

Lab1: back-propagation

Student id / name: A113599 / 楊淨富

I. Introduction:

利用兩層 hidden layer 的 Neural network 來訓練 input 為二維的資料(range from 0.0 to 1.0) 以及其所對應到的 label 為一維的 ground truth (0 or 1)。因為資料集較小，故 training set 同時也是 testing set。

II. Experiment setups

A. Activation functions

- Sigmoid function:

```
def sigmoid(self, x, derivative=False):  
    if derivative:  
        return (np.exp(-x))/((np.exp(-x)+1)**2)  
    return 1/(1 + np.exp(-x))
```

Sigmoid function 是一種 activation function，其輸出為 0 至 1。

derivative 若設為 True 則是 sigmoid 微分後的函數。反之則為 sigmoid function。

- Relu funtion:

```
def relu(self, x, derivative=False):  
    if derivative:  
        return np.where(x < 0, 0, 1)  
    return np.maximum(0, x)
```

Relu 也是一種 activation function，假定輸入為 x ，則輸出為 $\max(0, x)$ 。

比較麻煩的是其在 0 這個點無法進行微分(因為左右兩點的斜率不同)。

B. Neural network

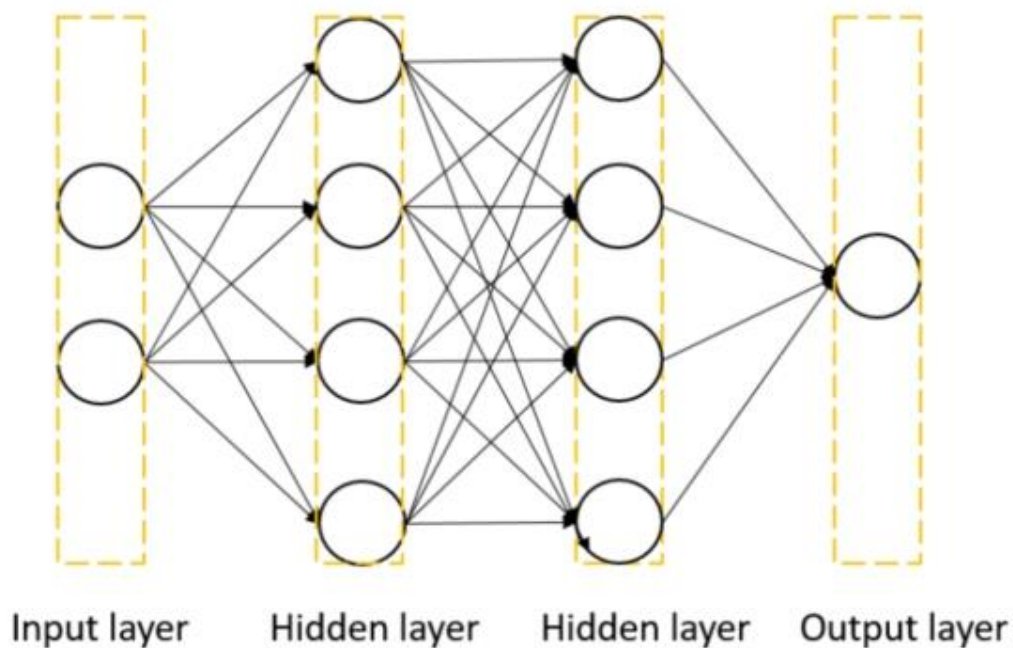


Figure 1. Two-layer neural network

- Layers:
 - Input layer: 由於 input 是二維，所以兩個 neuron。
 - Hidden layer: 要求至少兩層 hidden layers。我預設兩層(h1, h2)各 4 個 neuron。
 - Output layer: 由於 label 是 0 或 1，所以一個 neuron。
- Training process:
 1. 先隨機初始化 parameters (此 lab 即 weights)
 2. 透過 feed_forward 得出 output
 3. 進行 back_propagation(將 Loss function 對各 weights 做微分，算出各 weights 之 gradient)
 4. 透過 optimize 更新各 weights
 5. 重複進行 Step 2 - 4

C. Back propagation:

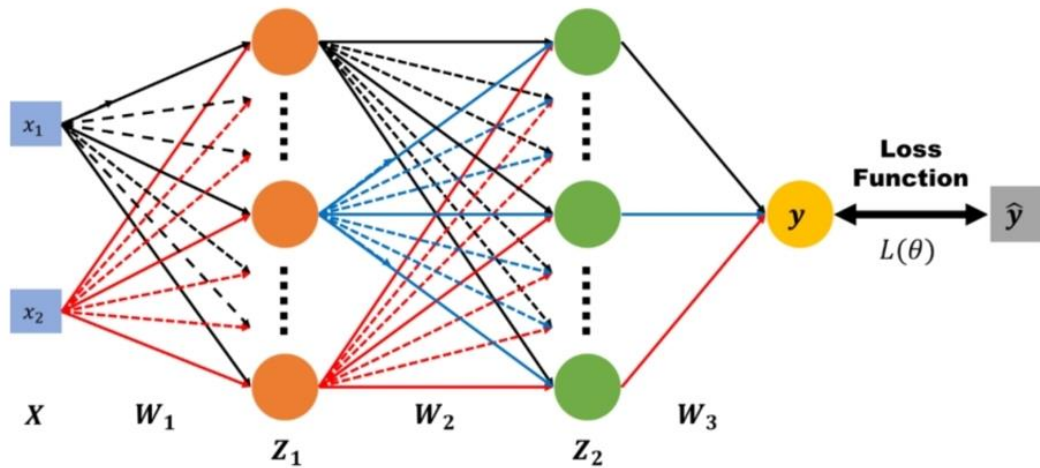


Figure 2. Forward pass

- In the figure 2, we use the following definitions for the notations:

1. x_1, x_2 : *nerual network inputs*
 2. $X : [x_1, x_2]$
 3. y : *nerual network outputs*
 4. \hat{y} : *ground truth*
 5. $L(\theta)$: *loss function*
 6. W_1, W_2, W_3 : *weight matrix of network layers*
- $$Z_1 = \sigma(XW_1) \quad Z_2 = \sigma(Z_1W_2) \quad y = \sigma(Z_2W_3)$$

- 由 output 依序向前更新各 weights
- 更新方法為:

將對 Loss function 對各 weight 做偏微分，計算出各 weight 之 gradients 後，算出 $\text{new_weight} = \text{old_weight} - \text{learning_rate} * \text{weight_gradient}$ 。如下圖所示 (依序更新 W_3, W_2, W_1)

$$\begin{array}{ccccc}
 x & \cdot & W1 & H1 & \xrightarrow{\sigma} & Z1 \\
 (100, 2) & & (2, 4) & (100, 4) & & (100, 4)
 \end{array}$$

$$\begin{array}{ccccc}
 Z1 & \cdot & W2 & H2 & \xrightarrow{\sigma} & Z2 \\
 (100, 4) & & (4, 4) & (100, 4) & & (100, 4)
 \end{array}$$

$$\begin{array}{ccccc}
 Z2 & \cdot & W3 & H3 & \xrightarrow{\sigma} & Z3 \text{ (output)} = (y - \hat{y})^2 \\
 (100, 4) & & (4, 1) & (100, 1) & & (100, 1)
 \end{array}$$

$$\begin{aligned}
 \frac{dL}{dW3} &= \frac{dL}{dZ3} \times \frac{dZ3}{dH3} \times \frac{dH3}{dW3} \\
 &= 2 \times (y - \hat{y}) \times (1 - 0) \times \text{sig-derivate}(H3) \times Z2
 \end{aligned}$$

