Logistic Regression: Learning

강필성
고려대학교 산업경영공학부
pilsung_kang@korea.ac.kr

AGENDA

01 Logistic Regression: Formulation

02 Logistic Regression: Learning

03 Logistic Regression: Interpretation

04 Classification Performance Evaluation

05 R Exercise

Logistic Regression: Learning

- Estimating the coefficients
  - ✓ Assume that we have two different logistic models, each of which makes the predictions for the same dataset as below, which model is better?

Model A

Glass	Label	P(Y=1)	P(Y=0)
1	1	0.908	0.092
2	0	0.201	0.799
3	1	0.708	0.292
4	0	0.214	0.786
5	1	0.955	0.045
6	0	0.017	0.983
7	1	0.807	0.193
8	0	0.126	0.874
9	1	0.937	0.063
10	0	0.068	0.932

Model B

Glass	Label	P(Y=1)	P(Y=0)
1	1	0.557	0.443
2	0	0.425	0.575
3	1	0.604	0.396
4	0	0.387	0.613
5	1	0.615	0.385
6	0	0.356	0.644
7	1	0.406	0.594
8	0	0.508	0.492
9	1	0.704	0.296
10	0	0.325	0.675

  - ✓ Model A is better than Model B because Model A generates higher probabilities for the actual labels

Logistic Regression: Learning

- Estimating the coefficients
  - ✓ Likelihood function
    - Likelihood for an individual object is its predicted probability being classified as the correct class
      - Likelihood of Glass 1 is 0.908
      - Likelihood of Glass 2 is 0.799
    - If the objects are assumed to be generated independently, the likelihood of the entire dataset is the product of every object's likelihood
    - Generally the likelihood of a dataset is very small (values between 0 and 1 are compounded), log-likelihood is commonly used

Model A

Glass	Label	P(Y=1)	P(Y=0)
1	1	0.908	0.092
2	0	0.201	0.799
3	1	0.708	0.292
4	0	0.214	0.786
5	1	0.955	0.045
6	0	0.017	0.983
7	1	0.807	0.193
8	0	0.126	0.874
9	1	0.937	0.063
10	0	0.068	0.932

Logistic Regression: Learning

- Estimating the coefficients
  - ✓ Likelihood function

Model A

Glass	Label	P(Y=1)	P(Y=0)	우도	로그 우도
1	1	0.908	0.092	0.908	-0.0965
2	0	0.201	0.799	0.799	-0.2244
3	1	0.708	0.292	0.708	-0.3453
4	0	0.214	0.786	0.786	-0.2408
5	1	0.955	0.045	0.955	-0.0460
6	0	0.017	0.983	0.983	-0.0171
7	1	0.807	0.193	0.807	-0.2144
8	0	0.126	0.874	0.874	-0.1347
9	1	0.937	0.063	0.937	-0.0651
10	0	0.068	0.932	0.932	-0.0704
				0.233446	**-0.1455**

Model B

Glass	Label	P(Y=1)	P(Y=0)	우도	로그 우도
1	1	0.557	0.443	0.557	-0.5852
2	0	0.425	0.575	0.575	-0.5534
3	1	0.604	0.396	0.604	-0.5042
4	0	0.387	0.613	0.613	-0.4894
5	1	0.615	0.385	0.615	-0.4861
6	0	0.356	0.644	0.644	-0.4401
7	1	0.406	0.594	0.406	-0.9014
8	0	0.508	0.492	0.492	-0.7093
9	1	0.704	0.296	0.704	-0.3510
10	0	0.325	0.675	0.675	-0.3930
				0.004458	**-0.5413**

- ✓ Model A's (log) likelihood is greater than that of Model B
- ✓ Model A can explain the dataset better than Model A

