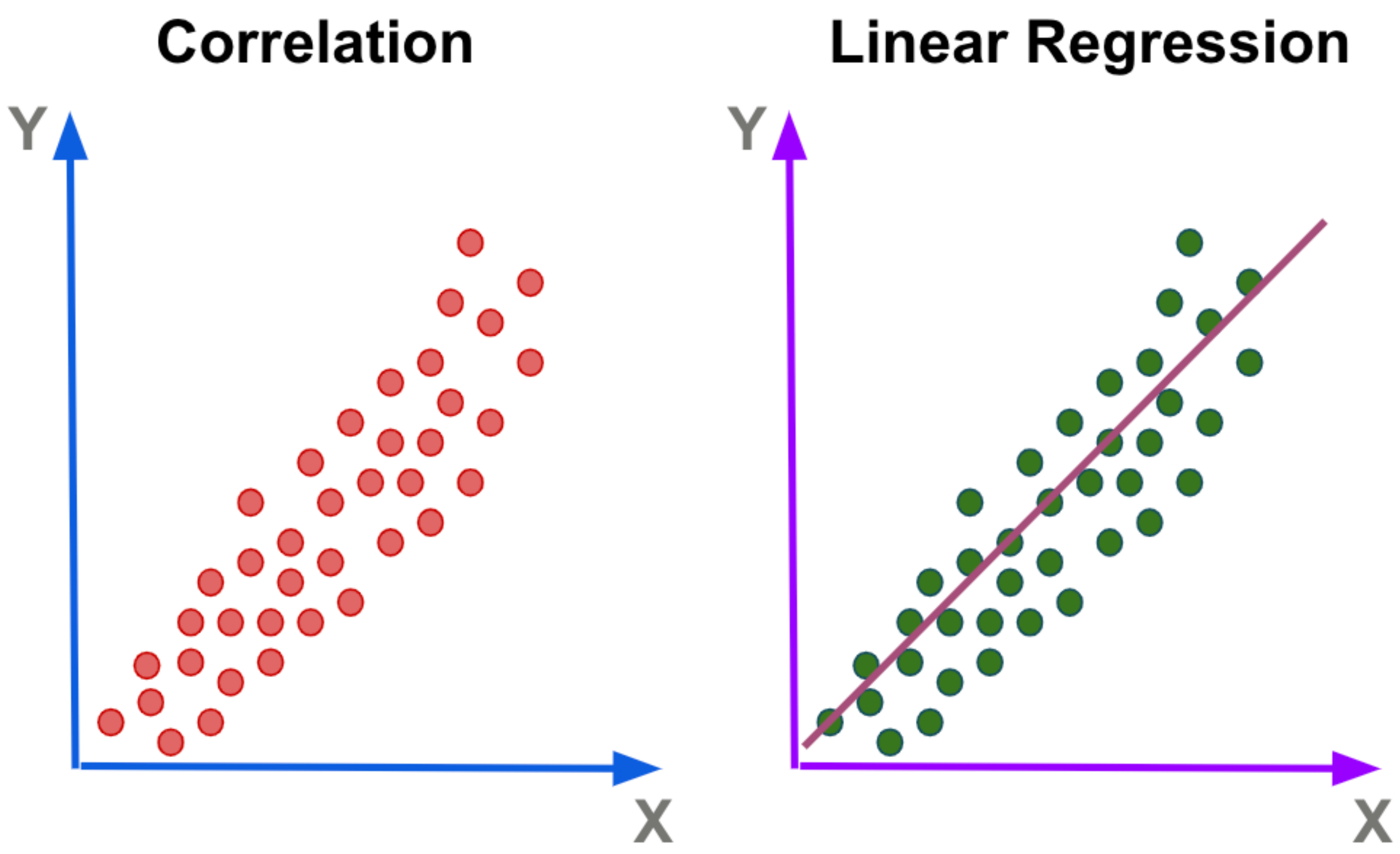
Week 2 MLP

Tutor:
Email:
Tutorial:
Code: https://github.com/Jinxu-Lin/COMP5329

Multilayer Neural Network

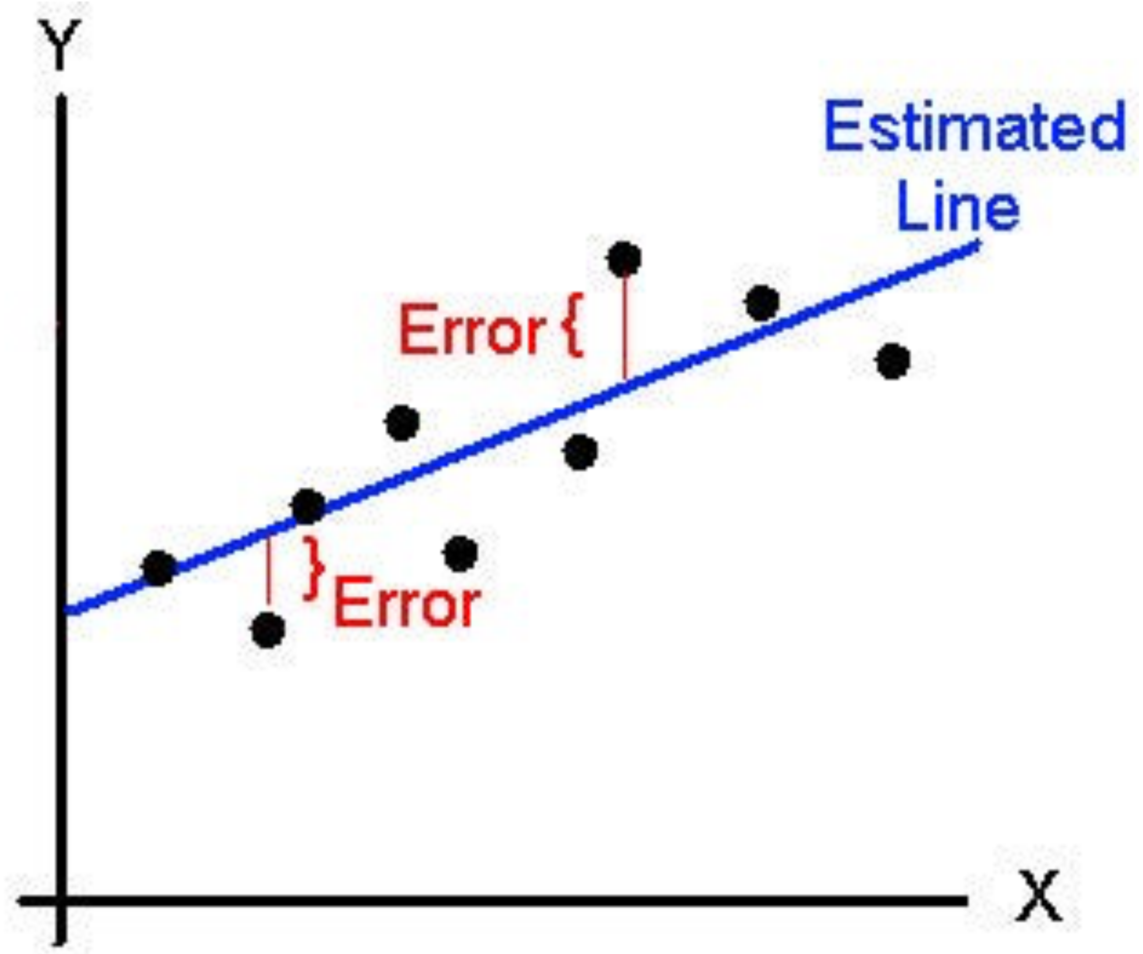
Linear Regression Model

- $L(X) = W * X + b$
- $X \in R^{N \times M}$, N samples, each sample has M features



Error Function for Linear Regression

- $J = \frac{1}{2N} || W * X + b - y ||^2$, with y is the target
- Objective: **Find** W so that J is minimized.
- Derivation: $\frac{\delta J}{\delta W} = 2X^T XW - 2X^T t$
- In order to find the optimal value for W, we will set the $\frac{\delta J}{\delta W} = 0$, we achieve:
- $2X^T XW - 2X^T t = 0$ <=> $W* = (X^T X)^{-1} X^T t$
- The solution that we found $W*$ is called the **Analytic solution**
- This solution can only be used if the solution is a **closed-form** solution



