NYCU DL
Lab1 - Backpropagation

TA 陳敬中 Bill

July 3, 2025

Outline

- Lab Objectives
- Scoring Criteria
- Time Schedule

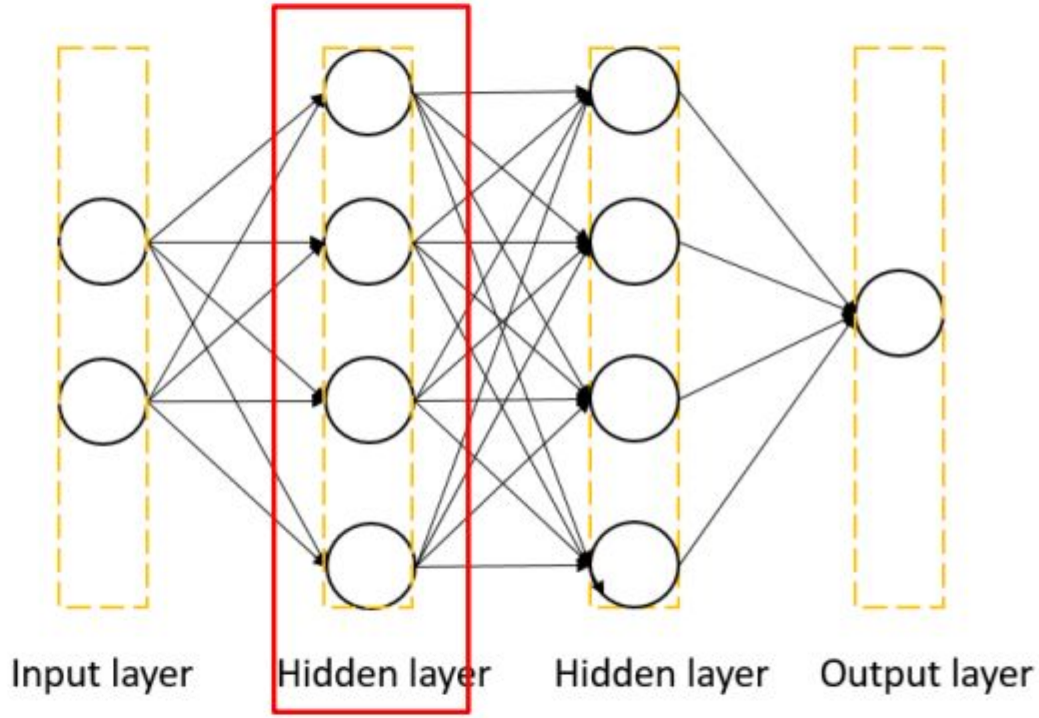
Lab Objectives

- **Implement a Simple Neural Network**: Build a fully-connected NN with **two hidden** layers using **only Numpy**
- **Master Backpropagation:** Implement a backpropagation algorithm to calculate gradients and update model weights
- **Visualize Learning:** Plot the learning curve and show the prediction-vs-ground_truth figure
- **Evaluation:** Calculate and print the accuracy of your trained model