Logistic Regression: Learning

- Maximum likelihood estimation (MLE)
  - ✓ Find the coefficients that maximizes the likelihood of the dataset
  - ✓ Likelihood of the object i

$$P(\mathbf{x}_i, y_i|\boldsymbol{\beta}) = \begin{cases} \sigma(\mathbf{x}_i|\boldsymbol{\beta}), & if \quad y_i = 1 \\ 1 - \sigma(\mathbf{x}_i|\boldsymbol{\beta}), & if \quad y_i = 0 \end{cases}$$

  - ✓ Since the y_i is either 0 or 1, we can rewrite the above probability as follows:

$$P(\mathbf{x}_i, y_i|\boldsymbol{\beta}) = \sigma(\mathbf{x}_i|\boldsymbol{\beta})^{y_i}(1 - \sigma(\mathbf{x}_i|\boldsymbol{\beta}))^{1-y_i}$$

Logistic Regression: Learning

- Maximum likelihood estimation (MLE)
  - ✓ Assume that the objects are independently generated, the likelihood of the entire dataset is expressed as follows:

$$L(\mathbf{X}, \mathbf{y}|\boldsymbol{\beta}) = \prod_{i=1}^{N} P(\mathbf{x}_i, y_i|\boldsymbol{\beta}) = \prod_{i=1}^{N} \sigma(\mathbf{x}_i|\boldsymbol{\beta})^{y_i}(1-\sigma(\mathbf{x}_i|\boldsymbol{\beta}))^{1-y_i}$$

  - ✓ Take a log for the both sides,

$$logL(\mathbf{X}, \mathbf{y}|\boldsymbol{\beta}) = \sum_{i=1}^{N} y_i \sigma(\mathbf{x}_i|\boldsymbol{\beta}) + (1-y_i)(1-\sigma(\mathbf{x}_i|\boldsymbol{\beta}))$$

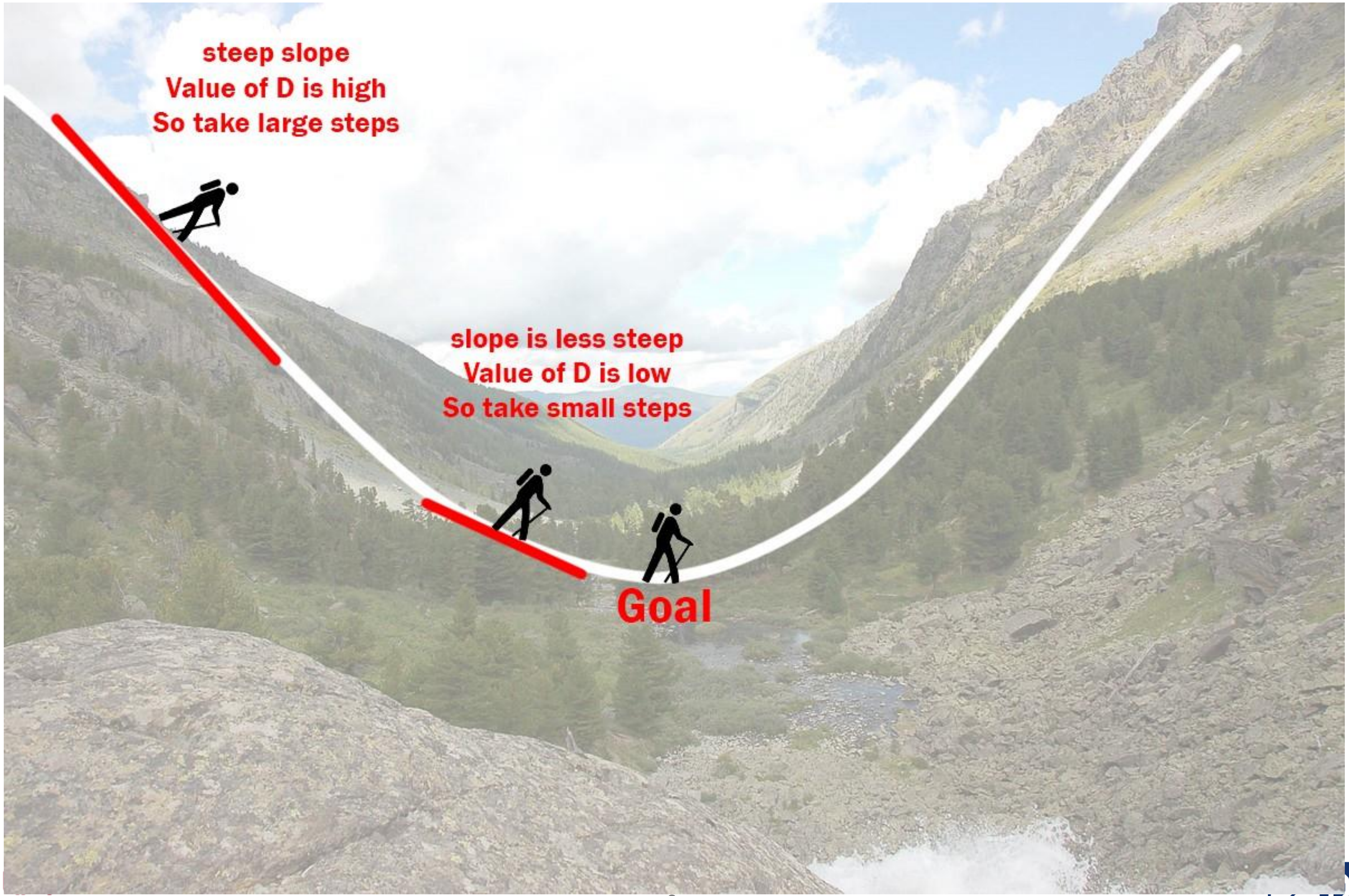
  - ✓ (Log) likelihood is non-linear with β, there is no explicit solution as in MLR
    - Find the solution with an optimization algorithm such as Gradient Descent