Perceptron Model

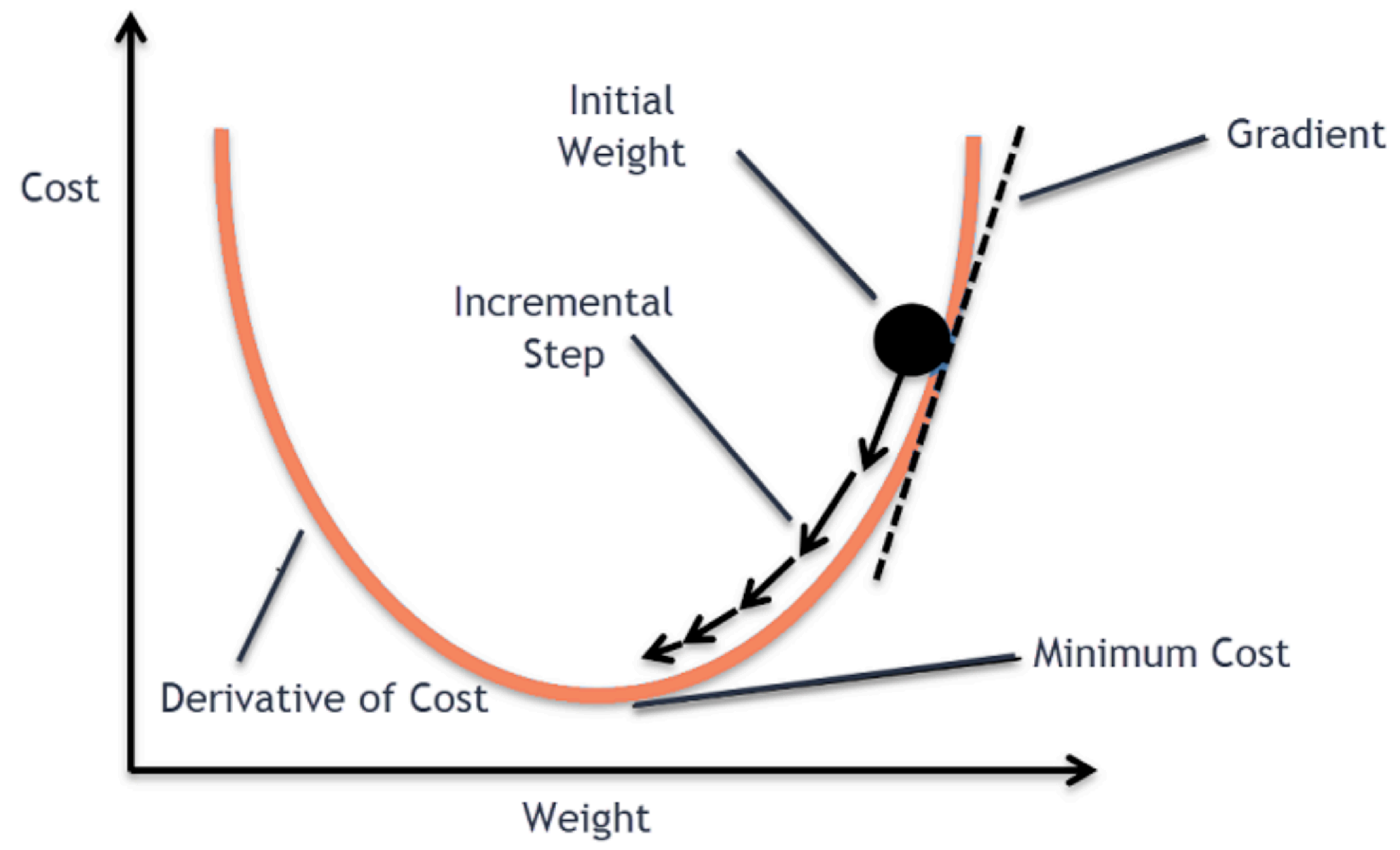
- $L(X) = \sigma(W * X + b)$
    - When $a \geq 0, \sigma(a) = +1$
    - When $a < 0, \sigma(a) = -1$
    - $\sigma(a)$ is the **non-linear** function
- Perceptron is a linear model wrapped with the **non-linear** function
- The Perceptron model is utilized for binary classification.

