HW5 ECE542

Ming Dai
unityID: mdai3

March 26, 2017

Q1

HW5

Q1. let $\vec{X}_2 = (X_{2,1}, X_{22}\ X_{23})^T$

$\vec{X}_3 = (X_{31}, X_{32}, X_{33})^T$

We have $X_1^T A X_2 = 0$

$$(1,0,0)\begin{pmatrix} 1.0 & 0.5 & 0.1 \\ 0.5 & 1.0 & 0.5 \\ 0.1 & 0.5 & 1.0 \end{pmatrix}\begin{pmatrix} x_{2,1} \\ x_{22} \\ x_{23} \end{pmatrix} = 0$$

$\therefore\ x_{21} + 0.5 x_{22} + 0.1 x_{23} = 0$

Therefore, one possible value for $\vec{X}_2$ is

$\vec{X}_2 = (-1, 2, 0)^T$

For $\vec{X}_3$, We have

$$\begin{cases} \vec{X}_1^T A \vec{X}_3 = 0 \\ \vec{X}_2^T A \vec{X}_3 = 0 \end{cases}$$

$$\therefore \begin{cases} x_{31} + 0.5 x_{32} + 0.1 x_{33} = 0 \\ 1.5 x_{32} + 0.9 x_{33} = 0 \end{cases}$$

$\therefore\ \vec{X}_3 = (1, -3, 5)^T$

$\therefore$ One solution for A-conjugate given $\vec{x}_1 = (1,0,0)^T$

is

$$\begin{cases} \vec{x}_2 = (-1, 2, 0)^T \\ \vec{x}_3 = (1, -3, 5)^T \end{cases}$$

Figure 1: Q1

Q2

data set setting:
dist = 5.0, width = 6, radius =10
train samples = 2000, test samples = 2000

experiments

- training method
  - gradient descent
  - conjugate descent
  - Levenverg-Marquart
- hidden neuron numbers
  - 5
  - 20

Result

- testing error
- training time (both epochs and real clock time)
- Repeat measure time = 5
- convergence criterion

MSE	Epochs	training time	convergence reason
parameters	hidden numbers = 5	trainFcn = trainlm	
4.152e-07	10	0.484s	performance goal reached
3.214e-07	6	0.141s	performance gaol reached
3.196e-07	11	0.188s	performance goal reached
8.428e-08	11	0.188s	performance goal reached
4.041e-07	9	0.141s	performance gaol reached
parameters	hidden numbers = 5	trainFcn = traingd	
1.069e-03	10000	16.328s	maximum epoch reached
1.194e-03	10000	15.484s	maximum epoch reached
9.472e-04	10000	15.922s	maximum epoch reached
1.074e-03	10000	16.266s	maximum epoch reached
1.337e-03	10000	16.172s	maximum epoch reached
parameters	hidden numbers = 5	trainFcn = traincgf	
7.757e-05	110	0.563s	minimum step size reached
1.113e-04	109	0.531s	minimum step size reached
1.515e-04	89	0.531s	minimum step size reached
2.348e-05	232	1.094s	minimum step size reached
5.258e-05	85	0.500s	minimum step size reached

Table 1: hidden number =5

MSE	Epochs	training time	convergence reason
parameters	hidden numbers = 20	trainFcn = trainlm	
6.104e-07	33	0.500s	performance goal reached
4.686e-07	16	0.313s	performance goal reached
4.248e-07	6	0.188s	performance goal reached
5.168e-07	6	0.172s	performance goal reached
6.978e-08	8	0.188s	performance goal reached
parameters	hidden numbers = 20	trainFcn = traingd	
3.254e-04	10000	19.953s	maximum epoch reached
3.179e-04	10000	20.719s	maximum epoch reached
3.429e-04	10000	20.438s	maximum epoch reached
4.101e-04	10000	20.656s	maximum epoch reached
4.013e-04	10000	19.984s	maximum epoch reached
parameters	hidden numbers = 20	trainFcn = traincgf	
4.070e-05	487	2.766s	minimum step size reached
3.727e-05	568	3.156s	minimum step size reached
5.127e-05	490	2.641s	minimum step size reached
5.668e-05	487	2.719s	minimum step size reached
1.226e-05	465	2.656s	minimum step size reached

Table 2: hidden number =20

Q3

Repeat mearsure number = 5

(a)

hidden number =5

plot:

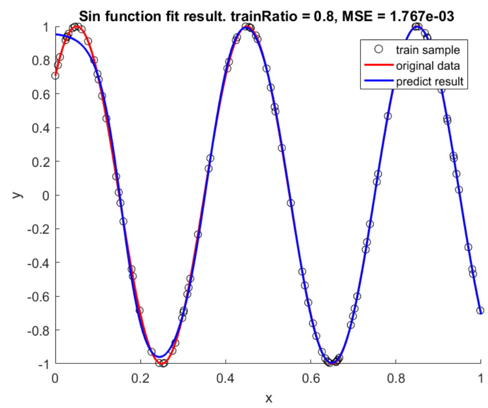

Figure 2: hidden number =5, trainFcn = trainlm

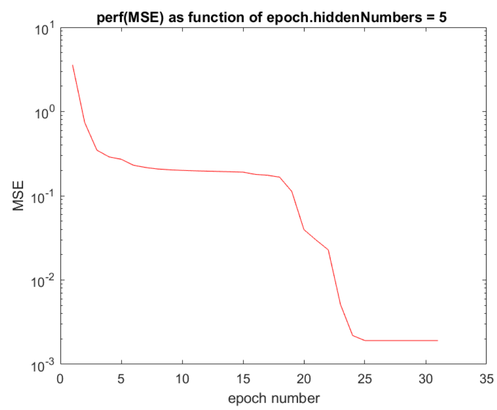

Figure 3: hidden number =5, trainFcn = trainlm

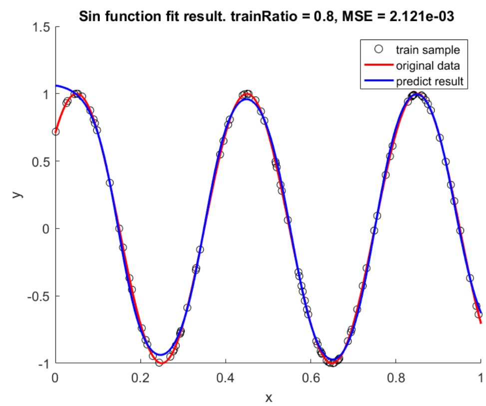

Figure 4: hidden number =5, trainFcn = traingd

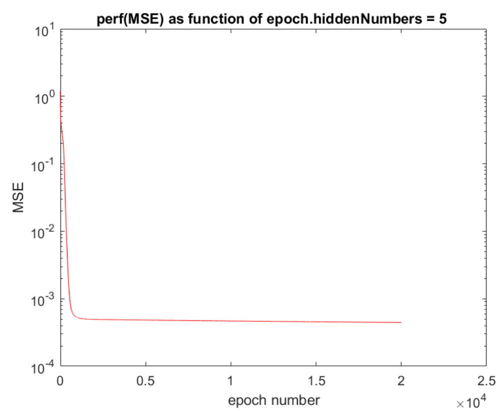

Figure 5: hidden number =5, trainFcn = traingd

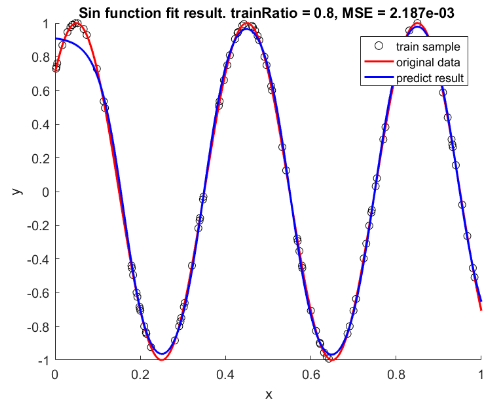

Figure 6: hidden number =5, trainFcn = traingcf

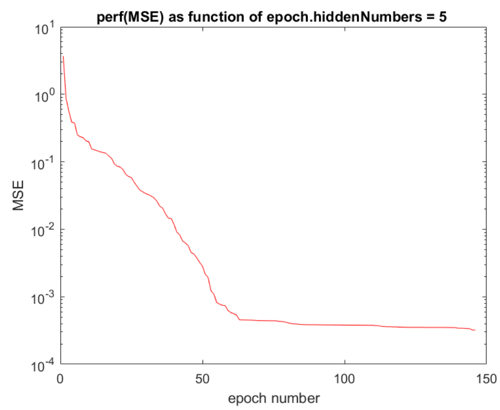

Figure 7: hidden number =5, trainFcn = traingcf

MSE(5 runs)	Epochs	reason of termination
Levenberg-Marquardt(trainlm)		
1.767e-03	39	minimum gradient
6.390e-04	23	minimum gradient
1.383e-03	100	maximum epoch(100)
1.640e-03	33	minimum gradient
3.169e-04	12	minimum gradient
gradient descent(traingd)		
2.121e-03	20000	maximum epoch(20000)
9.884e-04	20000	maximum epoch(20000)
2.050e-03	20000	maximum epoch(20000)
1.082e-03	20000	maximum epoch(20000)
1.592e-03	20000	maximum epoch(20000)
conjugate gradient(traincgf)		
7.272e-04	257	minimum step size
1.142e-03	390	minimum step size
3.231e-03	561	minimum step size
5.618e-04	577	minimum step size
3.382e-03	513	minimum step size

Table 3: hidden number =5

(b)

hidden number =20

plot:

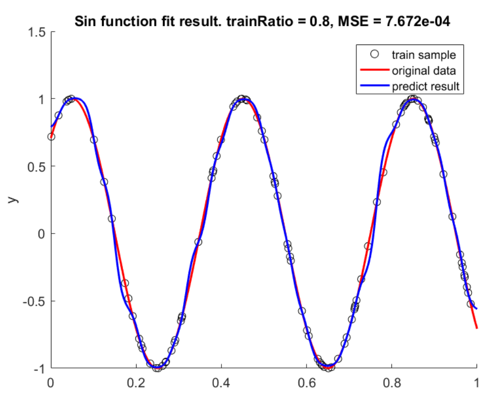

Figure 8: hidden number =20, trainFcn = trainlm

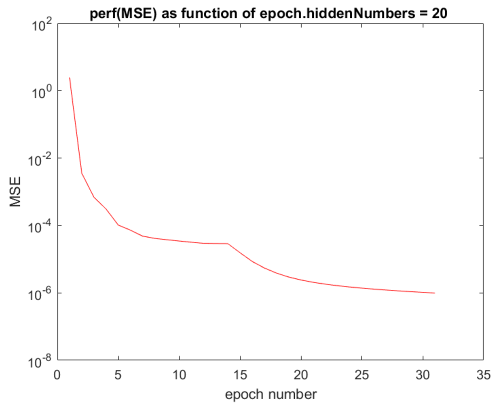

Figure 9: hidden number =20, trainFcn = trainlm

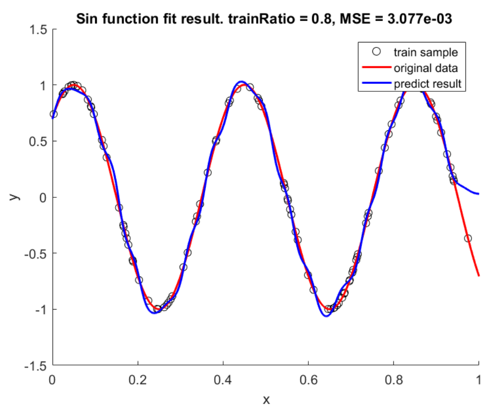

Figure 10: hidden number =20, trainFcn = traingd

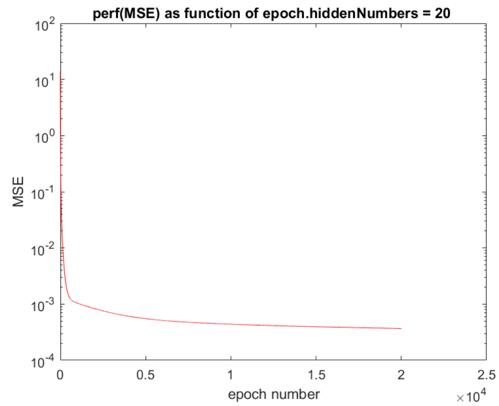

Figure 11: hidden number =20, trainFcn = traingd

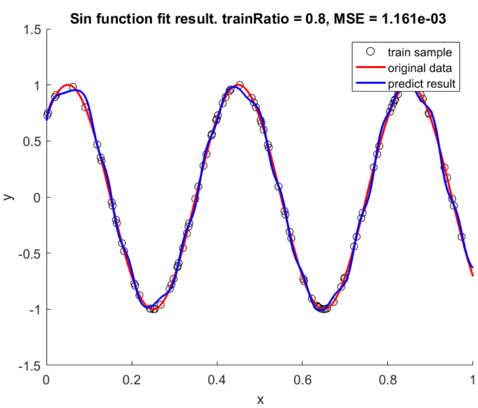

Figure 12: hidden number =20, trainFcn = traingcf

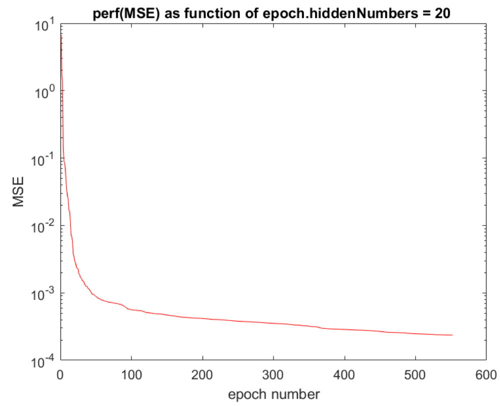

Figure 13: hidden number =20, trainFcn = traingcf

MSE	Epochs	reason of termination
Levenberg-Marquardt(trainlm)		
7.672e-04	30	Performance goal met
1.061e-04	35	Performance goal met
9.617e-04	47	Performance goal met
3.427e-04	44	Performance goal met
4.245e-03	26	Performance goal met
gradient descent(traingd)		
1.042e-03	20000	maximum epoch(20000)
3.077e-03	20000	maximum epoch(20000)
1.891e-03	20000	maximum epoch(20000)
1.686e-02	20000	maximum epoch(20000)
1.323e-03	20000	maximum epoch(20000)
conjugate gradient(traincgf)		
7.388e-04	916	minimum step size
1.161e-03	551	minimum step size
8.459e-04	675	minimum step size