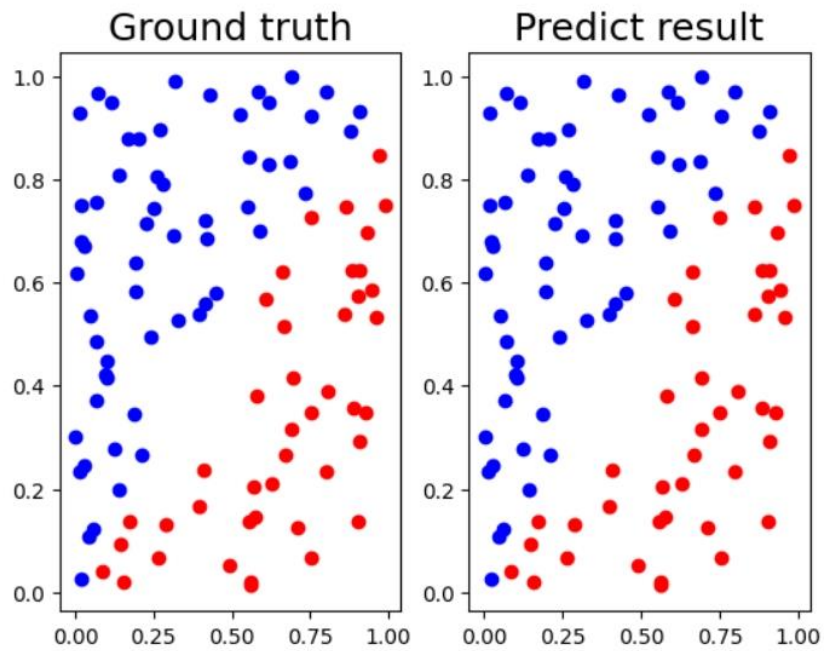
$$\begin{aligned}
 \frac{dL}{dW2} &= \left(\frac{dL}{dZ3} \times \frac{dZ3}{dH3} \right) \times \frac{dH3}{dZ2} \times \frac{dZ2}{dH2} \times \frac{dH2}{dW2} \\
 &= 2 \times (y - \hat{y}) \times (1 - 0) \times \text{sig-derivate}(H3) \times W3 \times \text{sig-derivate}(H2) \times Z1
 \end{aligned}$$

$$\begin{aligned}
 \frac{dL}{dW1} &= \left(\frac{dL}{dZ3} \times \frac{dZ3}{dH3} \times \frac{dH3}{dZ2} \times \frac{dZ2}{dH2} \right) \times \frac{dH2}{dZ1} \times \frac{dZ1}{dH1} \times \frac{dH1}{dW1} \\
 &= 2 \times (y - \hat{y}) \times \text{s-d}(H3) \times W3 \times \text{s-d}(H2) \times W2 \times \text{s-d}(H1) \times x
 \end{aligned}$$

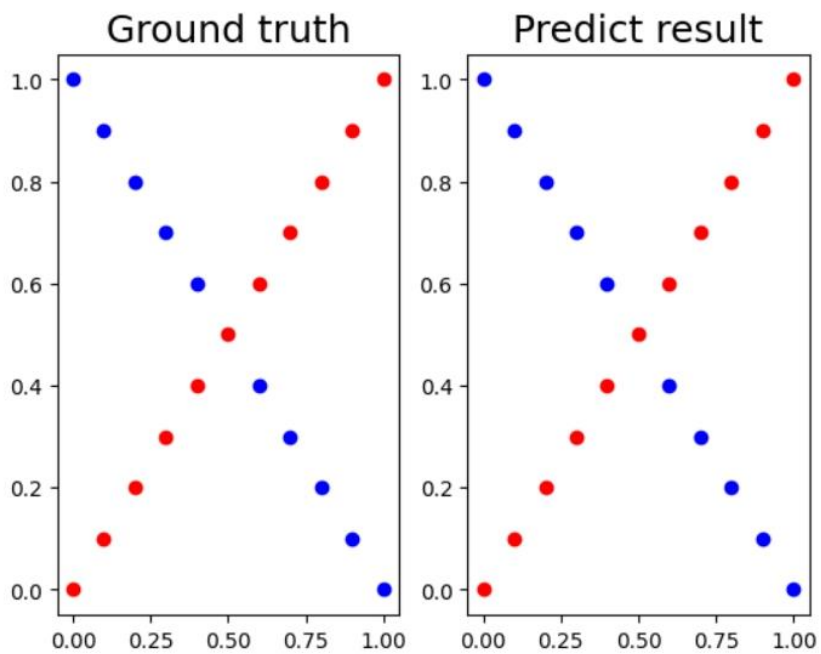
III. Results of your testing

A. Screenshot and comparison figure

- Case 1: Linear



- Case 2: XOR



B. Show the accuracy of your prediction

- Case 1: Linear

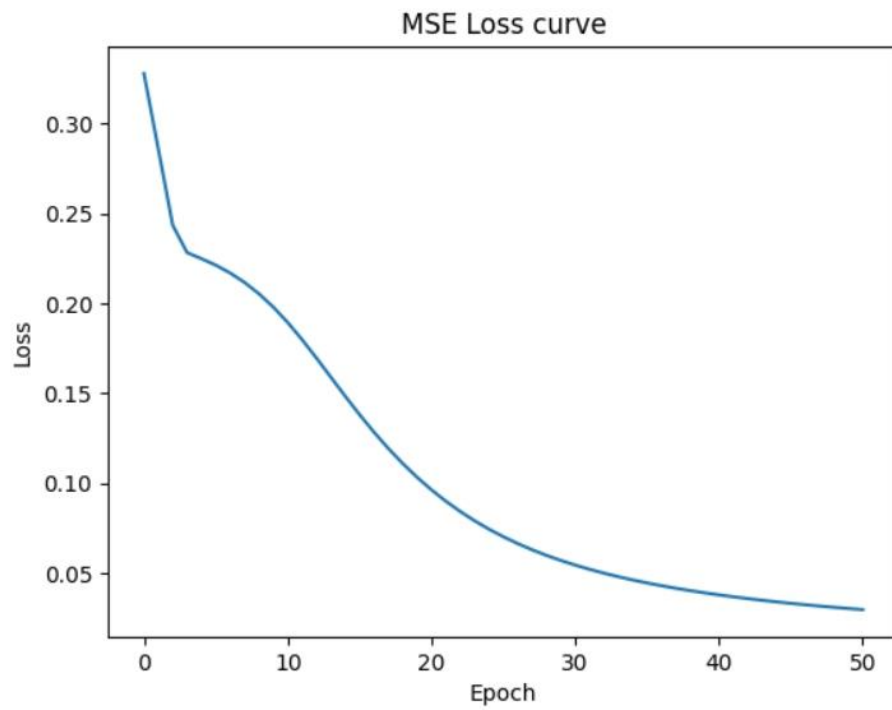
```
Epoch: 0 loss:0.327794 acc:58.00%
Epoch: 5 loss:0.221193 acc:58.00%
Epoch: 10 loss:0.189331 acc:77.00%
Epoch: 15 loss:0.138072 acc:92.00%
Epoch: 20 loss:0.096261 acc:96.00%
Epoch: 25 loss:0.070163 acc:99.00%
Epoch: 30 loss:0.054393 acc:99.00%
Epoch: 35 loss:0.044466 acc:99.00%
Epoch: 40 loss:0.037827 acc:100.00%
Epoch: 45 loss:0.033126 acc:100.00%
Epoch: 50 loss:0.029633 acc:100.00%
```