Error Function for Perceptron Model

- $E(W) = -\sum_n W * X_n * t_n$, where $L_n = \sigma(W * X_n + b)$
- Solution
  - Write the derivative equation for $E(W)$ regarding to W. => $\frac{\delta E(W)}{\delta W}$
    - Due to the non-linear operator σ, we **no longer** can have **closed-form of Analytic solution**
    - There are a number of ways to approximate the optimal solution, one of them is **Gradient Descent**

Gradient Descent

- Starting from a random point. We will update from step-to-step follwing the below equation:
- $W = W - \eta * \dfrac{\delta E(W)}{\delta W} X$, with η is the learning rate



2-layer Perceptron

- $L(X) = \sigma(W_2 * h(X) + b_2)$
    - $h(X) = \sigma(W_1 * X + b_1)$
    - $\sigma(a) = \dfrac{1}{1 + e^{-a}}$ is a non-linear function (sigmoid)/
    - σ can be any other non-linear functions.
    - MLP is a stack of linear models wrapped by a non-linear function.

Back Propagation

Back Propagation

- It is impossible to achieve the closed-form analytic solution now. We will also use **Gradient Descent** algorithm to optimize the W_1 and W_2.
- We can also not be able to apply the Gradient equation derived from Linear Regression model due to the **non-linear function** σ.
- We need some tools to calculate gradient for W_1 and W_2.

Back Propagation

- Steps
    - Output: $z_c = f(y_j * w_{jc}^{yz} + b_2)$
    - Loss function: $J = ||z_c - t||^2$, where t is the target
    - Based on gradient-descent, we want to find $\frac{\delta J}{\delta w_{ij}^{xy}}$, $\frac{\delta J}{\delta w_{ij}^{yz}}$, $\frac{\delta J}{\delta b_1}$ and $\frac{\delta J}{\delta b_2}$

Back Propagation

- 1. First objective is to find $\frac{\delta J}{\delta w_{ij}^{xy}}$
  - According to chain rule $\frac{\delta J}{\delta w_{ij}^{xy}} = \frac{\delta J}{\delta z_c} . \frac{\delta z_c}{\delta y_j} . \frac{\delta y_j}{w_{ij}^{xy}}$
    - $\frac{\delta J}{\delta z_c} = 2 * (z_c - t)$
    - $\frac{\delta z_c}{\delta y_j} = f'(y_j * w_{jc}^{yz} + b_2) * w_{jc}^{yz}$
    - $\frac{\delta y_j}{w_{ij}^{xy}} = x_i$

Back Propagation

- 1. First objective is to find $\frac{\delta J}{\delta w_{ij}^{xy}}$
  - According to chain rule $\frac{\delta J}{\delta w_{ij}^{xy}} = \frac{\delta J}{\delta z_c} . \frac{\delta z_c}{\delta y_j} . \frac{\delta y_j}{w_{ij}^{xy}}$
    - $\frac{\delta J}{\delta z_c} = 2 * (z_c - t), \frac{\delta z_c}{\delta y_j} = f'(y_j * w_{jc}^{yz} + b_2) * w_{jc}^{yz}, \frac{\delta y_j}{w_{ij}^{xy}} = x_i$
  - Final solution: $\frac{\delta J}{\delta w_{ij}} = \frac{\delta J}{\delta z_c} * f' * w_{jc}^{yz} * x_i$