7.019e-04	428	minimum step size
9.575e-04	611	minimum step size

Table 4: hidden number =20

(c) Comment

Apparently, with more hidden neurons, the fitting of the sin function is better. And with more training samples, the training error will be smaller , but the testing error may be larger due to the overffiting effect.

So when the training sample is not enough, increase the complexity of model can not always give us a better result.

Q4

(a)

Clearly, here the minimum value of K should be no greater than 4, since 4 is the sample number. And K should also be larger than 1 if we transform the data using radial function. So try K =2. And one possible solution is

$$\phi_1 = exp(-x^2), \phi_2 = exp(-(x-2)^2)$$

$x = [0, 2, 1, 3]^T$ will be transformed into:

$$\begin{bmatrix} 1 & 0.0183 \\ 0.0183 & 1 \\ 0.368 & 0.368 \\ 1.234e-4 & 0.368 \end{bmatrix}$$

The transformed data points and boundary is as follows:

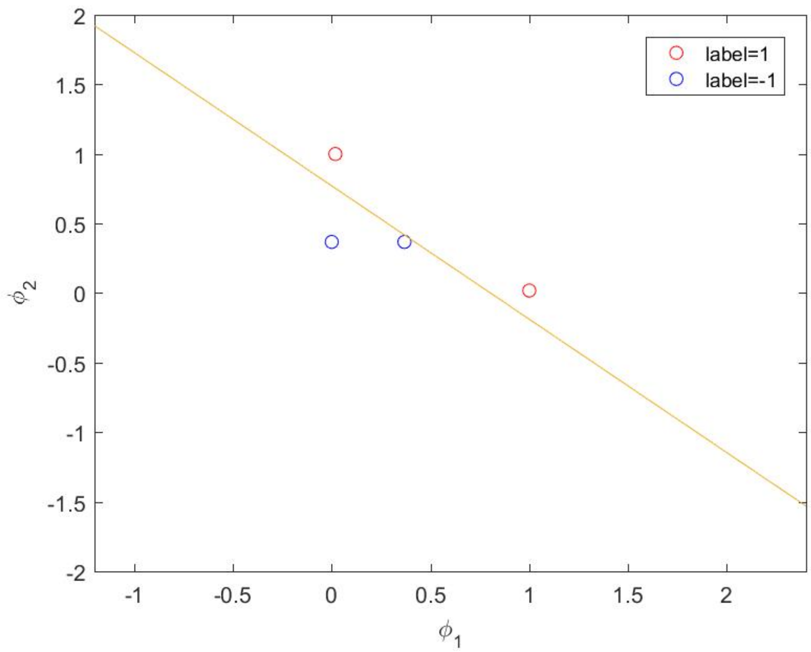

Figure 14: K=2

As the figure shows, now the data can be separate by line.

And the boundary point in the original 1D space is $0.727, 1.273, 2.515$. This was searched by matlab.