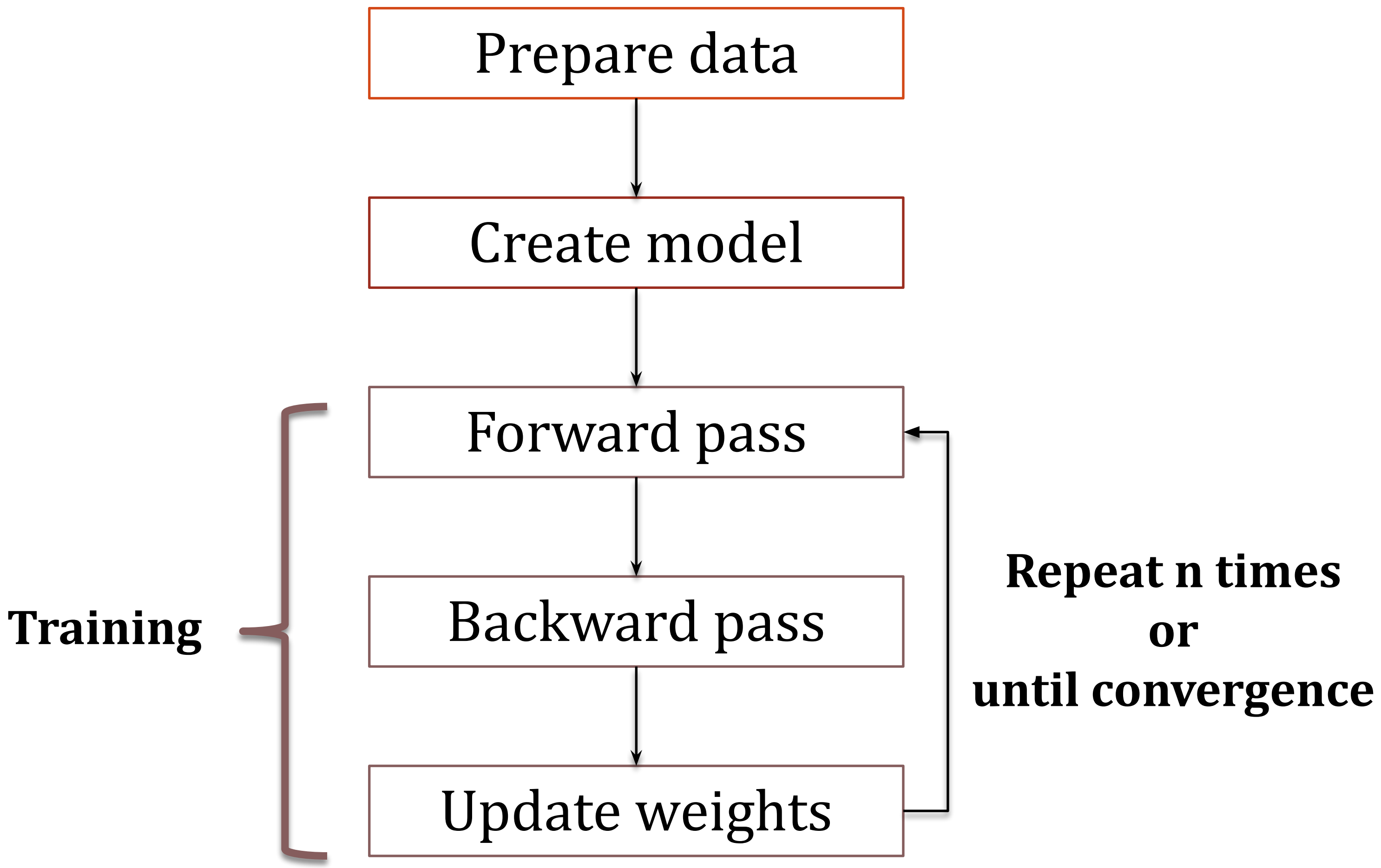
NOTE: You are NOT allowed to use DL frameworks like PyTorch or TensorFlow