- Case 2: XOR

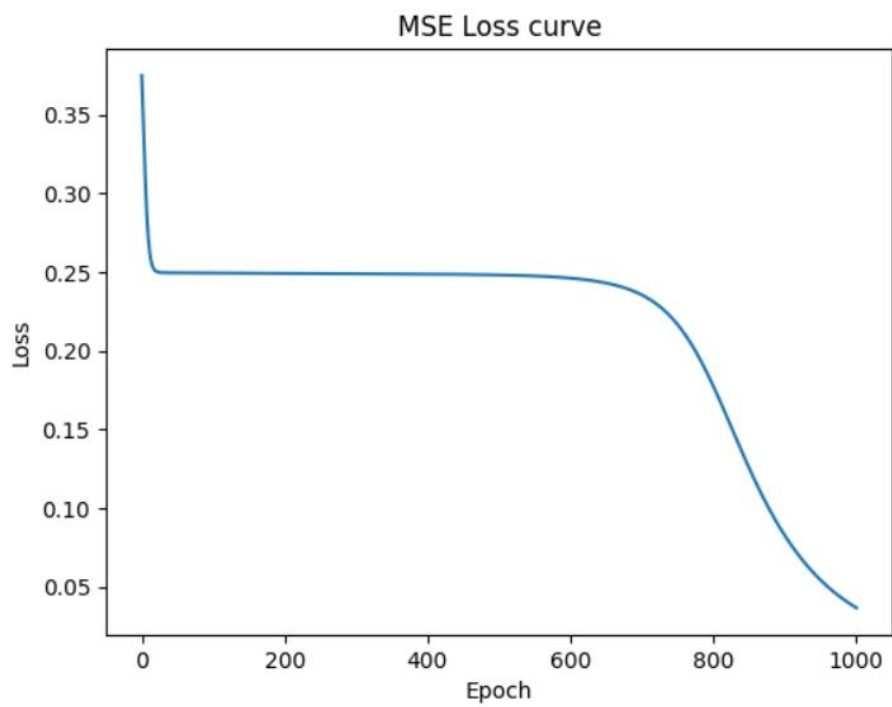
```
Epoch: 0 loss:0.374859 acc:47.62%
Epoch: 100 loss:0.249373 acc:52.38%
Epoch: 200 loss:0.249138 acc:52.38%
Epoch: 300 loss:0.248945 acc:52.38%
Epoch: 400 loss:0.248721 acc:52.38%
Epoch: 500 loss:0.248191 acc:52.38%
Epoch: 600 loss:0.246140 acc:57.14%
Epoch: 700 loss:0.235422 acc:71.43%
Epoch: 800 loss:0.177223 acc:90.48%
Epoch: 900 loss:0.082453 acc:95.24%
Epoch:1000 loss:0.036723 acc:100.00%
```

C. Learning curve(loss, epoch curve)

- Case 1: Linear



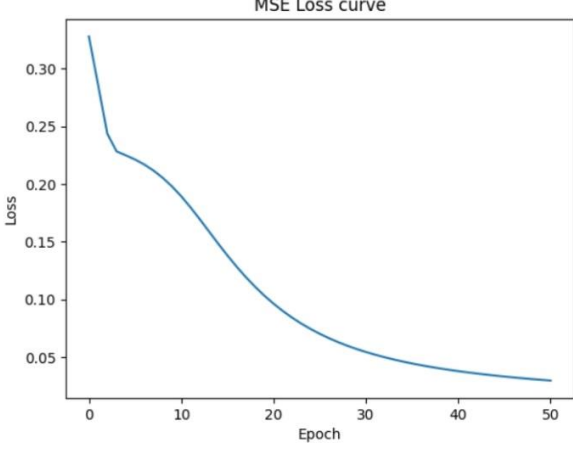
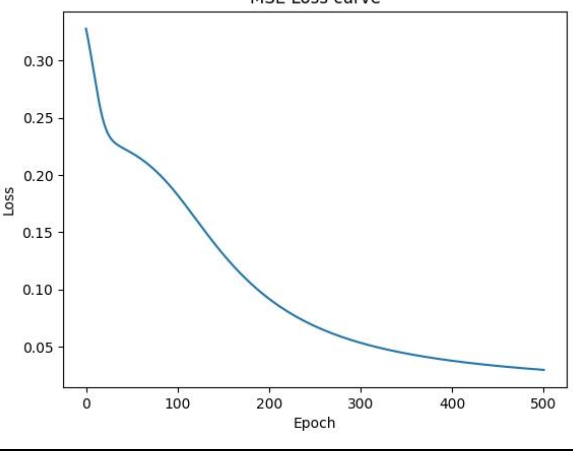
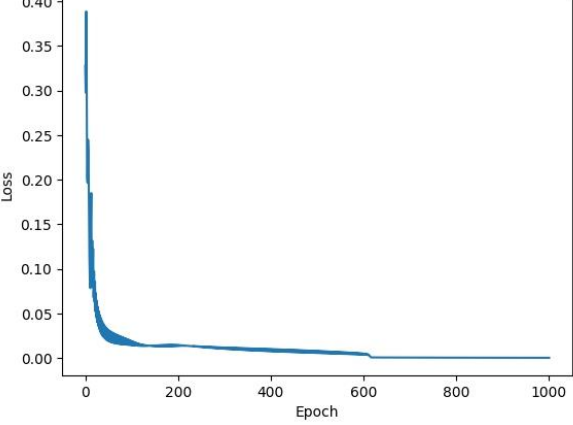
- Case 2: XOR



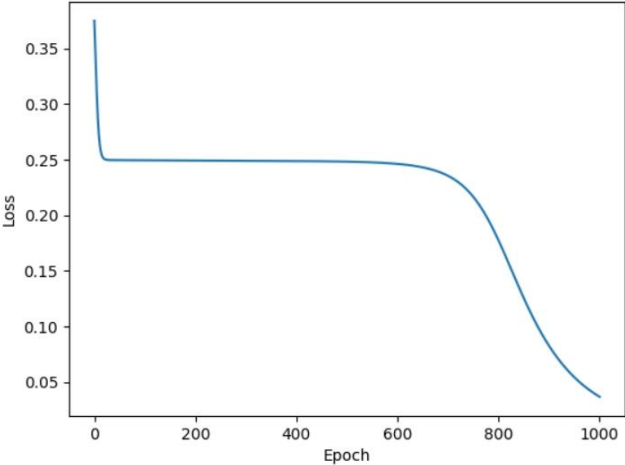
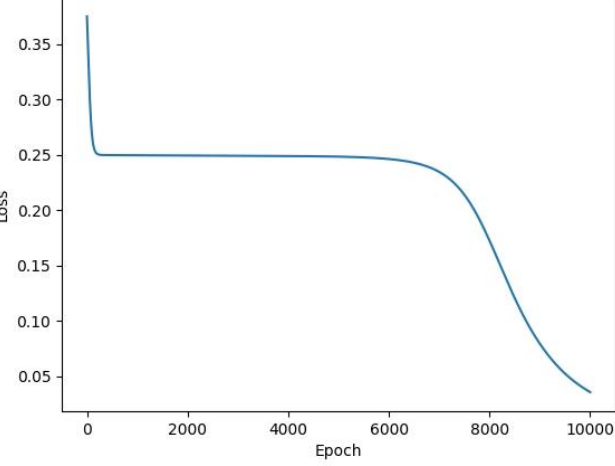
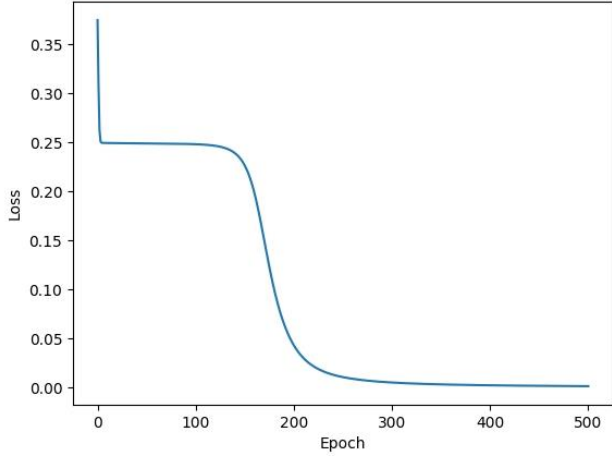
4. Discussion