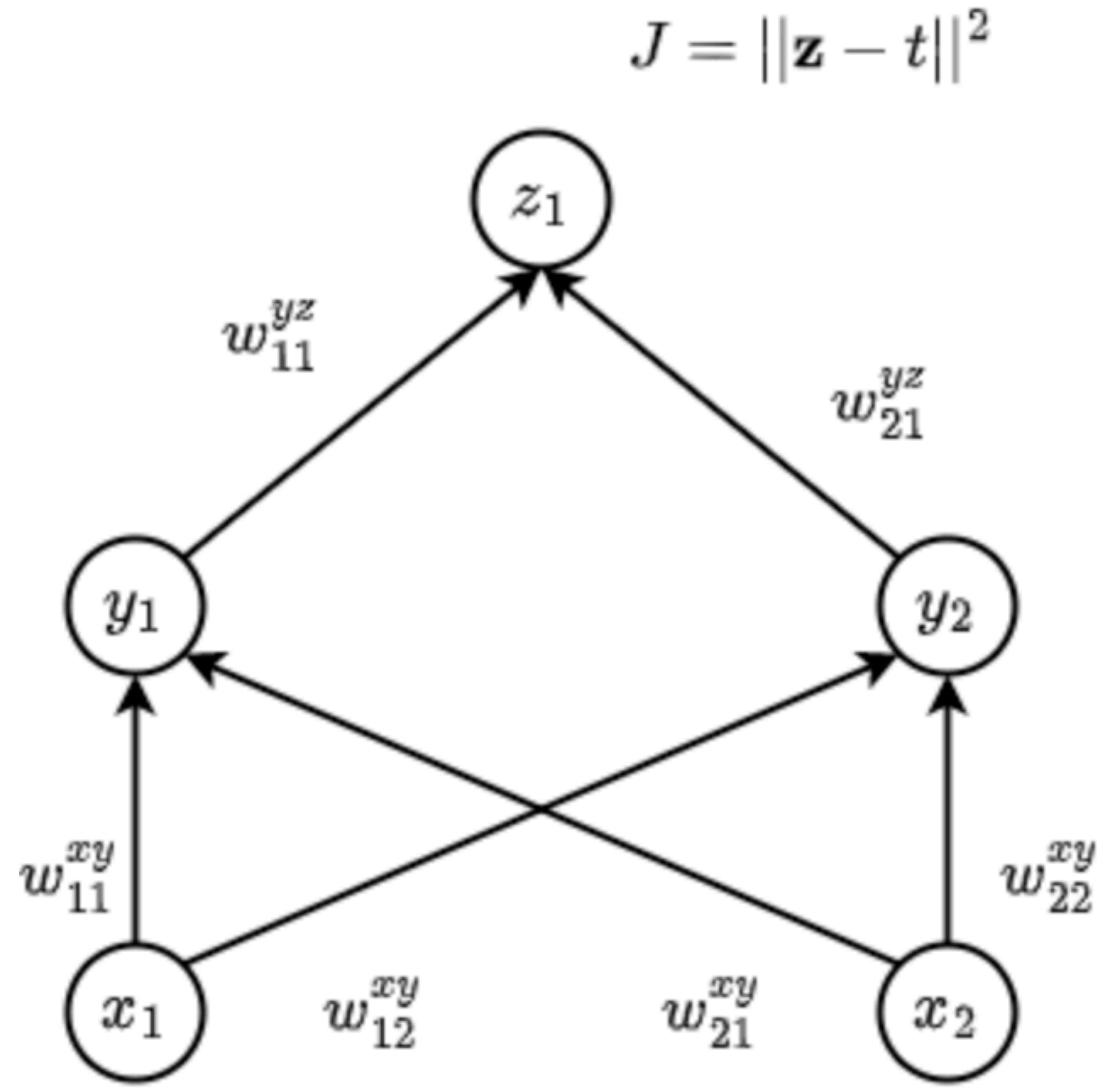
Back Propagation

- 2. Second objective is to find $\frac{\delta J}{\delta b_1}$
  - According to chain rule, $\frac{\delta J}{\delta w_{ij}} = \frac{\delta J}{\delta z_c} . \frac{\delta z_c}{\delta y_j} . \frac{\delta y_j}{\delta b_1}$
    - $\frac{\delta y_j}{\delta b_1} = 1$
  - Final solution: $\frac{\delta J}{\delta w_{ij}} = \frac{\delta J}{\delta z_c} * f' * w_{jc}^{yz}$