Logistic Regression: Learning

- Gradient Descent

Logistic Regression: Learning

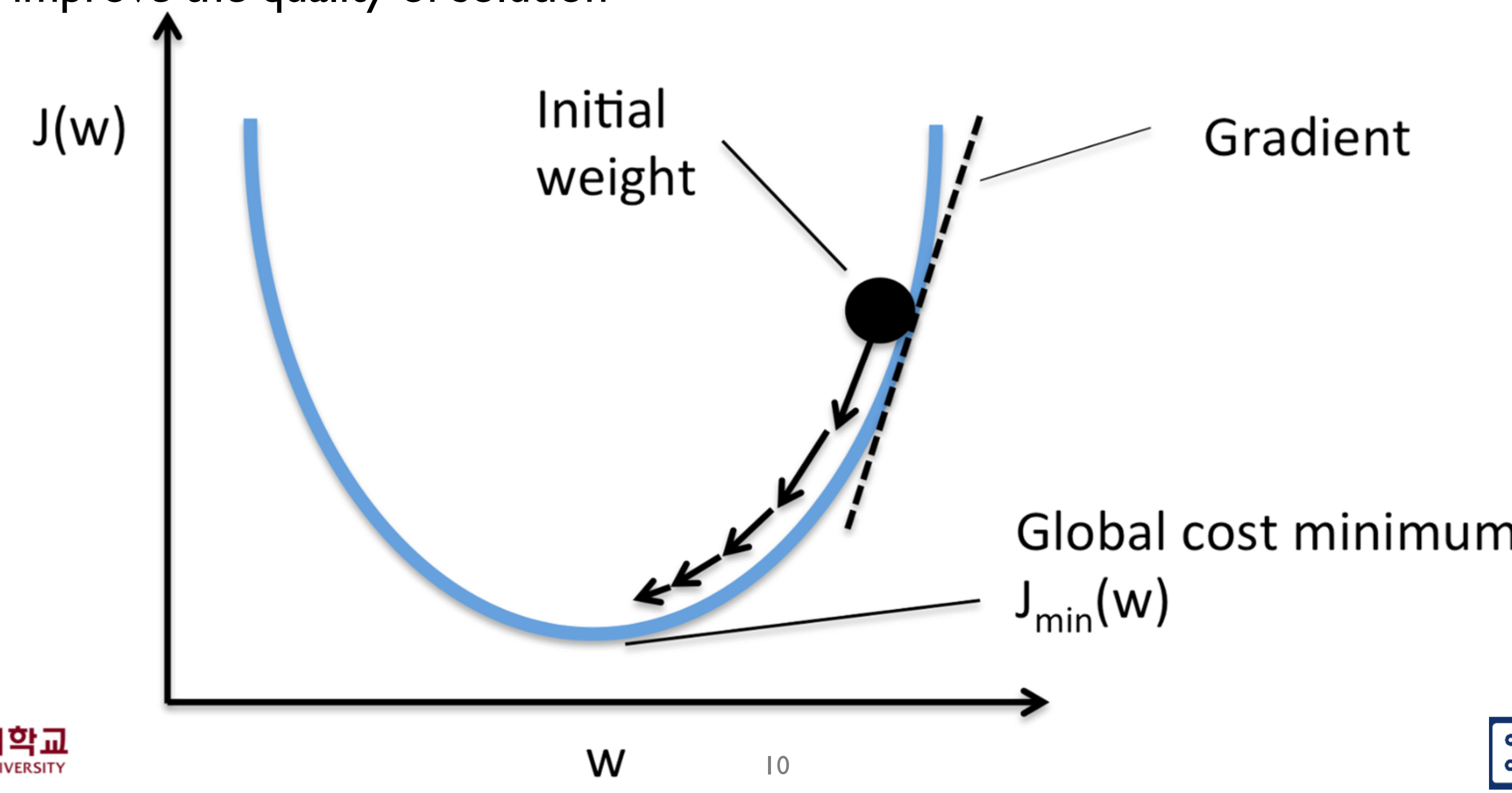
- Gradient Descent



https://towardsdatascience.com/linear-regression-using-gradient-descent-97a6c8700931

Logistic Regression: Learning

- Gradient Descent Algorithm
  - ✓ Blue line: the objective function to be minimized
  - ✓ Black circle: the current solution
  - ✓ Direction of the arrows: the direction that the current solution should move to improve the quality of solution



Logistic Regression: Learning

Gradient Descent Algorithm

- Take the first derivative of the cost function w.r.t the current weight w
  - ✓ <u>Is the gradient 0?</u>
    - Yes: Current weights are the optimum! → end of learning
    - No: Current weights can be improved → learn more
  - ✓ <u>How can we improve the current weights if the gradient is not 0?</u>
    - Move the current weight toward to the opposite direction of the gradient
  - ✓ How much should the weights be moved?
    - Not sure
    - Move them a little and compute the gradient again
    - It will converge

Logistic Regression: Learning

- Theoretical Background (Optional)

✓ Taylor expansion

$$f(w+\Delta w) = f(w) + \frac{f'(w)}{1!}\Delta w + \frac{f''(w)}{2!}(\Delta w)^2 + \cdots$$

✓ If the first derivative is not zero, we can decrease the function value by moving x toward the opposite direction of its first derivative

Which direction to go?

$$w_{new} = w_{old} - \alpha f'(w), \quad \text{where } 0 < \alpha < 1.$$

How far should we move?

