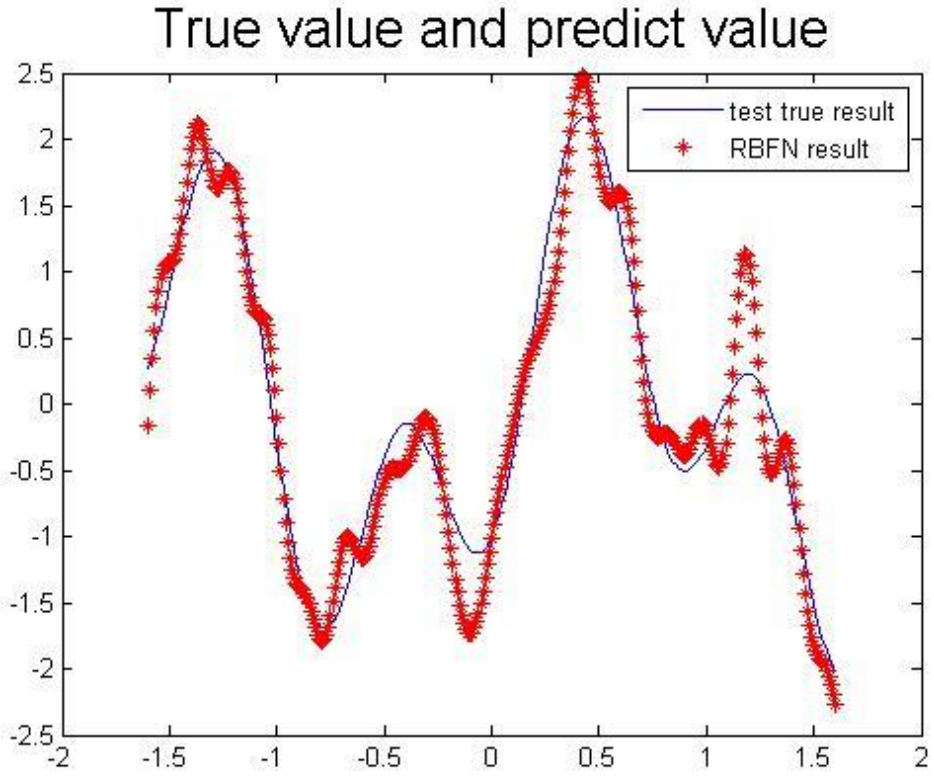
Q1.

a) Use the exact interpolation method (as described on pages 16-21 in the slides of lecture five) and determine the weights of the RBFN. Assume the RBF is Gaussian function with standard deviation of 0.1. Evaluate the approximation performance of the resulting RBFN using the test set.

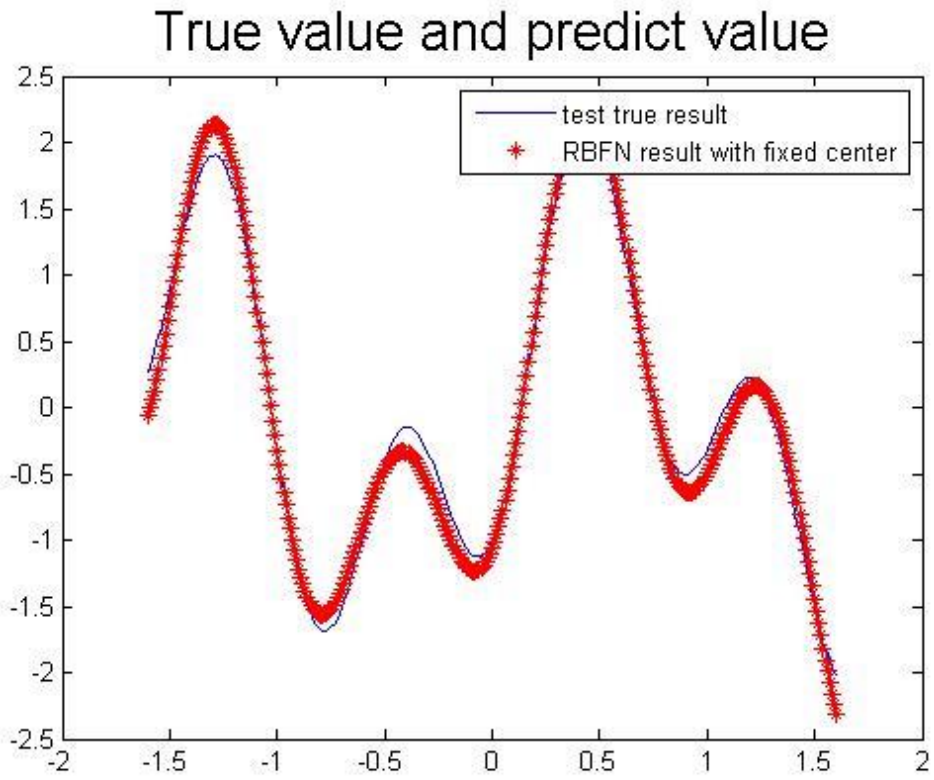
As shown in the figure below, there exist the difference between the test true value and the RBFN result since the random noise affects the weights and the path of training set is quite smaller than the path of test set. It is overfitting.