Back Propagation

Example 1

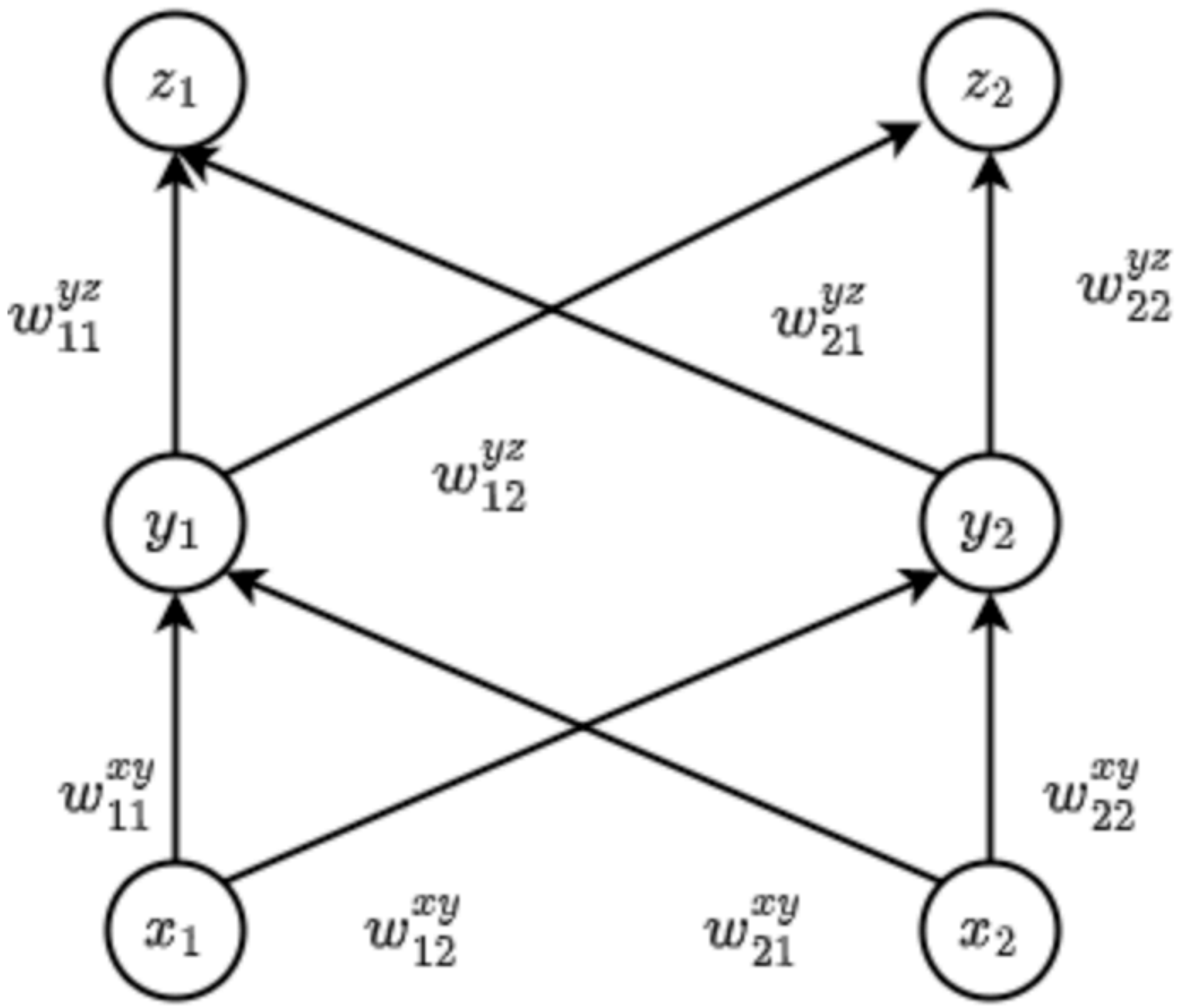
- $$\frac{\delta J}{\delta w_{11}^{yz}} = \frac{\delta J}{\delta z_1}\frac{\delta z_1}{w_{11}^{yz}}$$
  - $$\frac{\delta J}{\delta z_1} = 2(z_1 - t), (J = (z_1 - t)^2)$$
  - $$\frac{\delta z_1}{\delta w_{11}^{yz}} = y_1, (z_1 = y_1 \,.\, w_{11}^{yz} + b_{11}^{yz})$$
- $$\frac{\delta J}{\delta w_{11}^{yz}} = 2(z_1 - t) * y1$$