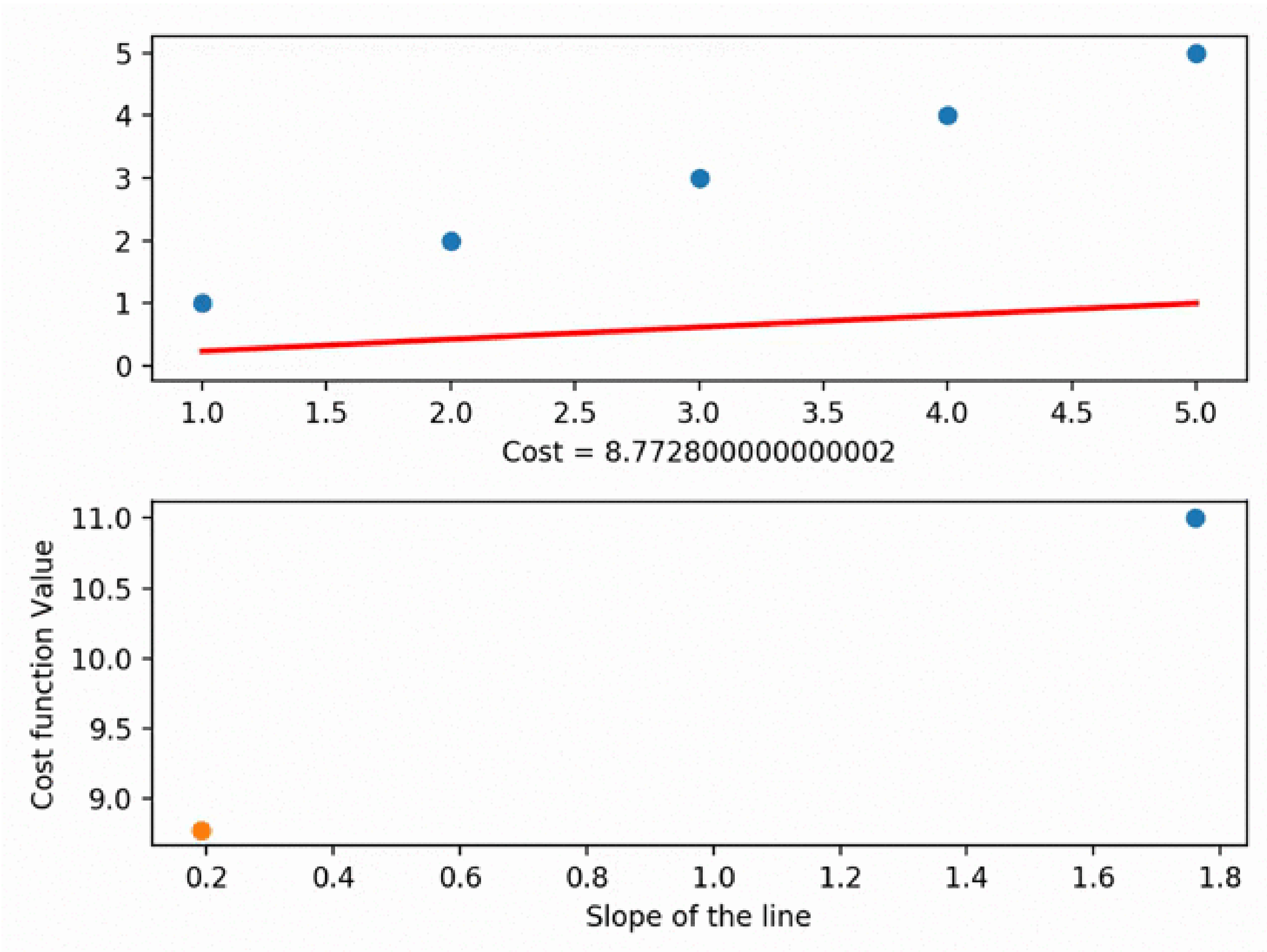
✓ Then the function value of the new x is always smaller than that of the old x

$$f(w_{new}) = f(w_{old} - \alpha f'(w_{old})) \cong f(w_{old}) - \alpha \left| f'(w) \right|^2 < f(w_{old})$$