(b)

Here, I can find when K=3, the boundary has no bias term in the transformed space. Chosen radical functions:

$$\phi_1 = exp(-x^2), \phi_2 = exp(-(x-2)^2), \phi_3 = exp(-(x-1)^2/100),$$

$x = [0, 2, 1, 3]^T$ will be transformed into:

$$x2 = \begin{bmatrix} 1.0000 & 0.0183 & 0.9900 \\ 0.0183 & 1.0000 & 0.9900 \\ 0.3679 & 0.3679 & 1.0000 \\ 0.0001 & 0.3679 & 0.9608 \end{bmatrix}$$

We can choose $w = [1, 1, -1]^T$, without bias term,

$$x2 * w = \begin{bmatrix} 0.0283 \\ 0.0283 \\ -0.2642 \\ -0.5928 \end{bmatrix}$$

And

$$sign(x2 * w) = \begin{bmatrix} 1 \\ 1 \\ -1 \\ -1 \end{bmatrix}$$

Here the boundary point is $0.242, 1.788, 2.182$, which were also searched by matlab.

Q5

HW5

Q5 (a)

Using Non-linear transform as follows.

$x_1 \rightarrow (1, 0, 0, \cdots 0)^T$

$x_2 \rightarrow (0, 1, 0, \cdots 0)^T$

$x_N \rightarrow (0, 0, \cdots, \underset{N_{th}}{1})^T$

So $\vec{x}_i$ will be transformed into

$\vec{\Phi}_i = (0, 0, 0, \cdots \underset{i_{th}}{1}, 0, \cdots 0)^T$, Only i_{th} element is 1.

(b) C_1: (0, 5, 15, 20), label 1

C_2: (1, 2, 3, 4, 6, 7, 8, 9, 11, 12, 13, 14, 16, 17, 18, 19), label -1

Using the mapping defined in (a)

We can have W, as

$W = (\underset{0}{1}, -1, -1, -1, -1, \underset{5_{th}}{1}, -1, -1, -1, -1, \underset{10_{th}}{1}, -1, -1, -1, -1, \underset{15_{th}}{1}, -1, -1, -1, -1, \underset{20_{th}}{1})^T$

such that $W^T \cdot \vec{\phi}_i = \begin{cases} 1 > 0, \text{if sample } i \in C_1 \\ -1 < 0, \text{if sample } i \in C_2 \end{cases}$

For example, 1^{st} sample 0 is mapped to $\vec{\Phi}_0 = (1, 0, \cdots 0)^T$

$\therefore W^T \phi_0^T = 1 > 0$

Similar for other data points

Figure 15: Q5

Q6

During the training, I disabled the validation and test dataset since here the number of training sample is so small (only 4).

