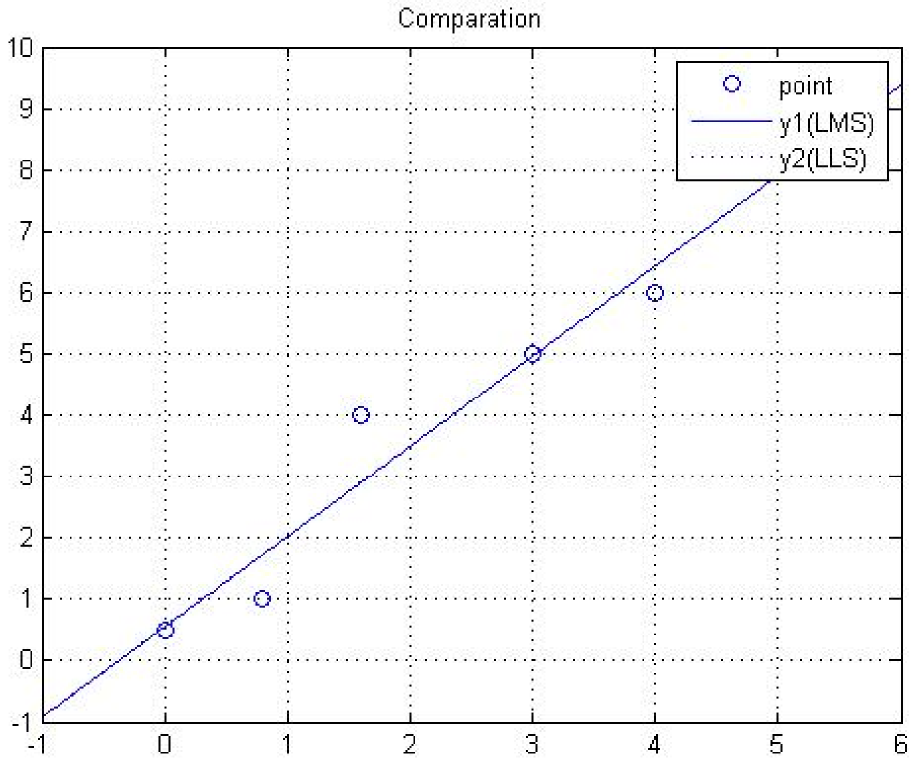


2) The fitting result of least-mean-square (LMS) algorithm and the trajectories of the weight versus learning steps.



As shown in the figure, the weights will converge after 100 epochs

3) The result of comparing LLS and LMS methods

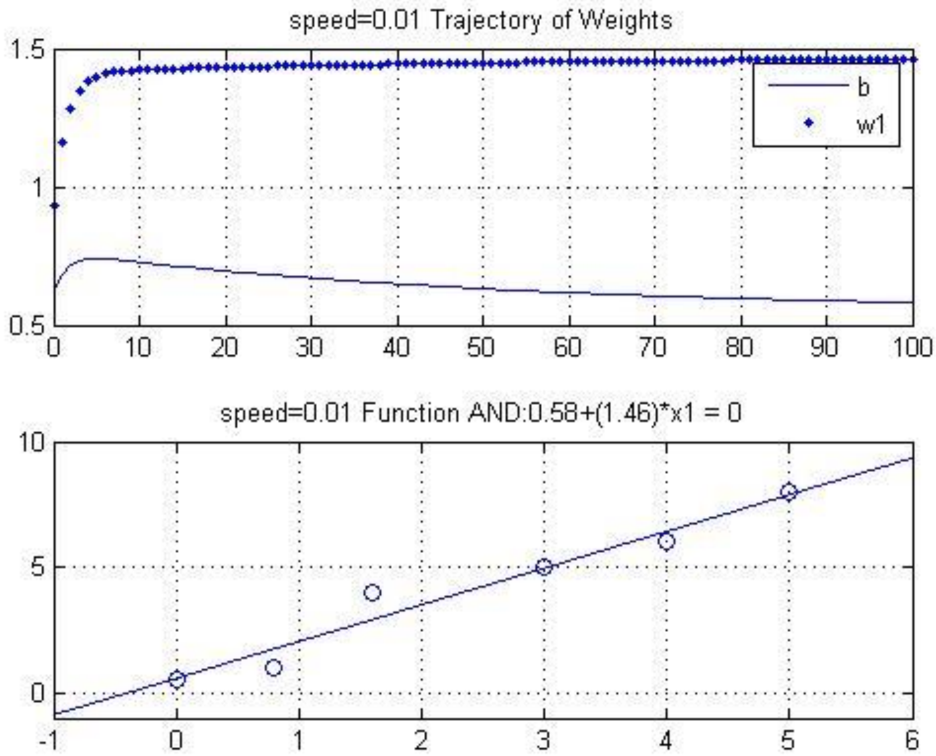


From the figure, there are almost no difference between the LMS and LLS. The result from the LMS could show that the method LLS could get the solution precisely and quickly.