https://www.youtube.com/watch?v=3d6DsjIBzJ4&t=322s

Logistic Regression: Learning

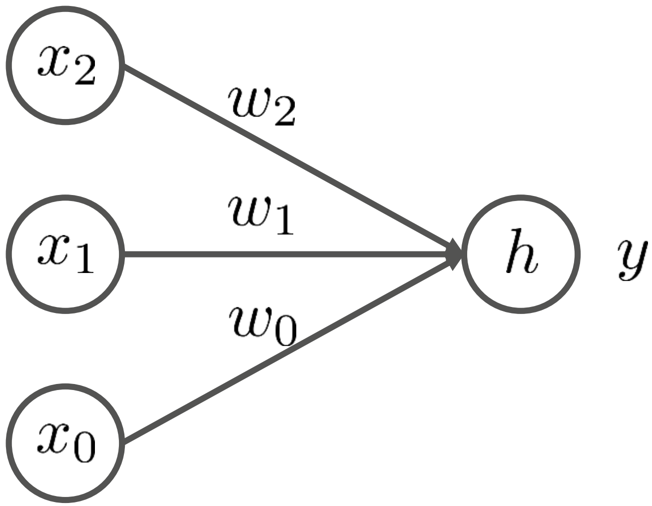
- Illustrative example



https://towardsdatascience.com/machine-learning-algorithms-in-laymans-terms-part-1-d0368d769a7b

Logistic Regression: Learning

- Gradient descent with two input variables



$$h = \sum_{i=0}^{2} w_i x_i$$

$$y = \frac{1}{1 + exp(-h)}$$

- Let's define the squared loss function for simplicity $L = \frac{1}{2}(t - y)^2$
- How to find the gradient w.r.t. w or x?

Logistic Regression: Learning

- Use chain rule

$$\frac{\partial L}{\partial y} = y - t$$

$$\frac{\partial y}{\partial h} = \frac{exp(-h)}{(1+exp(-h))^2} = \frac{1}{1+exp(-h)} \cdot \frac{exp(-h)}{1+exp(-h)} = y(1-y)$$

$$\frac{\partial h}{\partial w_i} = x_i$$

- Gradients for w and x

$$\frac{\partial L}{\partial w_i} = \frac{\partial L}{\partial y} \cdot \frac{\partial y}{\partial h} \cdot \frac{\partial h}{\partial w_i} = (y-t) \cdot y(1-y) \cdot x_i$$

- Update w

$$w_{new} = w_{old} - \alpha \times \frac{L}{\partial w_i} = w_{old} - \alpha \times (y-t) \cdot y(1-y) \cdot x_i$$

Logistic Regression: Learning

- Weight update by Gradient Descent

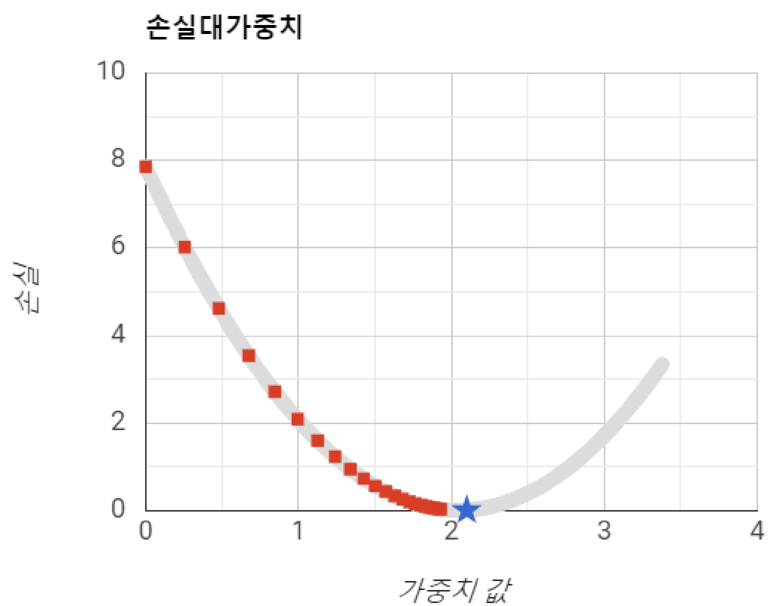
$$w_i^{new} = w_i^{old} - \alpha \times (y - t) \cdot 1 \cdot h(1-h) \cdot x_i$$

