1 Output Layer

1

$$x_1 w_1 + x_2 w_2 + b_1 = 2 \times 2 + 1 \times (-1) + 1 = 4$$
$$y_1 = \frac{1}{1 + e^{-4}} = 0.982$$

2

$$E(w) = \frac{1}{2}\sum_{n=1}^{N}(d_n - y_n)^2$$
$$\frac{\partial E}{\partial w_i} = \frac{\partial E}{\partial y_n}\frac{\partial y_n}{\partial v_n}\frac{\partial v_n}{\partial w_i}$$
$$\frac{\partial E}{\partial y_n} = \sum_{n=1}^{N} -(d_n - y_n)$$
$$\frac{\partial y_n}{\partial v_n} = \frac{\partial}{\partial v_n}\frac{1}{1 + \exp(-v_n)}$$
$$= \frac{-\frac{\partial}{\partial v_n}(1 + \exp(-v_n))}{(1 + \exp(-v_n))^2}$$
$$= y_n(1 - y_n)$$
$$\frac{\partial v_n}{\partial w_i} = \frac{\partial}{\partial w_i} w^T x_n = x_{ni}$$

$So,$

$$\frac{\partial E}{\partial w_i} = \sum_{n=1}^{N} -(d_n - y_n)y_n(1 - y_n)x_{ni}$$
$$w(n+1) = w(n) - \eta\sum_{n=1}^{N} -(d_n - y_n)y_n(1 - y_n)x_{ni}$$

3

$$\frac{\partial E}{\partial w_1} = -(1 - 0.982) \times 0.982 \times (1 - 0.982) \times 2$$
$$= -0.000636336$$

$$w_1^2 = w_1^1 - \eta \frac{\partial E}{\partial w_1^1} = 2 - \big(1 \times (-0.000636336)\big) = 2.000636336$$

By the same logic,

$$\frac{\partial E}{\partial w_2} = -(1 - 0.982) \times 0.982 \times (1 - 0.982) \times 1$$
$$= -0.000318168$$
$$w_2^2 = w_2^1 - \eta \frac{\partial E}{\partial w_2^1} = 1 - \big(1 \times (-0.000318168)\big) = 1.000318168$$
$$\frac{\partial E}{\partial b_1} = -(1 - 0.982) \times 0.982 \times (1 - 0.982) \times 1$$
$$= -0.000318168$$
$$b_1^2 = b_1^1 - \eta \frac{\partial E}{\partial b_1^1} = 1 - \big(1 \times (-0.000318168)\big) = 1.000318168$$

2 Single Hidden Layer

1

$$net_{h1} = 1 \times 1 + (-1) \times 2 + 3 = 2$$
$$net_{h2} = 1 \times (-1) + (-1) \times (-2) + 4 = 5$$
$$out_{h1} = \frac{1}{1 + e^{-2}} = 0.881$$
$$out_{h2} = \frac{1}{1 + e^{-5}} = 0.993$$
$$net_{out} = 0.881 \times (-1) + 0.993 \times 1 + 0 = 0.112$$
$$y_1 = \frac{1}{1 + e^{-0.112}} = 0.528$$

2

$$E(w) = \frac{1}{2} \sum_{n=1}^{N} e_n^2 = \frac{1}{2} \sum_{n=1}^{N} (d_n - y_n)^2 = \frac{1}{2} \sum_{n=1}^{N} \Big(d_n - \phi_n\big(v_n(n)\big)\Big)^2$$
$$\frac{\partial E(n)}{\partial w_{lj}} = \frac{\partial E(n)}{\partial e_l(n)} \frac{\partial e_l(n)}{\partial y_l(n)} \frac{\partial y_l(n)}{\partial v_l(n)} \frac{\partial v_l(n)}{\partial w_{lj}}$$
$$= [e_l][-1]\big[\phi'\big(v_n(n)\big)\big][y_{jl}(n)]$$
$$Let\ \delta_l(n) = -\frac{\partial E(n)}{\partial v_l(n)} = [e_l]\big[\phi'\big(v_n(n)\big)\big]$$