From the figure below, the approximation performance of the resulting RBFN is quite well. The variance is 0.0820 which is quite well for approximation.



b) Follow the strategy of "Fixed Centers Selected at Random" (as described on page 37 in the slides of lecture five), randomly select 20 centers among the sampling points. Determine the weights of the RBFN. Evaluate the approximation performance of the resulting RBFN using test set. Compare it to the result of part a).

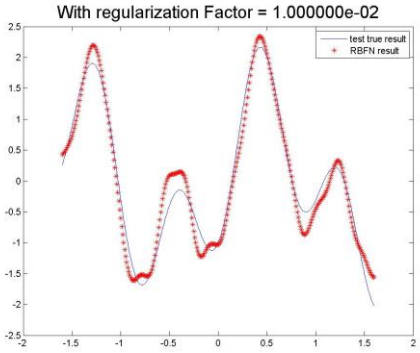
From the figure below, the approximation performance of the resulting RBFN is quite well. The variance is 0.0184 which is quite well for approximation, which is smaller than that in a). So with the help of fixed centers selected at random, the approximation performance is better.

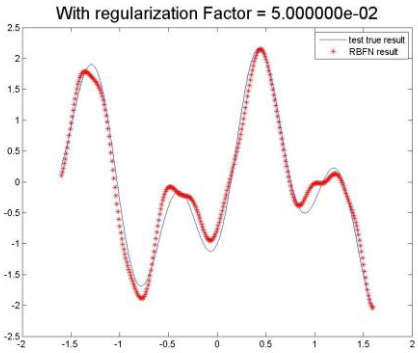


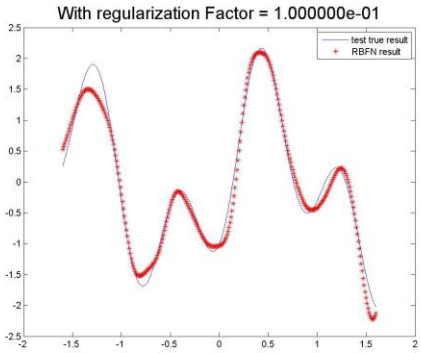
c) Use the same centers and widths as those determined in part a) and apply the regularization method as described on pages 42-45 in the slides for lecture five. Vary the value of the regularization factor and study its effect on the performance of RBFN.

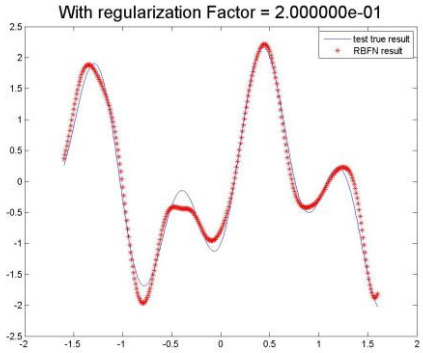
When the factor of regularization is small, the effect of it on the performance of RBFN is quite little. With the increasing of factor, the variance decrease to 0.0257 when the factor is 0.2. After that, the variance increase and the figure is under fitting. The effect of regularization is smaller than the fixed centers.