A. Try different learning rates

- Case 1: Linear

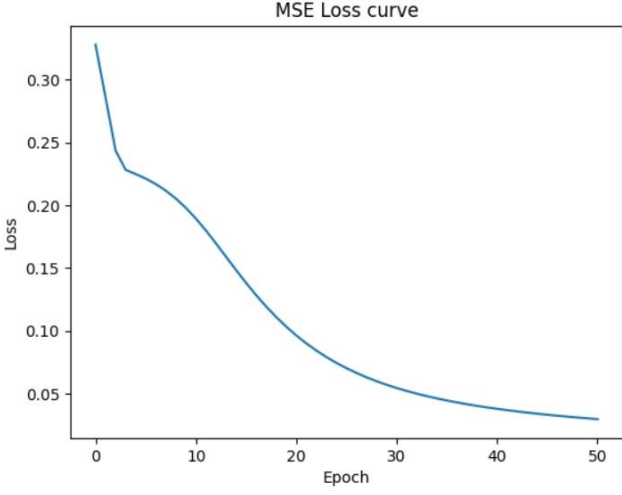
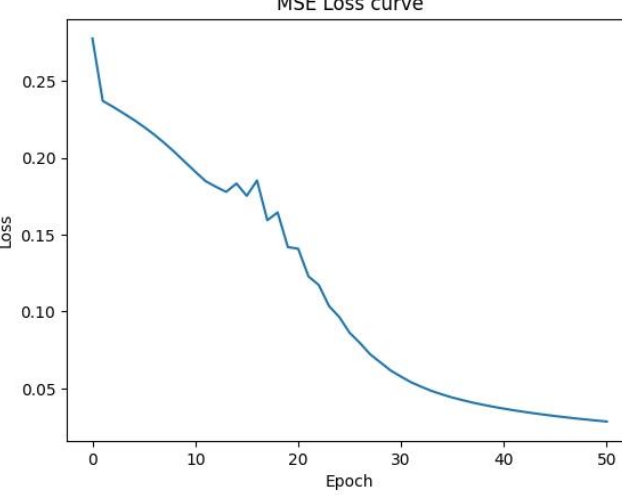
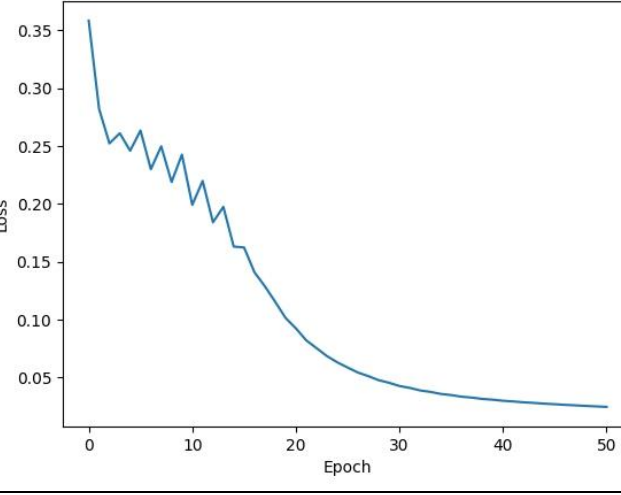
learing_rate	Epochs to acc=100%	Loss curve																
0.1	38	 <p>The graph shows the Mean Squared Error (MSE) loss over 50 epochs for a learning rate of 0.1. The loss starts at approximately 0.32 and decreases steadily, reaching about 0.03 by epoch 50. The curve is smooth and shows a slight inflection point around epoch 10.</p> <table><caption>Approximate data points for learning rate 0.1</caption><tr><th>Epoch</th><th>Loss</th></tr><tr><td>0</td><td>0.32</td></tr><tr><td>10</td><td>0.20</td></tr><tr><td>20</td><td>0.10</td></tr><tr><td>30</td><td>0.05</td></tr><tr><td>40</td><td>0.03</td></tr><tr><td>50</td><td>0.03</td></tr></table>	Epoch	Loss	0	0.32	10	0.20	20	0.10	30	0.05	40	0.03	50	0.03		
Epoch	Loss																	
0	0.32																	
10	0.20																	
20	0.10																	
30	0.05																	
40	0.03																	
50	0.03																	
0.01	366	 <p>The graph shows the Mean Squared Error (MSE) loss over 500 epochs for a learning rate of 0.01. The loss starts at approximately 0.32 and decreases steadily, reaching about 0.03 by epoch 500. The curve is smooth and shows a slight inflection point around epoch 100.</p> <table><caption>Approximate data points for learning rate 0.01</caption><tr><th>Epoch</th><th>Loss</th></tr><tr><td>0</td><td>0.32</td></tr><tr><td>100</td><td>0.20</td></tr><tr><td>200</td><td>0.10</td></tr><tr><td>300</td><td>0.05</td></tr><tr><td>400</td><td>0.03</td></tr><tr><td>500</td><td>0.03</td></tr></table>	Epoch	Loss	0	0.32	100	0.20	200	0.10	300	0.05	400	0.03	500	0.03		
Epoch	Loss																	
0	0.32																	
100	0.20																	
200	0.10																	
300	0.05																	
400	0.03																	
500	0.03																	
0.5	560	 <p>The graph shows the Mean Squared Error (MSE) loss over 1000 epochs for a learning rate of 0.5. The loss starts at approximately 0.38 and decreases very rapidly, reaching about 0.01 by epoch 100, and then continues to decrease very slowly, reaching about 0.005 by epoch 1000. The curve is smooth and shows a slight inflection point around epoch 100.</p> <table><caption>Approximate data points for learning rate 0.5</caption><tr><th>Epoch</th><th>Loss</th></tr><tr><td>0</td><td>0.38</td></tr><tr><td>100</td><td>0.01</td></tr><tr><td>200</td><td>0.005</td></tr><tr><td>400</td><td>0.005</td></tr><tr><td>600</td><td>0.005</td></tr><tr><td>800</td><td>0.005</td></tr><tr><td>1000</td><td>0.005</td></tr></table>	Epoch	Loss	0	0.38	100	0.01	200	0.005	400	0.005	600	0.005	800	0.005	1000	0.005
Epoch	Loss																	
0	0.38																	
100	0.01																	
200	0.005																	
400	0.005																	
600	0.005																	
800	0.005																	
1000	0.005																	