Lab Description

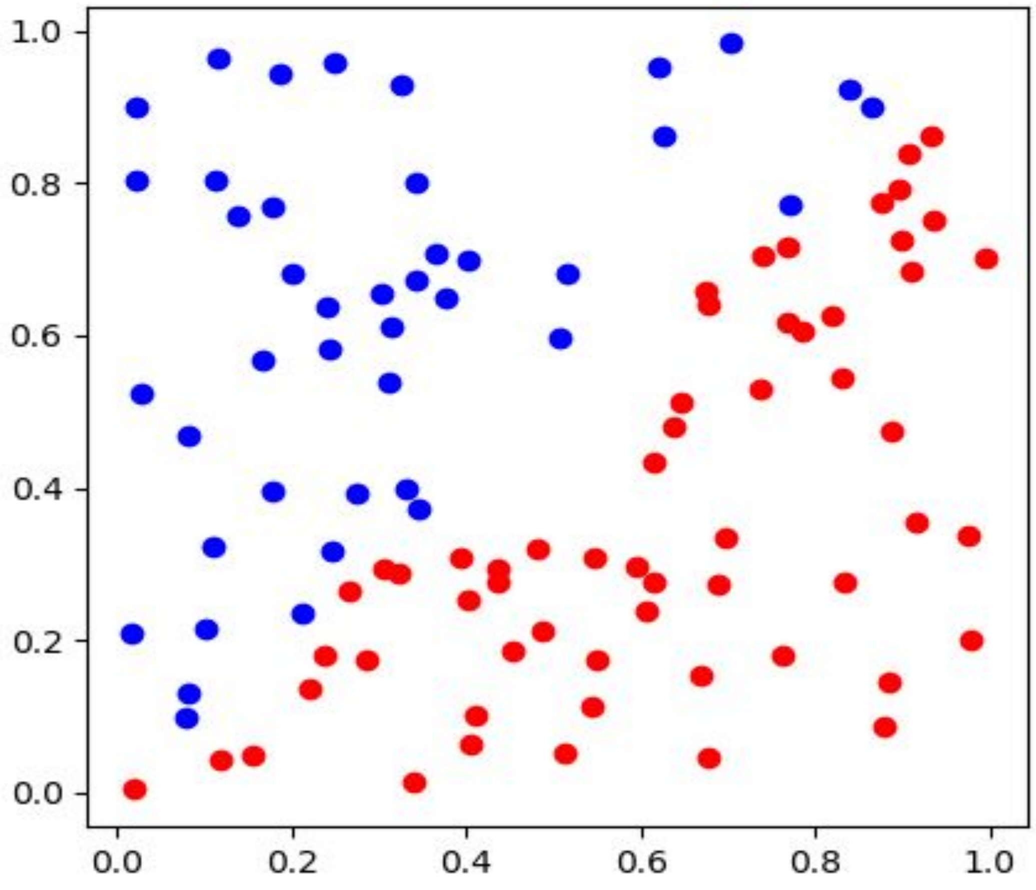
- Each layer should contain
  - One transformation (e.g., Linear, CNN, …)
  - One activation function (e.g., sigmoid, tanh, relu, …)