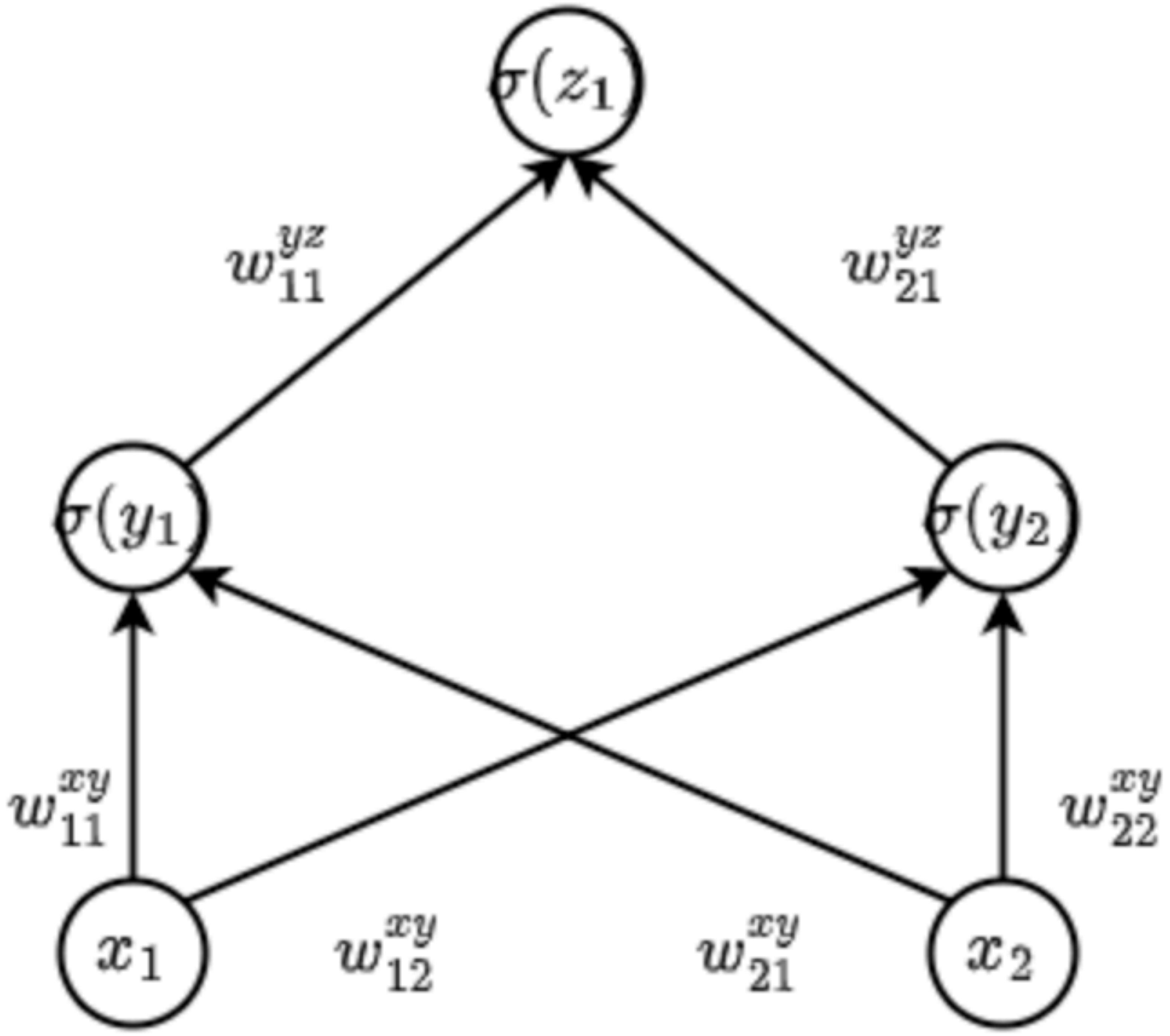
Back Propagation

Example 2&3

$$J = ||\mathbf{z} - t||^2$$



$$J = ||\sigma(\mathbf{z}) - t||^2$$



Exam-style Question

Code

Code

- ./materials/Week3_MLP/Week3_MLP.ipynb
- ./ResNet/Models/mlp.py
- ./ResNet/Models/resnet.py: line 70