Update the coefficient more if the current output y is very different from the target t

Update the coefficients more if the value of corresponding input variable is large

Logistic Regression: Learning

- The Effect of learning rate α



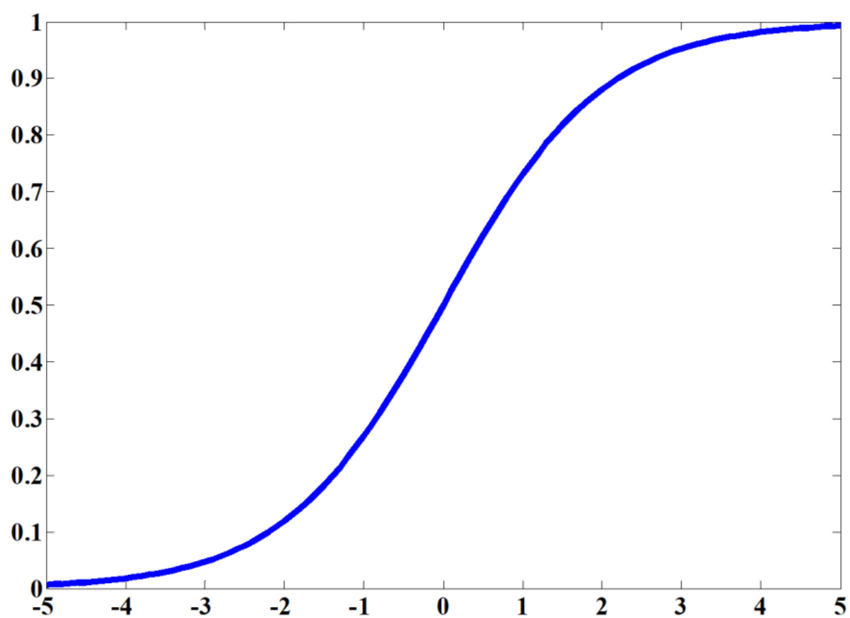
https://developers.google.com/machine-learning/crash-course/fitter/graph

Logistic Regression: Prediction

- Success probability
  - ✓ When a set of predictors (independent variables) are given, we can estimate the probability of the success.

$$p = \frac{1}{1 + e^{-(\hat{\beta}_0 + \hat{\beta}_1 x_1 + \hat{\beta}_2 x_2 \cdots + \hat{\beta}_d x_d)}}$$