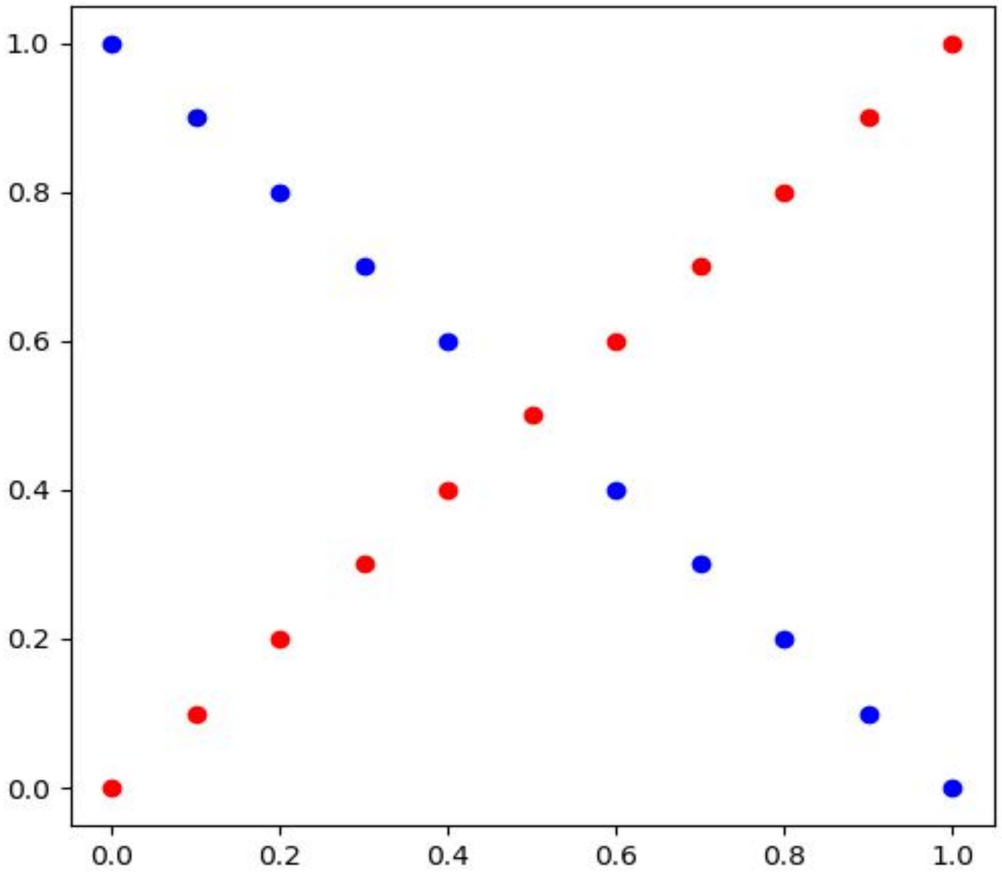
Lab Description – Flowchart



Lab Description - Data



generate_linear()



generate_XOR_easy()

Data Generation

- Do **NOT** overwrite these functions
- Training and Testing with the same data
- For the linear dataset, train and test with n=100

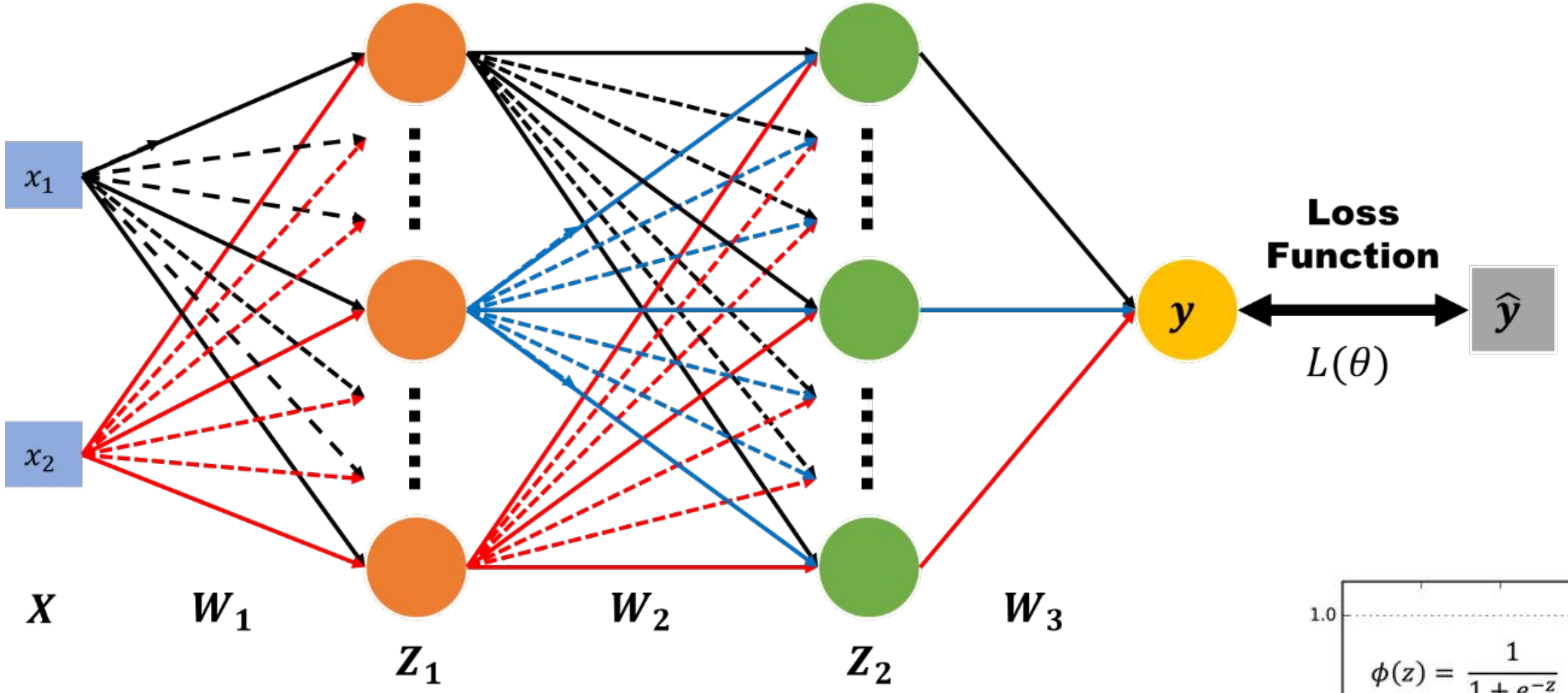
Function usage

```
x, y = generate_linear(n=100)
x, y = generate_XOR_easy()
```

```
def generate_linear(n=100):
    import numpy as np
    pts = np.random.uniform(0, 1, (n, 2))
    inputs = []
    labels = []
    for pt in pts:
        inputs.append([pt[0], pt[1]])
        distance = (pt[0]-pt[1])/1.414
        if pt[0] > pt[1]:
            labels.append(0)
        else:
            labels.append(1)
    return np.array(inputs), np.array(labels).reshape(n, 1)
```