two hidden nuerons

two typical plots

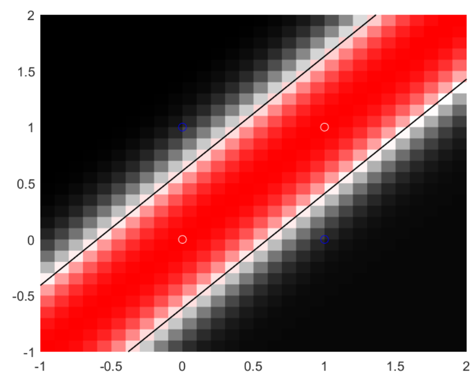

Figure 16: hidden number =2,trainFcn = trainlm

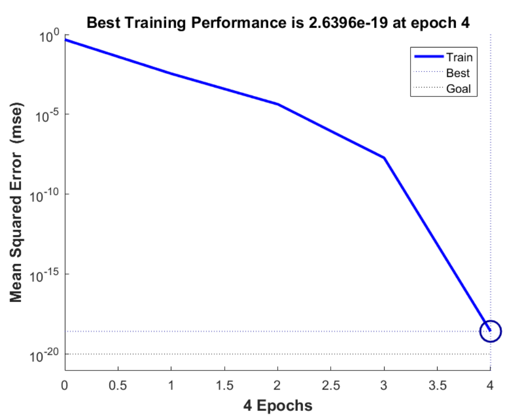

Figure 17: hidden number =2,trainFcn = trainlm

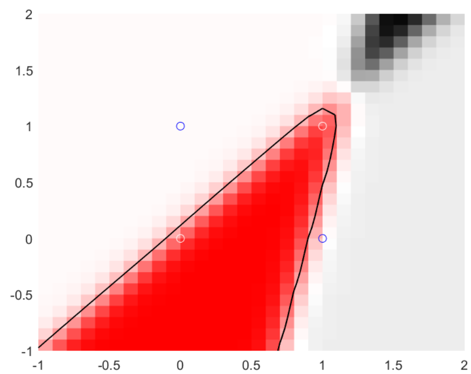

Figure 18: hidden number =2,trainFcn = trainlm

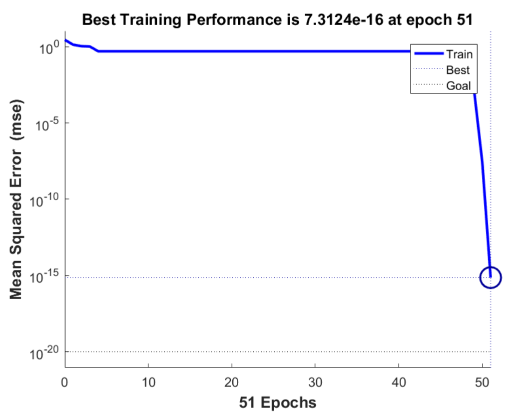

Figure 19: hidden number =2,trainFcn = trainlm

four hidden nuerons

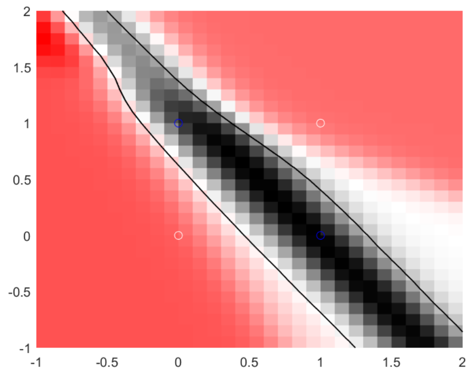

Figure 20: hidden number =4,trainFcn = trainlm

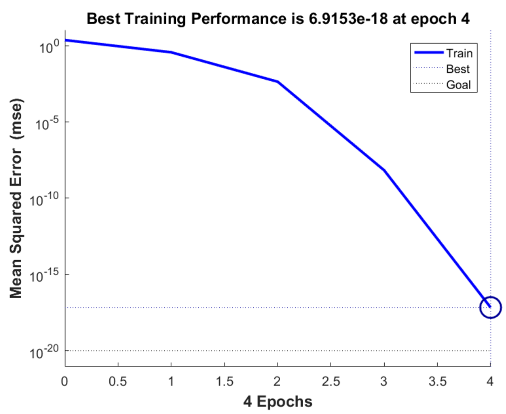

Figure 21: hidden number =4,trainFcn = trainlm

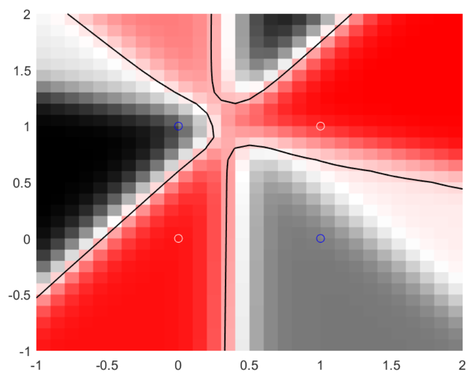

Figure 22: hidden number =4,trainFcn = trainlm

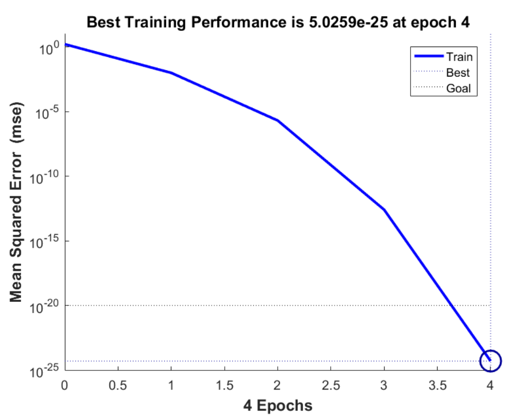

Figure 23: hidden number =4,trainFcn = trainlm

Comment

Similar with MLPin the hw4, although 2 hidden neuron is enough, and the boundary of 4 hidden neurons may be similar to the one of 2 hidden neurons, we can not delete the some of the weights of the 4 hidden neurons.

Q7

HW5

7.

According to Cover's Theorem

We have $P(4,2) = \left(\frac{1}{2}\right)^3 \left[\binom{3}{1} + \binom{3}{0} \right]$

$$= \frac{1}{8} [3+1]$$

$$= \frac{1}{2}$$

So the probability is $\frac{1}{2}$

Figure 24: Q7