• **Case 2: XOR**

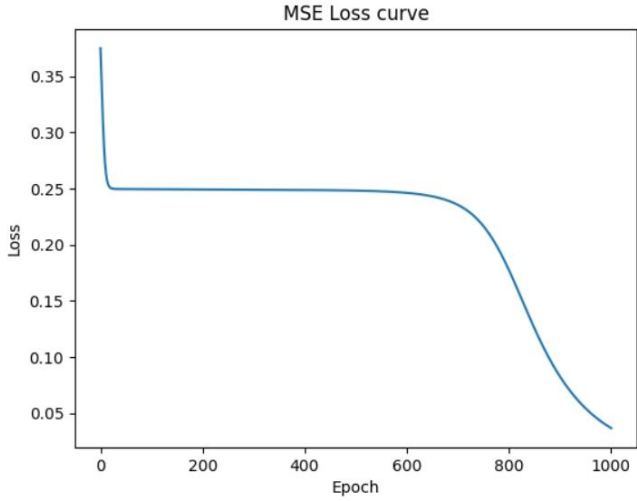
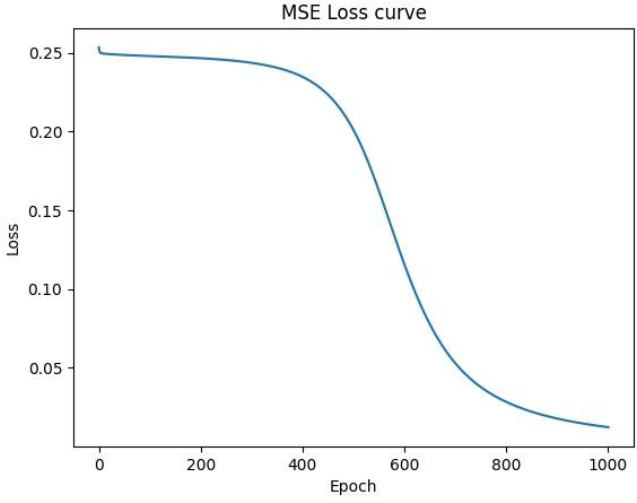
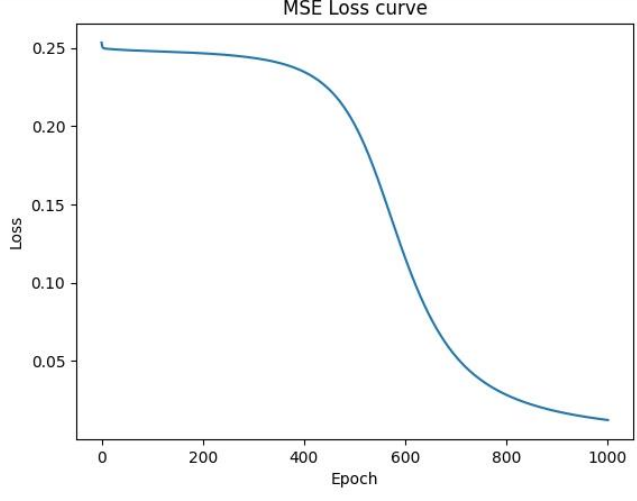
learning_rate	Epochs to acc=100%	Loss curve
0.1	916	 <p>The graph shows the Mean Squared Error (MSE) loss over 1000 epochs for a learning rate of 0.1. The y-axis represents the loss, ranging from 0.05 to 0.35. The x-axis represents the epoch, ranging from 0 to 1000. The loss starts at approximately 0.37, drops sharply to about 0.25 by epoch 50, and then remains relatively flat until epoch 600. After epoch 600, the loss decreases more gradually, reaching approximately 0.04 by epoch 1000.</p>
0.01	9100	 <p>The graph shows the Mean Squared Error (MSE) loss over 10000 epochs for a learning rate of 0.01. The y-axis represents the loss, ranging from 0.05 to 0.35. The x-axis represents the epoch, ranging from 0 to 10000. The loss starts at approximately 0.37, drops sharply to about 0.25 by epoch 500, and then remains relatively flat until epoch 6000. After epoch 6000, the loss decreases more gradually, reaching approximately 0.04 by epoch 10000.</p>
0.5	188	 <p>The graph shows the Mean Squared Error (MSE) loss over 500 epochs for a learning rate of 0.5. The y-axis represents the loss, ranging from 0.00 to 0.35. The x-axis represents the epoch, ranging from 0 to 500. The loss starts at approximately 0.37, drops sharply to about 0.25 by epoch 50, and then remains relatively flat until epoch 150. After epoch 150, the loss decreases more gradually, reaching approximately 0.01 by epoch 500.</p>

B. Try different numbers of hidden units

- Case 1: Linear

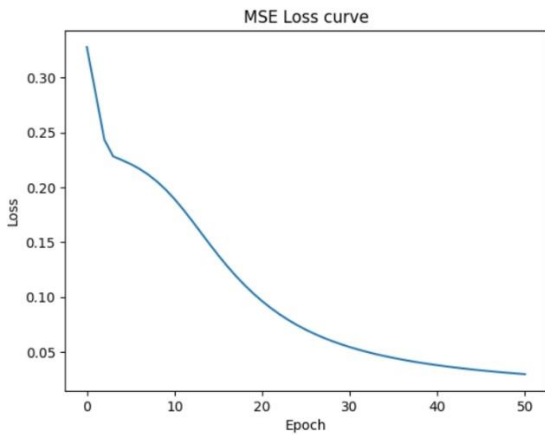
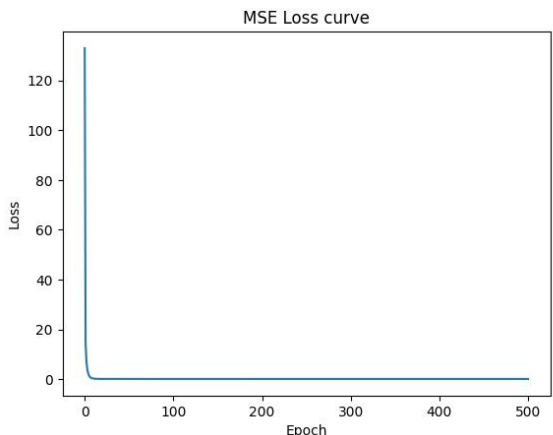
Hidden layers	Epochs to acc=100%	Loss curve
[2, 4, 4, 1]	38	 <p>The graph shows the Mean Squared Error (MSE) loss over 50 epochs for a neural network with 2 input, 4 hidden, 4 hidden, and 1 output units. The loss starts at approximately 0.32 and decreases steadily to about 0.03 by epoch 50. The curve is smooth, indicating good convergence.</p>
[2, 6, 6, 1]	33	 <p>The graph shows the MSE loss over 50 epochs for a neural network with 2 input, 6 hidden, 6 hidden, and 1 output units. The loss starts at approximately 0.28 and decreases to about 0.03 by epoch 50. The curve shows some initial fluctuations but then smooths out, indicating convergence.</p>
[2, 10, 10, 1]	36	 <p>The graph shows the MSE loss over 50 epochs for a neural network with 2 input, 10 hidden, 10 hidden, and 1 output units. The loss starts at approximately 0.35 and decreases to about 0.03 by epoch 50. The curve shows significant initial fluctuations, particularly in the first 15 epochs, before settling into a smooth decline.</p>

- **Case 2: XOR**

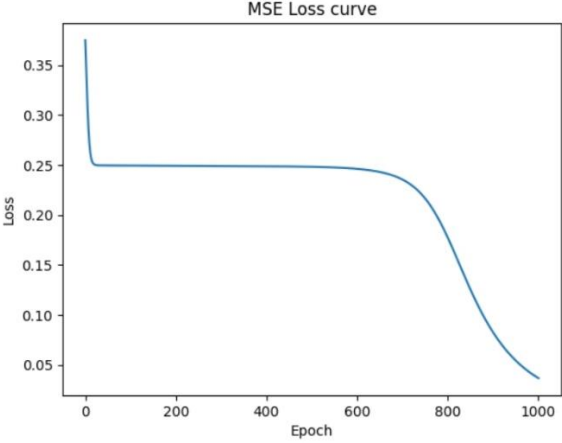
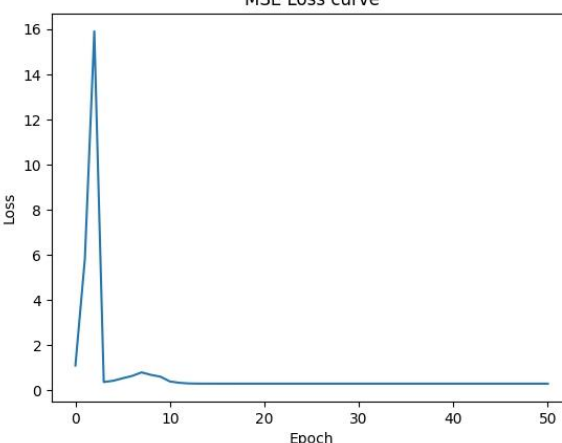
Hidden layers	Epochs to acc=100%	Loss curve
[2, 4, 4, 1]	916	 <p>The graph shows the Mean Squared Error (MSE) loss over 1000 epochs for a neural network with 2 input, 4 hidden, 4 hidden, and 1 output nodes. The loss starts at approximately 0.35, drops sharply to about 0.25 by epoch 50, and then remains relatively flat until epoch 600. After epoch 600, the loss decreases more rapidly, reaching approximately 0.04 by epoch 1000.</p>
[2, 6, 6, 1]	661	 <p>The graph shows the Mean Squared Error (MSE) loss over 1000 epochs for a neural network with 2 input, 6 hidden, 6 hidden, and 1 output nodes. The loss starts at approximately 0.25, remains relatively flat until epoch 400, and then decreases steadily to approximately 0.02 by epoch 1000.</p>
[2, 10, 10, 1]	587	 <p>The graph shows the Mean Squared Error (MSE) loss over 1000 epochs for a neural network with 2 input, 10 hidden, 10 hidden, and 1 output nodes. The loss starts at approximately 0.25, remains relatively flat until epoch 400, and then decreases steadily to approximately 0.02 by epoch 1000.</p>