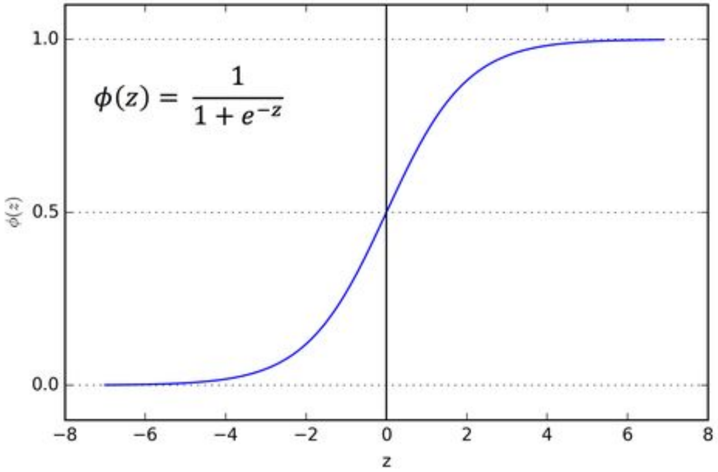
```
def generate_XOR_easy():
    import numpy as np
    inputs = []
    labels = []

    for i in range(11):
        inputs.append([0.1*i, 0.1*i])
        labels.append(0)

        if 0.1*i == 0.5:
            continue

        inputs.append([0.1*i, 1-0.1*i])
        labels.append(1)

    return np.array(inputs), np.array(labels).reshape(21, 1)
```

Lab Description – Forward

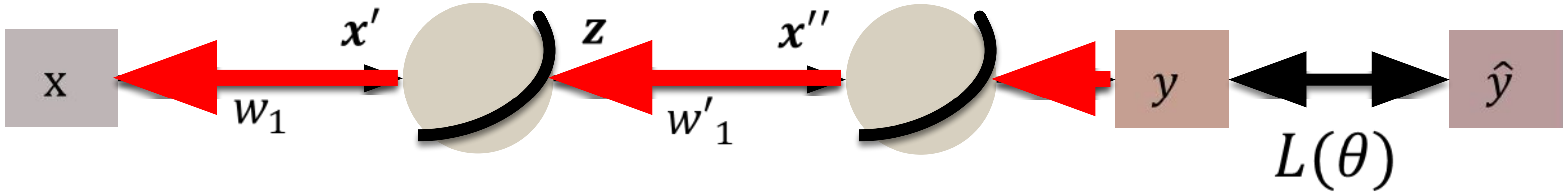


$$Z_1 = \sigma(XW_1) \qquad Z_2 = \sigma(Z_1W_2) \qquad y = \sigma(Z_2W_3)$$

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$



Lab Description – Backward



$$\boldsymbol{x' = x w_1} \qquad \boldsymbol{z = \sigma(x')} \qquad \boldsymbol{x'' = z w'_1} \qquad \boldsymbol{y = \sigma(x'')}$$