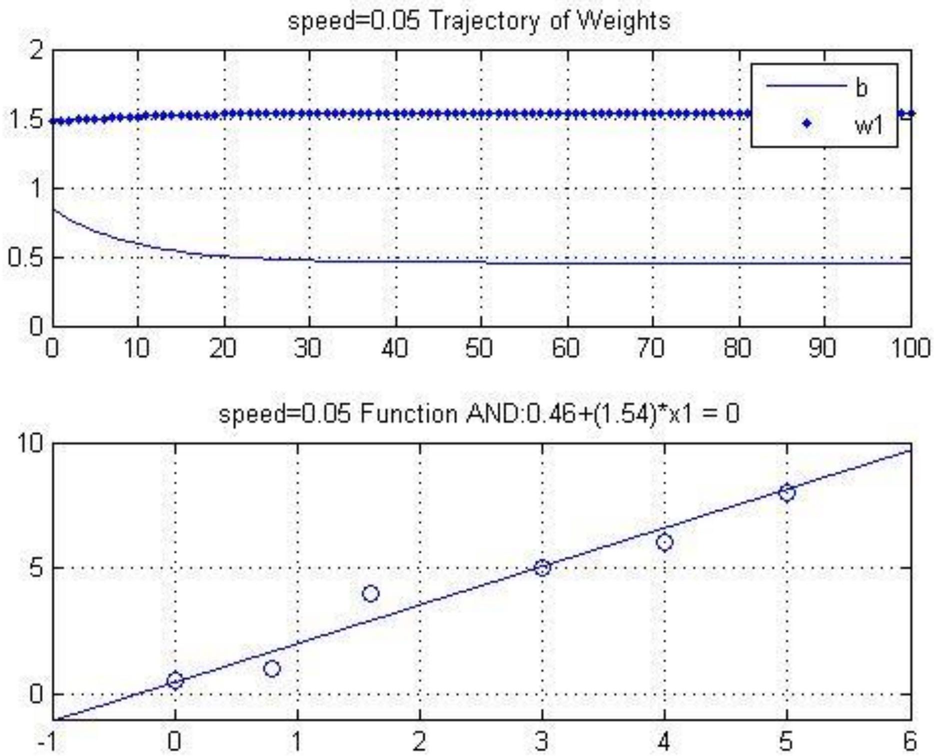
LMS: y1=1.46x+0.580

LLS: y2=1.47x+0.555

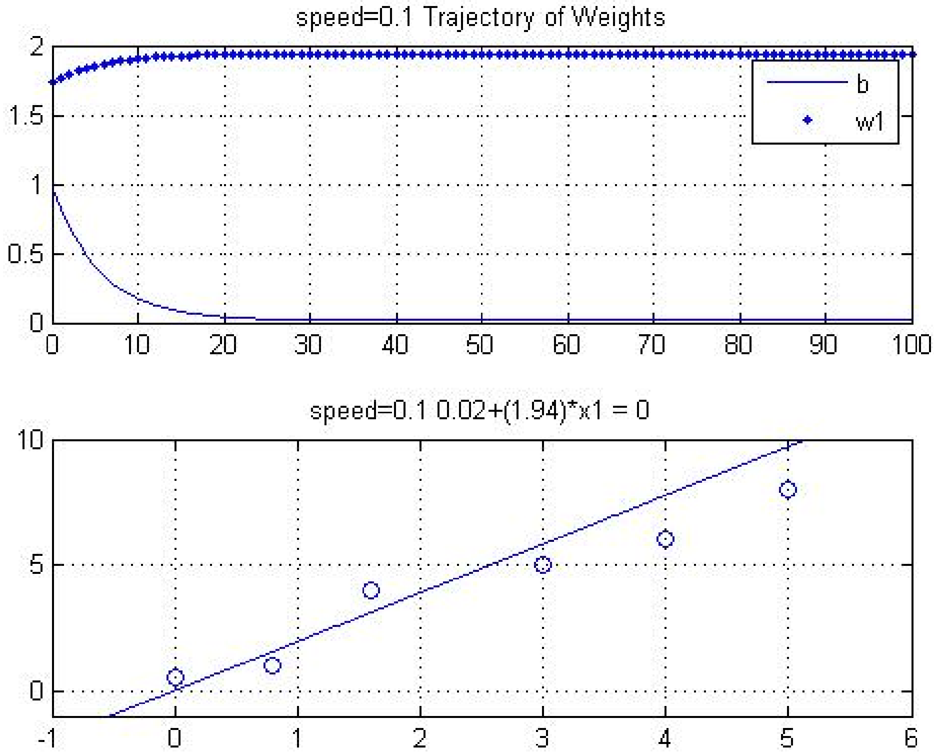
4) With different learning rates:0.01 0.05 0.10 0.20, the results are shown below
   Learning rates: 0.01