C. Try without activation functions

- Case 1: Linear

With sigmoid	Without sigmoid
Learning_rate = 0.1	Learning_rate = 0.0001 (if use 0.1, cannot converge)
Epoch: 0 loss:0.327794 acc:58.00% Epoch: 5 loss:0.221193 acc:58.00% Epoch: 10 loss:0.189331 acc:77.00% Epoch: 15 loss:0.138072 acc:92.00% Epoch: 20 loss:0.096261 acc:96.00% Epoch: 25 loss:0.070163 acc:99.00% Epoch: 30 loss:0.054393 acc:99.00% Epoch: 35 loss:0.044466 acc:99.00% Epoch: 40 loss:0.037827 acc:100.00% Epoch: 45 loss:0.033126 acc:100.00% Epoch: 50 loss:0.029633 acc:100.00%	Epoch: 0 loss:132.981598 acc:1.00% Epoch: 50 loss:0.131182 acc:82.00% Epoch: 100 loss:0.125672 acc:84.00% Epoch: 150 loss:0.123602 acc:84.00% Epoch: 200 loss:0.122839 acc:86.00% Epoch: 250 loss:0.122562 acc:86.00% Epoch: 300 loss:0.122462 acc:87.00% Epoch: 350 loss:0.122426 acc:87.00% Epoch: 400 loss:0.122413 acc:87.00% Epoch: 450 loss:0.122409 acc:87.00% Epoch: 500 loss:0.122407 acc:87.00%
	

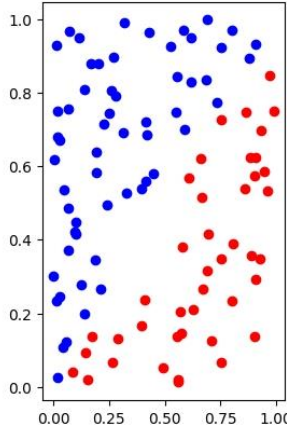
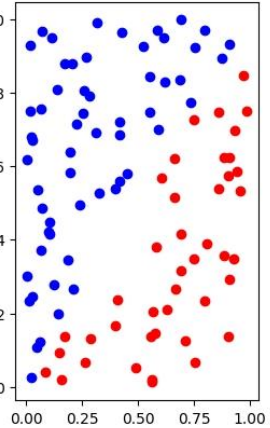
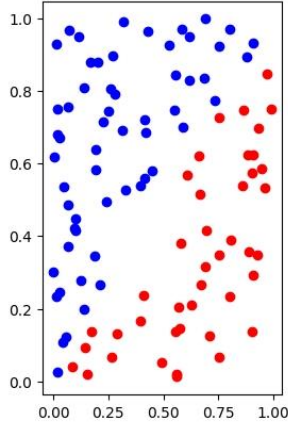
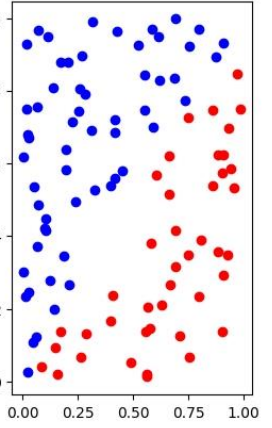
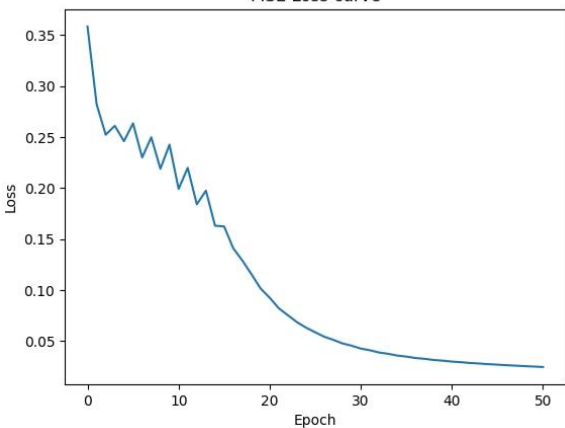
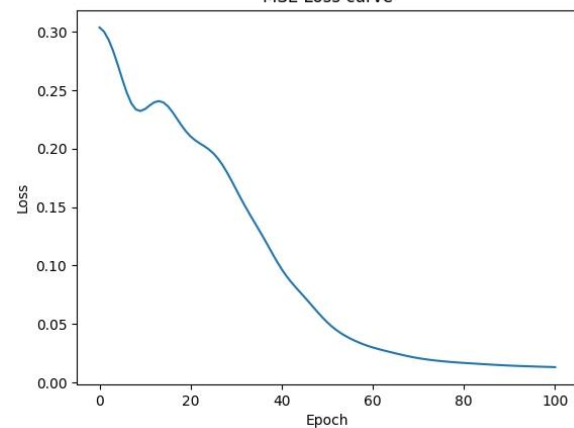
• **Case 2: XOR**

With sigmoid	Without sigmoid
Learning_rate = 0.1	Learning_rate = 0.01
	(if use 0.1, cannot converge)
Epoch: 0 loss:0.374859 acc:47.62% Epoch: 100 loss:0.249373 acc:52.38% Epoch: 200 loss:0.249138 acc:52.38% Epoch: 300 loss:0.248945 acc:52.38% Epoch: 400 loss:0.248721 acc:52.38% Epoch: 500 loss:0.248191 acc:52.38% Epoch: 600 loss:0.246140 acc:57.14% Epoch: 700 loss:0.235422 acc:71.43% Epoch: 800 loss:0.177223 acc:90.48% Epoch: 900 loss:0.082453 acc:95.24% Epoch:1000 loss:0.036723 acc:100.00%	Epoch: 0 loss:1.093362 acc:52.38% Epoch: 5 loss:0.528004 acc:52.38% Epoch: 10 loss:0.386753 acc:66.67% Epoch: 15 loss:0.288937 acc:33.33% Epoch: 20 loss:0.288715 acc:33.33% Epoch: 25 loss:0.288714 acc:33.33% Epoch: 30 loss:0.288714 acc:33.33% Epoch: 35 loss:0.288714 acc:33.33% Epoch: 40 loss:0.288714 acc:33.33% Epoch: 45 loss:0.288714 acc:33.33% Epoch: 50 loss:0.288714 acc:33.33%
	