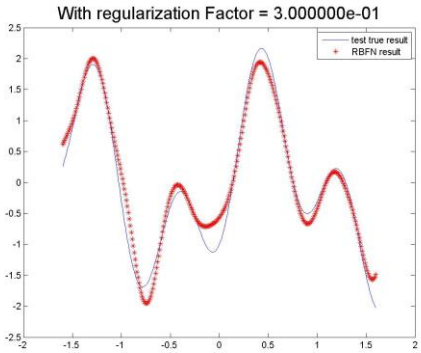
Factor	0.01	0.05	0.1	0.2	0.3	0.4	0.5
variance	0.0489	0.0372	0.0311	0.0257	0.0424	0.0405	0.437
Factor	0.7	0.9	1.0	1.5	2.0	5.0	10.0
variance	0.0344	0.0611	0.0557	0.0577	0.0982	0.1967	0.3948

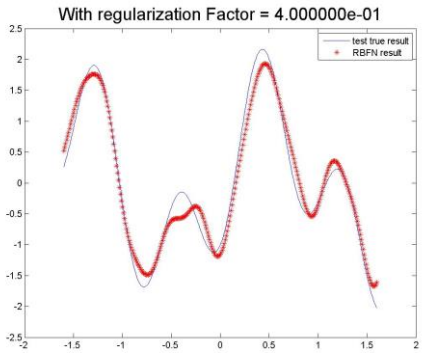




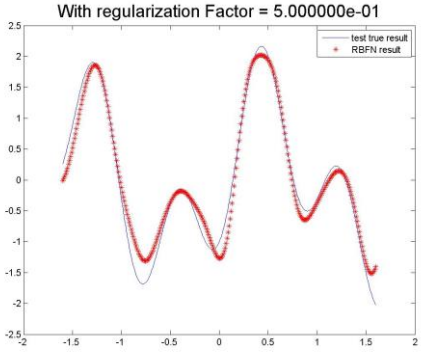




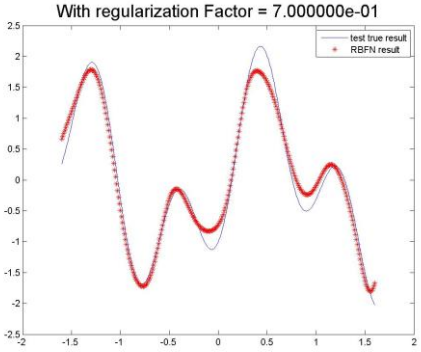




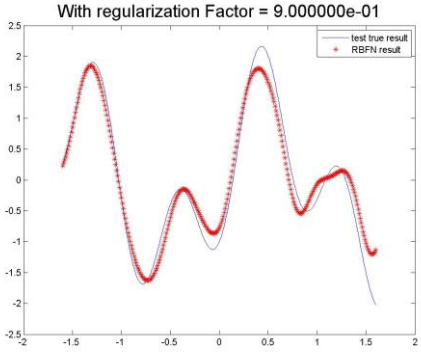
With regularization Factor = 5.000000e-01



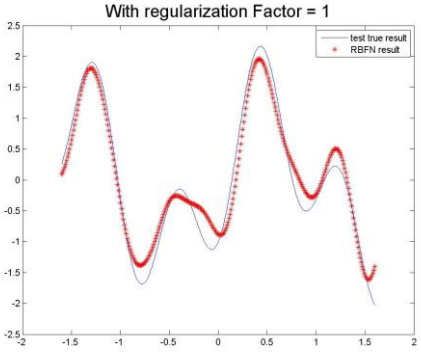
With regularization Factor = 7.000000e-01



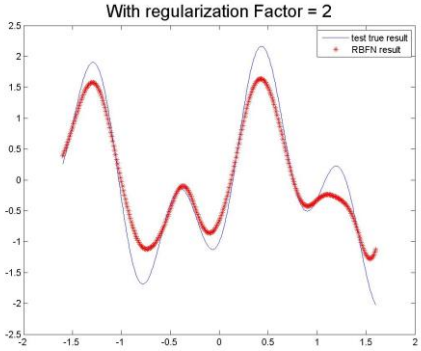
With regularization Factor = 9.000000e-01