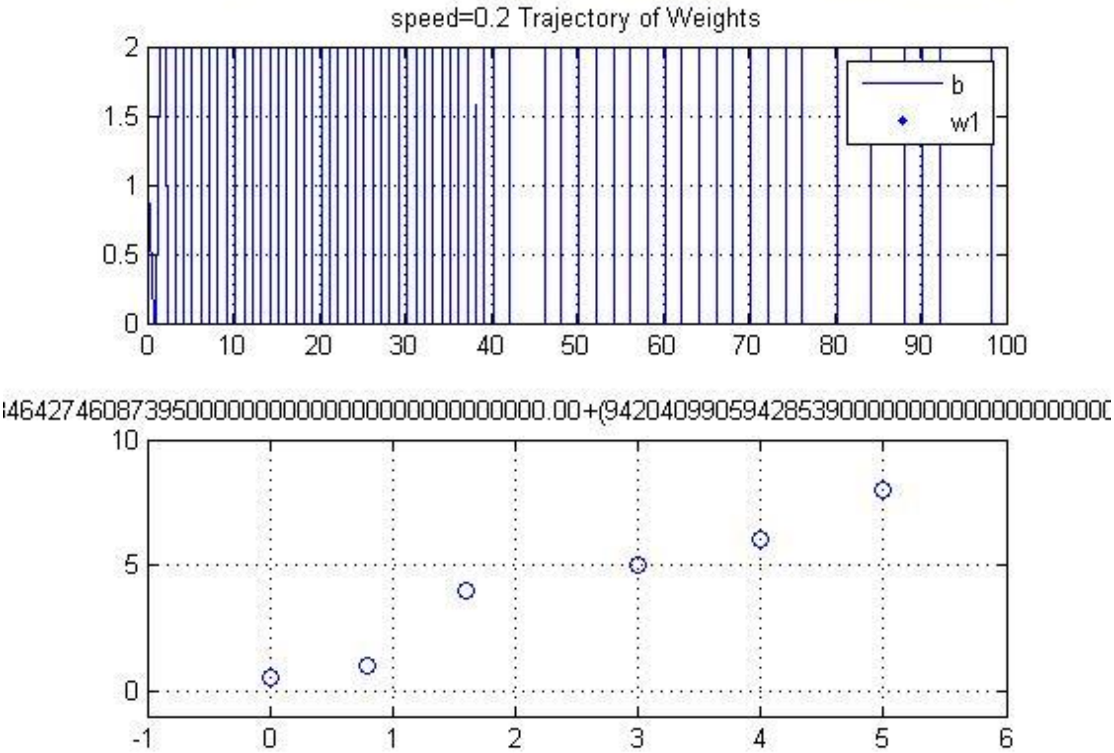
Learning rates: 0.05



Learning rates: 0.10



Learning rates: 0.20



During the repeating the simulation study in b) with different learning rates, it's obviously that the weights converge faster with the increase of rate. What's more, when the learning rate becomes 0.1, the result changes a lot, and the weight also converge in the end, which means that this result also fulfill the requirement.
But when we choose the learning rate to 0.20, the weight could not converge. The main reason is that the LMS algorithm rely on the Taylor series which is useful for a small learning rate. Because of that, when the learning exceed the limiting of Taylor series, it will not converge in the end.

The following is the MATLAB code of Q4:

```
%% INPUT
epoch = 100;
X = [1 0.0
     1 0.8
     1 1.6
     1 3.0
     1 4.0
     1 5.0];
d = [0.5 1 4 5 6 8]';
W = [0.5;0.5];
speed = 0.01;


%% MAIN CODE
for n = 0 : epoch
    for i = 1:length(d)
        e = d(i) - X(i,:) * W;
        W = W + e * speed * X(i,:)';
        w1(n+1) = W(1);
        w2(n+1) = W(2);
    end
end
%% OUTPUT
figure;
subplot(2,1,1);
x = 0:epoch;
plot(x,w1,'-');
grid on;
hold on;
plot(x,w2,'.');
hold on;
axis([0,100,0.5,1.5]);
legend('b','w1');
title('Trajectory of Weights');
hold off;


subplot(2,1,2);
plot([0 0.8 1.6 3 4 5],[0.5 1 4 5 6 8],'o');
grid on;
hold on;
axis([-1,6,-1,10]);
```

```
s = sprintf('%.2f+(%.2f)*x1 = 0',W(1),W(2));
title(s);
x = -1:.01:6;
y = x*W(2)+W(1);
plot(x,y);
```