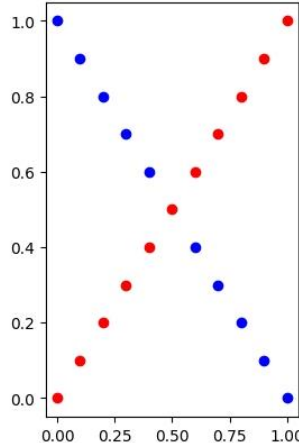
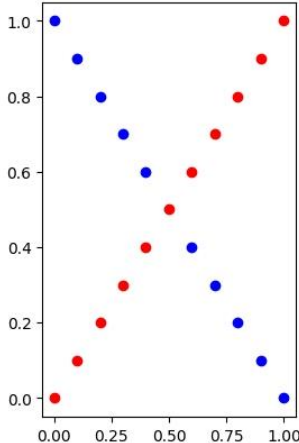
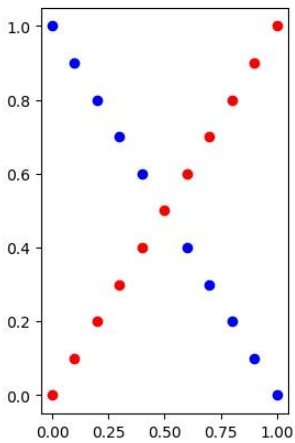
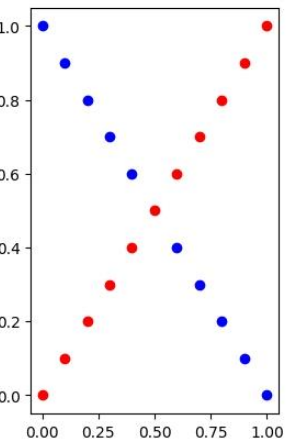
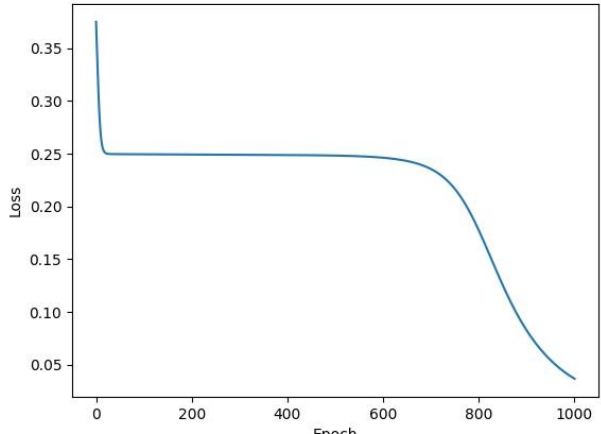
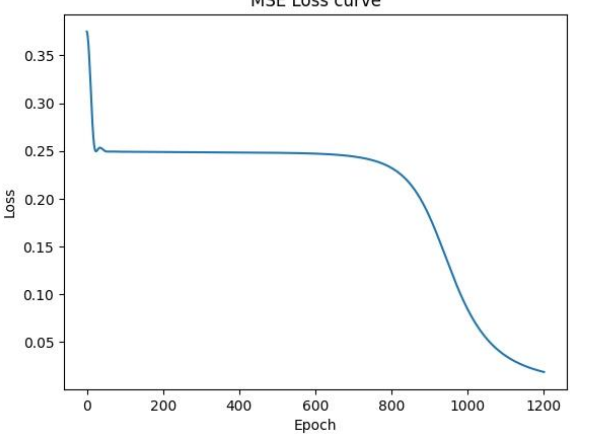
5. Extra

A. Implement different optimizers

- Case 1: Linear

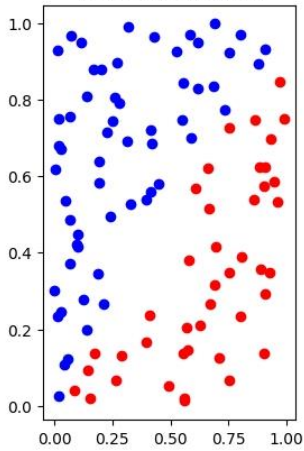
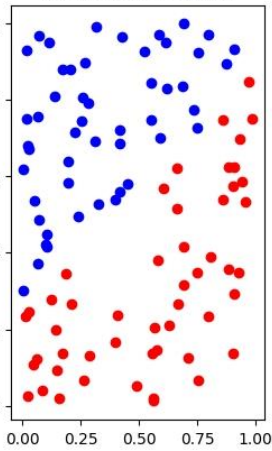
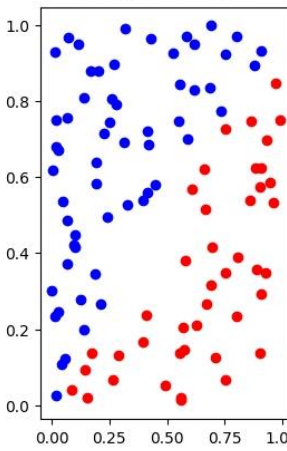
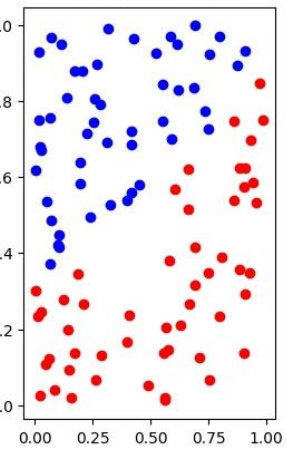
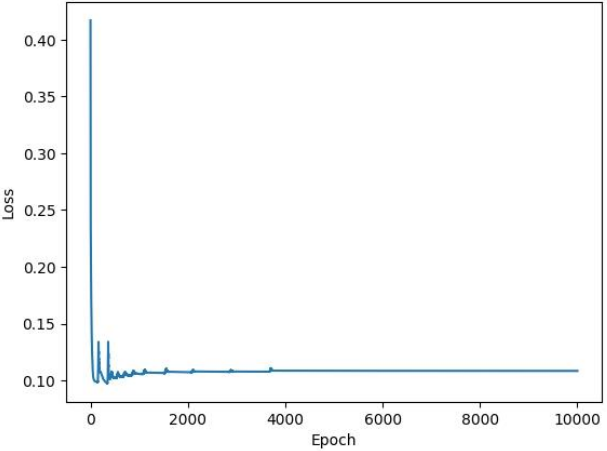
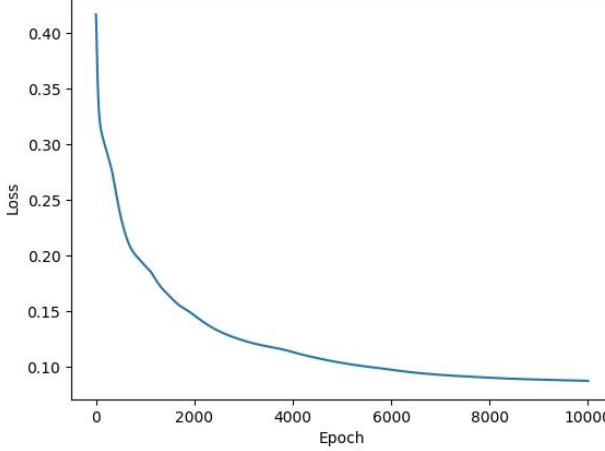
Sigmoid + SGD	Sigmoid + momentum
Epoch: 0 loss:0.358400 acc:58.00% Epoch: 5 loss:0.263512 acc:58.00% Epoch: 10 loss:0.199258 acc:70.00% Epoch: 15 loss:0.162447 acc:67.00% Epoch: 20 loss:0.092607 acc:90.00% Epoch: 25 loss:0.058728 acc:94.00% Epoch: 30 loss:0.042800 acc:97.00% Epoch: 35 loss:0.034942 acc:98.00% Epoch: 40 loss:0.030036 acc:100.00% Epoch: 45 loss:0.027053 acc:100.00% Epoch: 50 loss:0.024744 acc:100.00%	Epoch: 0 loss:0.303472 acc:58.00% Epoch: 10 loss:0.233734 acc:74.00% Epoch: 20 loss:0.210273 acc:71.00% Epoch: 30 loss:0.164940 acc:83.00% Epoch: 40 loss:0.096741 acc:95.00% Epoch: 50 loss:0.051386 acc:96.00% Epoch: 60 loss:0.029887 acc:98.00% Epoch: 70 loss:0.020754 acc:99.00% Epoch: 80 loss:0.016687 acc:100.00% Epoch: 90 loss:0.014425 acc:100.00% Epoch: 100 loss:0.013088 acc:100.00%
<div>Ground truth</div>  <div>Predict result</div> 	<div>Ground truth</div>  <div>Predict result</div> 
<div>MSE Loss curve</div> 	<div>MSE Loss curve</div> 