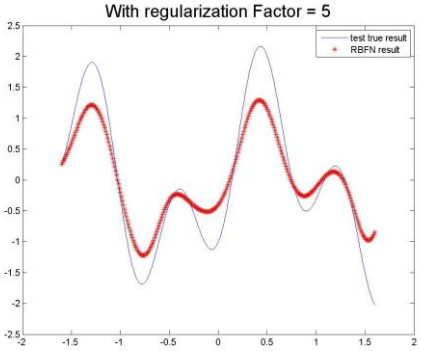
With regularization Factor = 1



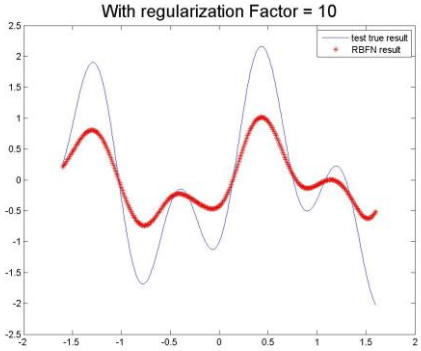
With regularization Factor = 2



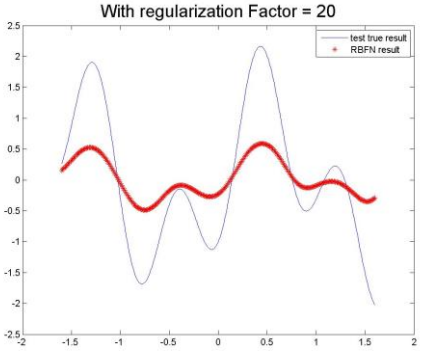
With regularization Factor = 5



With regularization Factor = 10



With regularization Factor = 20



The following is the MATLAB code of Q1_1:

```
%%
%INPUT
clear all;
close all;
x = -1.6:0.08:1.6;%training set
x_test = -1.6:0.01:1.6;%test set
random_noise = randn(size(x)) ;
y = 1.2*sin(pi*x)-cos(2.4*pi*x)+0.3*random_noise;%target values of
training set
y_test = 1.2*sin(pi*x_test)-cos(2.4*pi*x_test);%true value of test
set
%%
%CACULATION
function_r = exp(-(dist(x)).^2/0.02);%Gaussian Functions
w = inv(function_r)*y'; %Weight matrix

function_r_test = exp(-(dist(x_test',x)).^2/0.02);%Gaussian Functions
y_test_out = (function_r_test*w)';

evaluate = sum((y_test-y_test_out).^2)/length(x_test);%Evaluate the
performance
%%
%OUTPUT
figure;
plot(x_test,y_test,'b-');
hold on;
plot(x_test,y_test_out,'r*');
legend('test true result','RBFN result');
title('True value and predict value','FontSize',20);
```

The following is the MATLAB code of Q1_2:

```
%%
%INPUT
clear all;
close all;
x = -1.6:0.08:1.6;%training set
x_test = -1.6:0.01:1.6;%test set
random_noise = randn(size(x)) ;
y = 1.2*sin(pi*x)-cos(2.4*pi*x)+0.3*random_noise;%target values of
training set
y_test = 1.2*sin(pi*x_test)-cos(2.4*pi*x_test);%true value of test
set
random = x(randperm(length(x)));
random_20 = random(1:20);%fixed centres selected at random
dm = max(max(dist(random_20',random_20)));
%%
%CACULATION
function_r = exp(-
length(random_20)/(dm^2)*(dist(x',random_20)).^2);%Gaussian Functions
w = pinv(function_r'*function_r)*function_r'*y'; %Weight matrix

function_r_test = exp(-
length(random_20)/(dm^2)*(dist(x_test',random_20)).^2);%Gaussian
Functions
y_test_out = (function_r_test*w)';

evaluate = sum((y_test-y_test_out).^2)/length(x_test);%Evaluate the
performance
%%
%OUTPUT
figure;
plot(x_test,y_test,'b-');
hold on;
plot(x_test,y_test_out,'r*');
legend('test true result','RBFN result with fixed center');
title('True value and predict value','FontSize',20);
```

The following is the MATLAB code of Q1_3:

```
%%
%INPUT
clear all;
close all;
x = -1.6:0.08:1.6;%training set
x_test = -1.6:0.01:1.6;%test set
random_noise = randn(size(x)) ;
y = 1.2*sin(pi*x)-cos(2.4*pi*x)+0.3*random_noise;%target values of
training set
y_test = 1.2*sin(pi*x_test)-cos(2.4*pi*x_test);%true value of test
set
i = 10;%the value of the regularization factor
%%
%CACULATION
function_r = exp(-(dist(x)).^2/0.02);%Gaussian Functions
w =
inv(function_r'*function_r+i*eye(size(function_r)))*function_r'*y'; %
Weight matrix

function_r_test = exp(-(dist(x_test',x)).^2/0.02);%Gaussian Functions
y_test_out = (function_r_test*w)';

evaluate = sum((y_test-y_test_out).^2)/length(x_test);%Evaluate the
performance
%%
%OUTPUT
figure;
plot(x_test,y_test,'b-');
hold on;
plot(x_test,y_test_out,'r*');
legend('test true result','RBFN result');
hold off;
discription1 = sprintf('With regularization Factor = %d',i);
title(discription1,'FontSize',20);
discription2 = sprintf('Q1_3_%d.jpg',i);
saveas(gcf,discription2);
close;
```