Chain rule

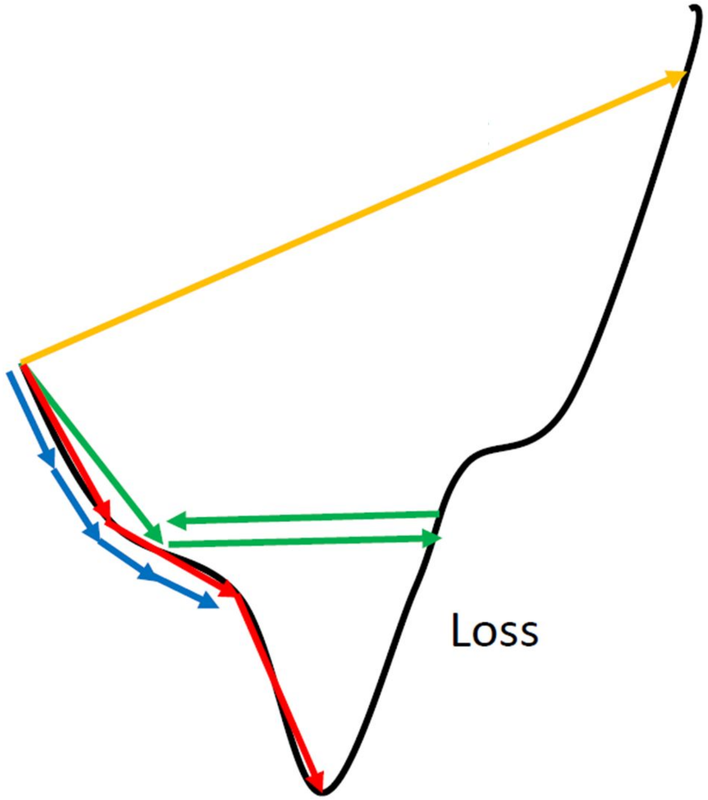
$$y = g(x) \qquad z = h(y)$$

$$\mathbf{x} \xrightarrow{\mathbf{g}()} \boldsymbol{y} \xrightarrow{\boldsymbol{h}()} \boldsymbol{z} \qquad \frac{dz}{dx} = \frac{dz}{\cancel{dy}} \frac{\cancel{dy}}{dx}$$

$$\begin{aligned}
\frac{\partial L(\theta)}{\partial w_1} &= \frac{\partial y}{\partial w_1} \frac{\partial L(\theta)}{\partial y} \\
&= \frac{\partial x''}{\partial w_1} \frac{\partial y}{\partial x''} \frac{\partial L(\theta)}{\partial y} \\
&= \frac{\partial z}{\partial w_1} \frac{\partial x''}{\partial z} \frac{\partial y}{\partial x''} \frac{\partial L(\theta)}{\partial y} \\
&= \frac{\partial x'}{\partial w_1} \frac{\partial z}{\partial x'} \frac{\partial x''}{\partial z} \frac{\partial y}{\partial x''} \frac{\partial L(\theta)}{\partial y}
\end{aligned}$$