- **Case 2: XOR**

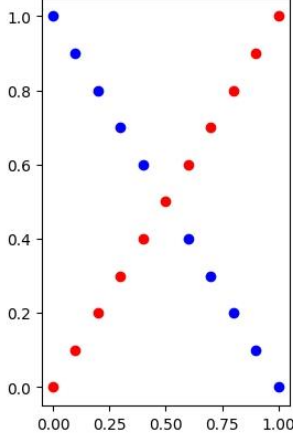
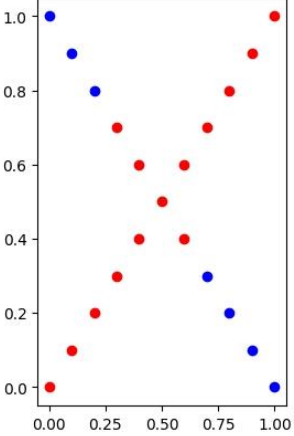
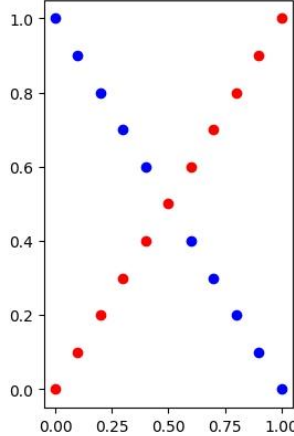
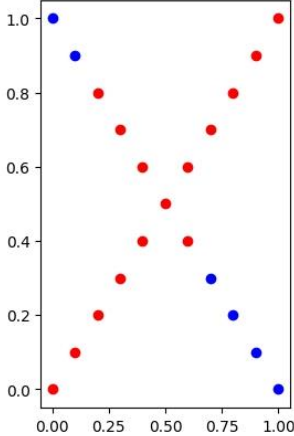
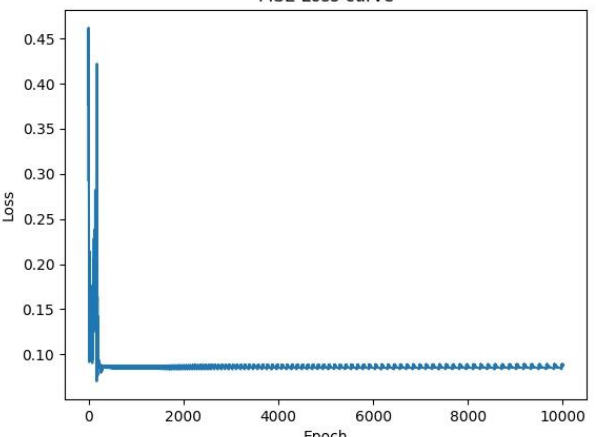
Sigmoid + SGD	Sigmoid + momentum
Epoch: 0 loss:0.374859 acc:47.62% Epoch: 100 loss:0.249373 acc:52.38% Epoch: 200 loss:0.249138 acc:52.38% Epoch: 300 loss:0.248945 acc:52.38% Epoch: 400 loss:0.248721 acc:52.38% Epoch: 500 loss:0.248191 acc:52.38% Epoch: 600 loss:0.246140 acc:57.14% Epoch: 700 loss:0.235422 acc:71.43% Epoch: 800 loss:0.177223 acc:90.48% Epoch: 900 loss:0.082453 acc:95.24% Epoch:1000 loss:0.036723 acc:100.00%	Epoch: 0 loss:0.374859 acc:47.62% Epoch: 120 loss:0.249060 acc:52.38% Epoch: 240 loss:0.248729 acc:52.38% Epoch: 360 loss:0.248454 acc:52.38% Epoch: 480 loss:0.248127 acc:52.38% Epoch: 600 loss:0.247180 acc:47.62% Epoch: 720 loss:0.242926 acc:66.67% Epoch: 840 loss:0.219451 acc:90.48% Epoch: 960 loss:0.120322 acc:95.24% Epoch:1080 loss:0.041479 acc:100.00% Epoch:1200 loss:0.018676 acc:100.00%
<div> <div>Ground truth</div>  </div> <div> <div>Predict result</div>  </div>	<div> <div>Ground truth</div>  </div> <div> <div>Predict result</div>  </div>
<div>MSE Loss curve</div> 	<div>MSE Loss curve</div> 

B. Implement different activation functions

- Case 1: Linear

Relu + SGD	Relu + momentum
<pre>Epoch: 0 loss:0.417119 acc:42.00% Epoch:1000 loss:0.105709 acc:88.00% Epoch:2000 loss:0.107252 acc:88.00% Epoch:3000 loss:0.107806 acc:89.00% Epoch:4000 loss:0.108537 acc:87.00% Epoch:5000 loss:0.108462 acc:87.00% Epoch:6000 loss:0.108420 acc:87.00% Epoch:7000 loss:0.108394 acc:87.00% Epoch:8000 loss:0.108376 acc:87.00% Epoch:9000 loss:0.108364 acc:88.00% Epoch:10000 loss:0.108354 acc:88.00%</pre>	<pre>Epoch: 0 loss:0.417119 acc:42.00% Epoch:1000 loss:0.190789 acc:72.00% Epoch:2000 loss:0.146076 acc:76.00% Epoch:3000 loss:0.123402 acc:80.00% Epoch:4000 loss:0.112995 acc:83.00% Epoch:5000 loss:0.103259 acc:85.00% Epoch:6000 loss:0.097123 acc:87.00% Epoch:7000 loss:0.092489 acc:88.00% Epoch:8000 loss:0.089913 acc:89.00% Epoch:9000 loss:0.088215 acc:89.00% Epoch:10000 loss:0.087129 acc:89.00%</pre>
<div><div>Ground truth</div><div>Predict result</div></div>	<div><div>Ground truth</div><div>Predict result</div></div>
<div>MSE Loss curve</div> 	<div>MSE Loss curve</div> 

- **Case 2: XOR**

Relu + SGD	Relu + momentum
Epoch: 0 loss:0.376679 acc:52.38% Epoch:1000 loss:0.086383 acc:90.48% Epoch:2000 loss:0.087548 acc:90.48% Epoch:3000 loss:0.085431 acc:90.48% Epoch:4000 loss:0.085886 acc:90.48% Epoch:5000 loss:0.085632 acc:90.48% Epoch:6000 loss:0.085172 acc:90.48% Epoch:7000 loss:0.088785 acc:85.71% Epoch:8000 loss:0.086329 acc:90.48% Epoch:9000 loss:0.084738 acc:90.48% Epoch:10000 loss:0.088944 acc:85.71%	Epoch: 0 loss:0.376679 acc:52.38% Epoch: 100 loss loss:0.195015 acc:66.67% Epoch: 200 loss:0.183786 acc:71.43% Epoch: 300 loss:0.184458 acc:71.43% Epoch: 400 loss:0.183645 acc:71.43% Epoch: 500 loss:0.182706 acc:71.43% Epoch: 600 loss:0.473566 acc:52.38% Epoch: 700 loss:0.144633 acc:80.95% Epoch: 800 loss:0.332644 acc:57.14% Epoch: 900 loss:0.202241 acc:80.95% Epoch:1000 loss:0.201889 acc:80.95%
<div> <div>Ground truth</div>  </div> <div> <div>Predict result</div>  </div>	<div> <div>Ground truth</div>  </div> <div> <div>Predict result</div>  </div>
<div>MSE Loss curve</div> 	<div>MSE Loss curve</div> 