$$Let\ \delta_j(n) = -\frac{\partial E(n)}{\partial v_j(n)} = -\frac{\partial E(n)}{\partial y_j(n)}\phi'\left(v_j(n)\right)$$

$$\frac{\partial E(n)}{\partial y_j(n)} = \sum_l e_l(n)\frac{\partial e_l(n)}{\partial v_l(n)}\frac{\partial v_l(n)}{\partial y_j(n)}$$

$$= \sum_l e_l(n)\left[-\phi'\big(v_l(n)\big)\right][w_{lj}(n)]$$

$So,$

$$\delta_j(n) = \sum_l e_l(n)\left[-\phi'\big(v_l(n)\big)\right][w_{lj}(n)]\phi'\left(v_j(n)\right)$$

$$= \phi'\left(v_j(n)\right)\sum_l \delta_l(n)w_{lj}(n)$$

$$w(n+1) = w(n) + \eta\delta_j(n)y_i(n)$$

3

$$\frac{\partial E}{\partial w_5} = -(0-0.528)\times 0.528\times(1-0.528)\times 0.881$$

$$= 0.116$$

$$w_5^2 = w_5^1 - \eta\frac{\partial E}{\partial w_5^1} = -1 - \big(1\times(0.116)\big) = -1.116$$

$$\frac{\partial E}{\partial w_6} = -(0-0.528)\times 0.528\times(1-0.528)\times 0.993$$

$$= 0.131$$

$$w_6^2 = w_6^1 - \eta\frac{\partial E}{\partial w_6^1} = 1 - \big(1\times(0.131)\big) = 0.869$$

$$\frac{\partial E}{\partial w_1} = -(0-0.528)\times 0.528\times(1-0.528)\times(-1.116)\times 0.881\times(1-0.881)\times 1$$

$$= -0.0154$$

$$w_1^2 = w_1^1 - \eta\frac{\partial E}{\partial w_1^1} = 1 - \big(1\times(-0.0154)\big) = 1.0154$$

$$\frac{\partial E}{\partial w_2} = -(0-0.528)\times 0.528\times(1-0.528)\times 0.869\times 0.993\times(1-0.993)\times 1$$

$$= 0.0008$$

$$w_2^2 = w_2^1 - \eta\frac{\partial E}{\partial w_2^1} = -1 - \big(1\times(0.0008)\big) = -1.0008$$

3 UF Network

1

I will need both 80 units in the first and second hidden layers. Because if there are too few units, the training and testing result is not good (model fitting ability and accuracy are lower). But too many units may lead to overfitting and spend too much time. So, I choose 80 units after I tried to choose other number of units.

```
Train Epoch: 16700      Loss: 0.034519
Train Epoch: 16800      Loss: 0.026608
Train Epoch: 16900      Loss: 0.023400
Train Epoch: 17000      Loss: 0.031680
Train Epoch: 17100      Loss: 0.031186
Train Epoch: 17200      Loss: 0.023038
Train Epoch: 17300      Loss: 0.024377
Train Epoch: 17400      Loss: 0.030428
Train Epoch: 17500      Loss: 0.024355
Train Epoch: 17600      Loss: 0.023003
Train Epoch: 17700      Loss: 0.029710
Train Epoch: 17800      Loss: 0.028891
Train Epoch: 17900      Loss: 0.022046
Train Epoch: 18000      Loss: 0.022188
Train Epoch: 18100      Loss: 0.030373
Train Epoch: 18200      Loss: 0.023541
Train Epoch: 18300      Loss: 0.019114
Train Epoch: 18400      Loss: 0.023821
Train Epoch: 18500      Loss: 0.022669
Train Epoch: 18600      Loss: 0.016139
Train Epoch: 18700      Loss: 0.021252
Train Epoch: 18800      Loss: 0.024759
Train Epoch: 18900      Loss: 0.017068
Train Epoch: 19000      Loss: 0.014305
Train Epoch: 19100      Loss: 0.020567
Train Epoch: 19200      Loss: 0.020425
Train Epoch: 19300      Loss: 0.014453
Train Epoch: 19400      Loss: 0.013901
Train Epoch: 19500      Loss: 0.017450
Train Epoch: 19600      Loss: 0.012326
Train Epoch: 19700      Loss: 0.010319
Train Epoch: 19800      Loss: 0.013262
Train Epoch: 19900      Loss: 0.015245
```