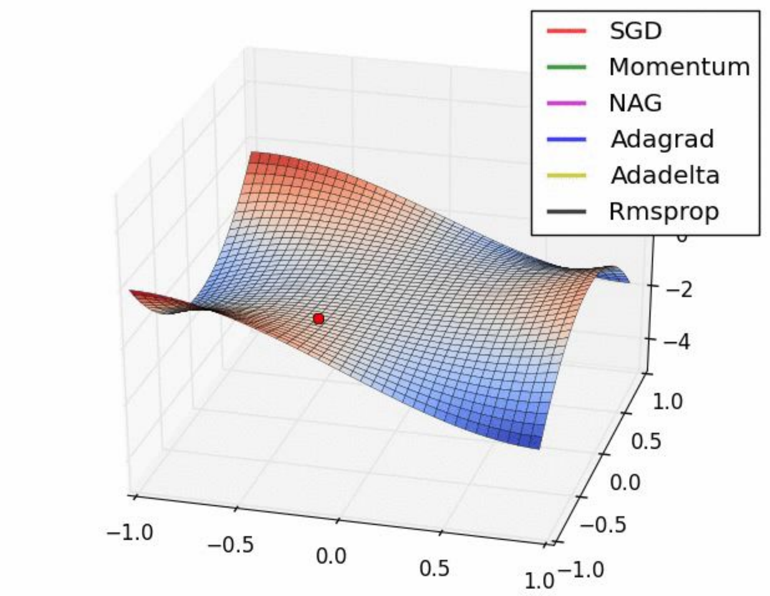
Lab Description – Gradient descent

Network Parameters $\theta = \{w_1, w_2, w_3, w_4, \cdots\}$



$$\theta^1 = \theta^0 - \rho \nabla L(\theta^0)$$
$$\theta^2 = \theta^1 - \rho \nabla L(\theta^1)$$
$$\theta^3 = \theta^2 - \rho \nabla L(\theta^2)$$

ρ : Learning rate



http://www.denizyuret.com/2015/03/alec-radfords-animations-for.html

Lab Description - Prediction

- During training, print the loss

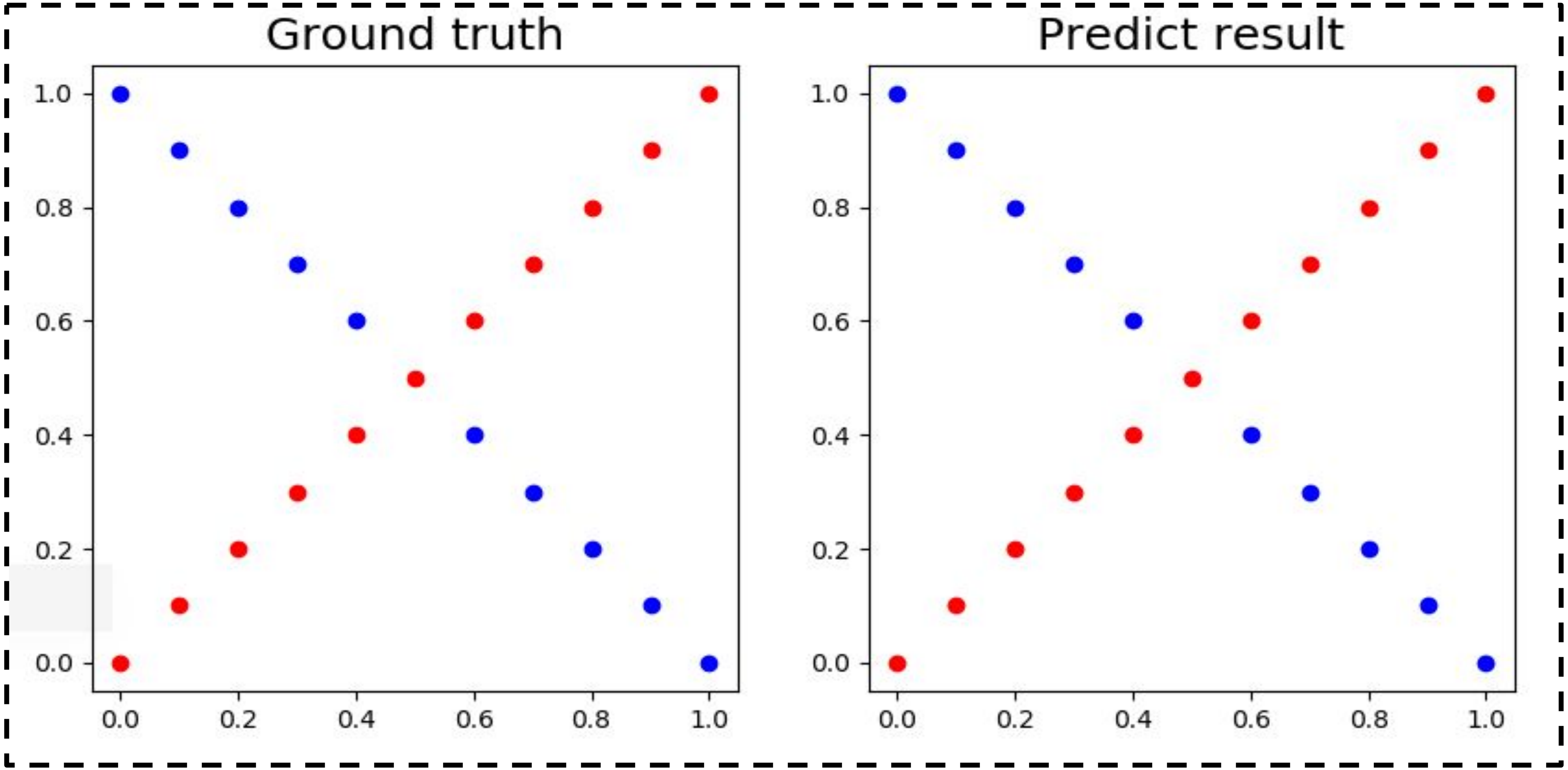
```
epoch 10000 loss : 0.16234523253277644
epoch 15000 loss : 0.2524336634177614
epoch 20000 loss : 0.1590783047540092
epoch 25000 loss : 0.22099447030234853
epoch 30000 loss : 0.3292173477217561
epoch 35000 loss : 0.40406233282426085
epoch 40000 loss : 0.43052897480298924
epoch 45000 loss : 0.4207525735586605
epoch 50000 loss : 0.3934759509342479
epoch 55000 loss : 0.3615008372106921
epoch 60000 loss : 0.33077879872648525
epoch 65000 loss : 0.30333537090819584
epoch 70000 loss : 0.2794858089741792
epoch 75000 loss : 0.25892812312991587
epoch 80000 loss : 0.24119780823897027
epoch 85000 loss : 0.22583656353511342
epoch 90000 loss : 0.21244497028971704
epoch 95000 loss : 0.2006912468389013
```