Figure 1 units=[70,70]

```
Train Epoch: 16700      Loss: 0.004054
Train Epoch: 16800      Loss: 0.003420
Train Epoch: 16900      Loss: 0.003392
Train Epoch: 17000      Loss: 0.003165
Train Epoch: 17100      Loss: 0.002132
Train Epoch: 17200      Loss: 0.003902
Train Epoch: 17300      Loss: 0.002223
Train Epoch: 17400      Loss: 0.003272
Train Epoch: 17500      Loss: 0.002339
Train Epoch: 17600      Loss: 0.003243
Train Epoch: 17700      Loss: 0.002390
Train Epoch: 17800      Loss: 0.001936
Train Epoch: 17900      Loss: 0.002383
Train Epoch: 18000      Loss: 0.002623
Train Epoch: 18100      Loss: 0.002707
Train Epoch: 18200      Loss: 0.002566
Train Epoch: 18300      Loss: 0.002562
Train Epoch: 18400      Loss: 0.002317
Train Epoch: 18500      Loss: 0.001957
Train Epoch: 18600      Loss: 0.001757
Train Epoch: 18700      Loss: 0.001578
Train Epoch: 18800      Loss: 0.001247
Train Epoch: 18900      Loss: 0.001237
Train Epoch: 19000      Loss: 0.001682
Train Epoch: 19100      Loss: 0.001547
Train Epoch: 19200      Loss: 0.000999
Train Epoch: 19300      Loss: 0.000813
Train Epoch: 19400      Loss: 0.001101
Train Epoch: 19500      Loss: 0.001292
Train Epoch: 19600      Loss: 0.000937
Train Epoch: 19700      Loss: 0.000715
Train Epoch: 19800      Loss: 0.000705
Train Epoch: 19900      Loss: 0.000719
```

Figure 2 units=[80,80]

```
Train Epoch: 16700      Loss: 0.010801
Train Epoch: 16800      Loss: 0.011089
Train Epoch: 16900      Loss: 0.010842
Train Epoch: 17000      Loss: 0.010450
Train Epoch: 17100      Loss: 0.012272
Train Epoch: 17200      Loss: 0.010379
Train Epoch: 17300      Loss: 0.008910
Train Epoch: 17400      Loss: 0.009287
Train Epoch: 17500      Loss: 0.011722
Train Epoch: 17600      Loss: 0.009600
Train Epoch: 17700      Loss: 0.009119
Train Epoch: 17800      Loss: 0.008355
Train Epoch: 17900      Loss: 0.006461
Train Epoch: 18000      Loss: 0.004057
Train Epoch: 18100      Loss: 0.004910
Train Epoch: 18200      Loss: 0.004223
Train Epoch: 18300      Loss: 0.002938
Train Epoch: 18400      Loss: 0.002157
Train Epoch: 18500      Loss: 0.001681
Train Epoch: 18600      Loss: 0.001999
Train Epoch: 18700      Loss: 0.002756
Train Epoch: 18800      Loss: 0.002342
Train Epoch: 18900      Loss: 0.002615
Train Epoch: 19000      Loss: 0.001402
Train Epoch: 19100      Loss: 0.001019
Train Epoch: 19200      Loss: 0.000920
Train Epoch: 19300      Loss: 0.000944
Train Epoch: 19400      Loss: 0.001645
Train Epoch: 19500      Loss: 0.001324
Train Epoch: 19600      Loss: 0.001092
Train Epoch: 19700      Loss: 0.001270
Train Epoch: 19800      Loss: 0.000939
Train Epoch: 19900      Loss: 0.000755
```

Figure 3 units=[90,90]

2

a learning curve

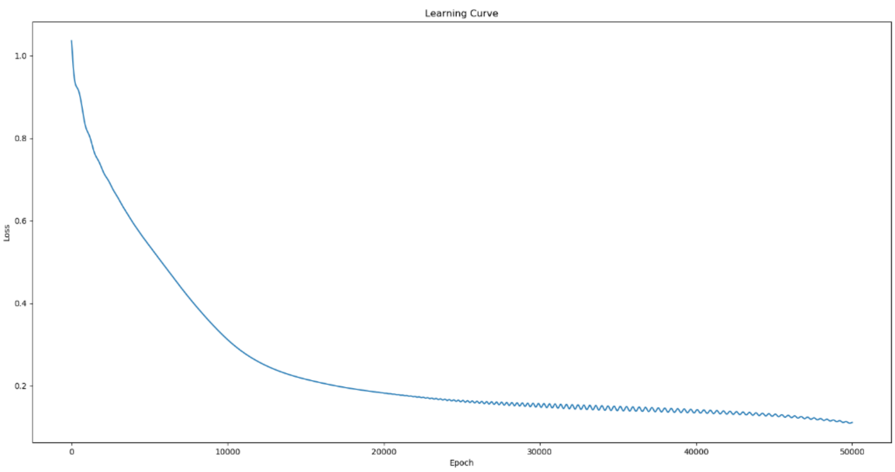

Figure 4 learning rate=0.00001 epochs=50000

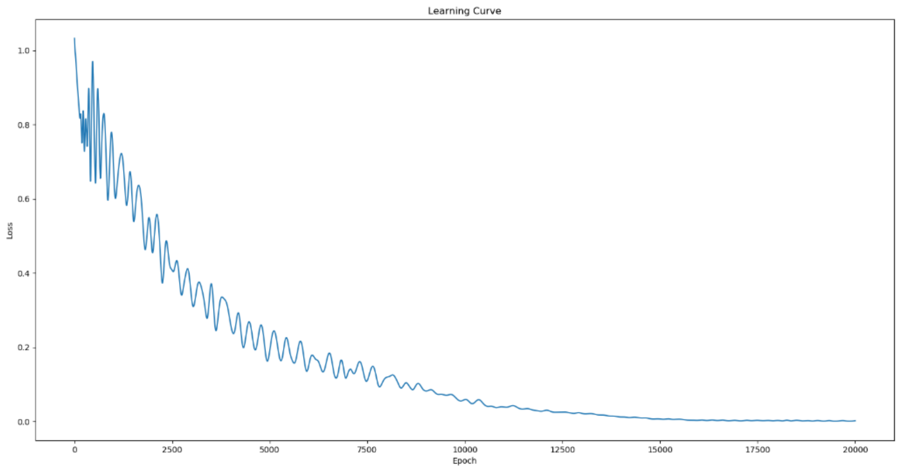

Figure 5　learning rate=0.0001 epochs=20000

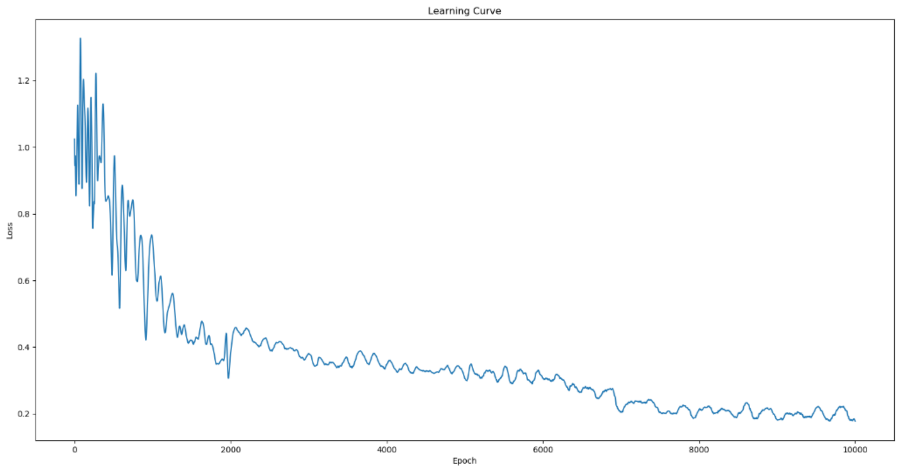

Figure 6　learning rate=0.001 epochs=10000

b decision boundary

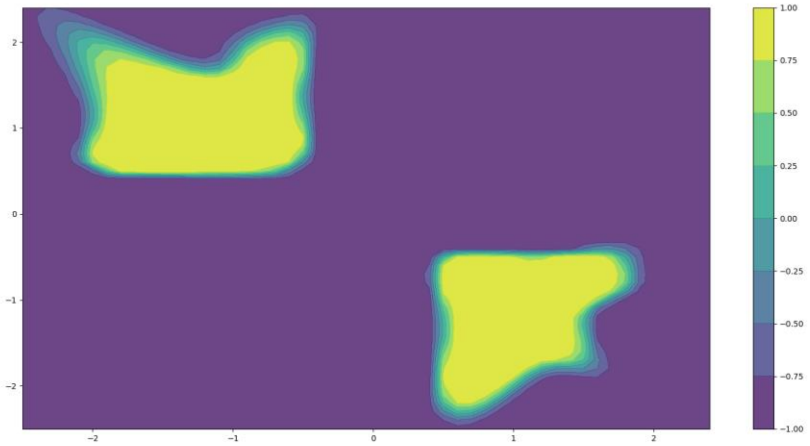

Figure 7　learning rate=0.00001 epochs=50000

Figure 8　learning rate=0.0001 epochs=20000

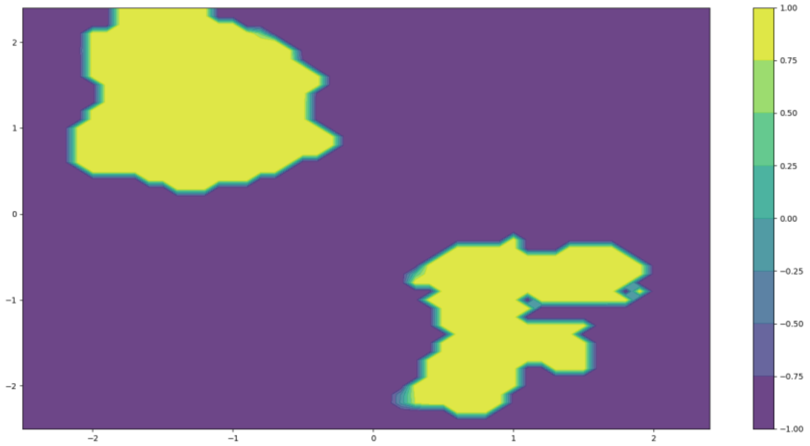

Figure 9　learning rate=0.001 epochs=10000

3

a

If learning rate is small, gradient descent can be slow. And it will be cost much more time to optimize the model. If learning rate is high, gradient descent can overshoot the minimum. It may fail to converge, or even diverge.

If the number of epochs is small, the updates of the weights in the neural network is small so that the model is underfitting. If the number of epochs is big, the updates of the weights in the neural network is big and the model may overfitting. Also, it will cost too much time to training the model.

b

Because I have tried many values and selected the best learning rate and number epochs whose learning curve and decision boundary look better.