- During testing, show the predictions and the accuracy

```
Iter91 |     Ground truth: 1.0 |     prediction: 0.99943 |
Iter92 |     Ground truth: 1.0 |     prediction: 0.99987 |
Iter93 |     Ground truth: 1.0 |     prediction: 0.99719 |
Iter94 |     Ground truth: 1.0 |     prediction: 0.99991 |
Iter95 |     Ground truth: 0.0 |     prediction: 0.00013 |
Iter96 |     Ground truth: 1.0 |     prediction: 0.77035 |
Iter97 |     Ground truth: 1.0 |     prediction: 0.98981 |
Iter98 |     Ground truth: 1.0 |     prediction: 0.99337 |
Iter99 |     Ground truth: 0.0 |     prediction: 0.20275 |
loss=0.03844 accuracy=100.00%
```

Lab Description - Prediction

- Visualize the predictions and ground truth at the end of the training process



Scoring Criteria

- Write clearly your what, why, and how.
- Visualize your results
- Strictly follow the spec

- **<u>Report Spec:</u>**

1. Introduction (5%)
2. Implementation Details (25%):
   A. Network Architecture
   B. Activation Functions
   C. Backpropagation
   D. Extra Implementation
3. Experimental Results (40%)
   A. Screenshot and comparison figure (5%)
   B. Show the accuracy of your prediction (30% * accuracy)
   C. Learning curve (loss-epoch curve) (5%)
   D. Anything you want to present
4. Discussions (21%)
   A. Try different learning rates
   B. Try different numbers of hidden units
   C. Try without activation functions
   D. Extra Implementation Discussions
5. Questions (9%)
   A. What are the purposes of activation functions? (3%)
   B. What if the learning rate is too large or too small? (3%)
   C. What are the purposes of weights and biases in a neural network? (3%)
6. Bonus (10%)
   A. Optimizers. (5%)
   B. Activation functions. (5%)

Important Date

- Report Submission Deadline: 7/10 (Thu.) 23:59
- Zip all files in one file
  - Report (report.pdf)
  - Source code
- Do NOT submit model weights and figures
- Name it like「DL_LAB1_yourstudentID_name.zip」
  - E.g., 「DL_LAB1_313551157_陳敬中.zip」
- If there are any format errors in your files, or if you do not follow the requirements, you will receive a **10-point** penalty each