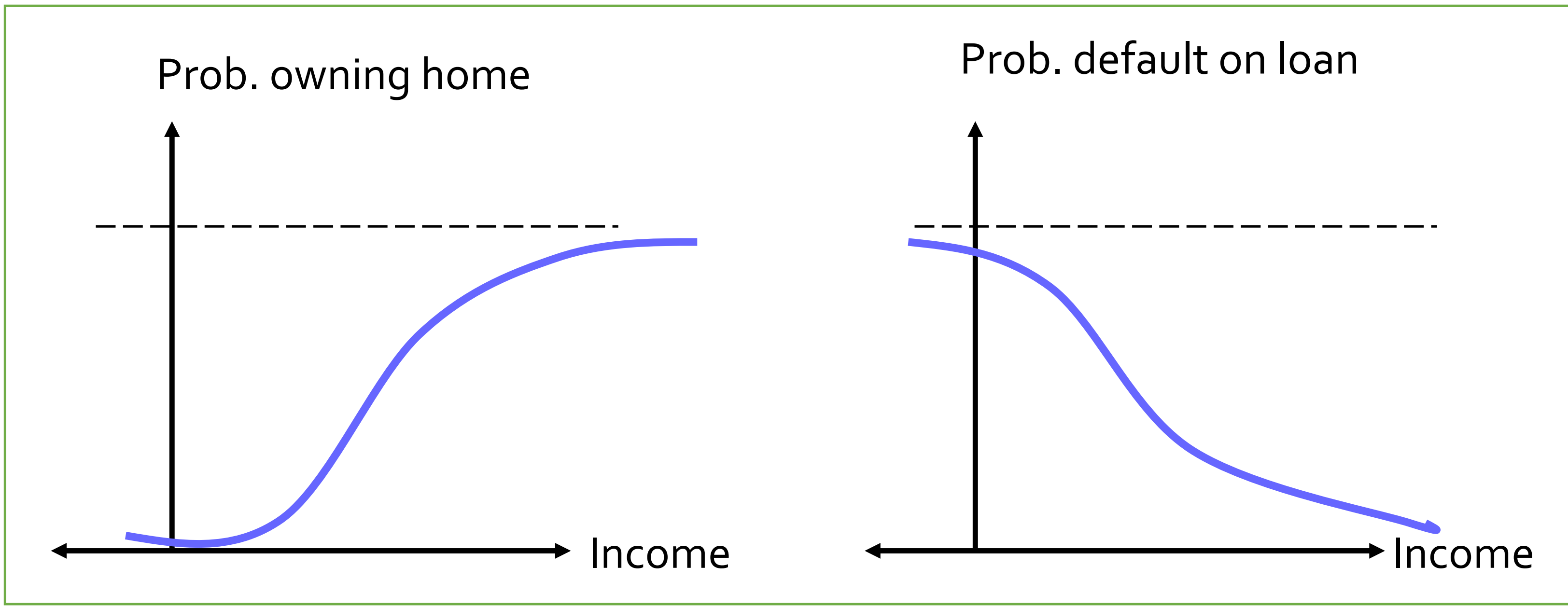
$$\frac{1}{1+e^{-a}}$$



a

For Classification Task

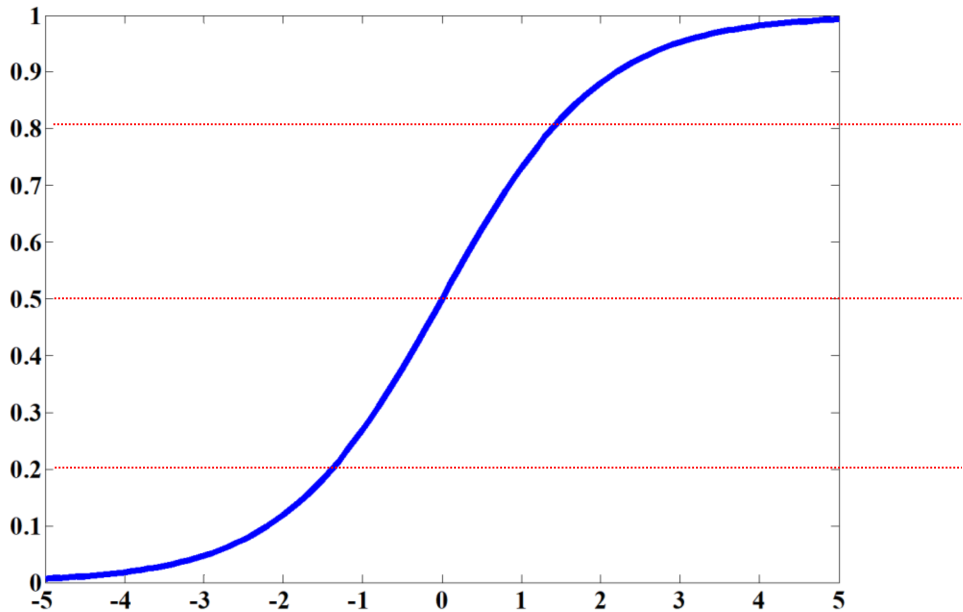
- In real cases…
  - ✓ The probability may follow a certain type of curve rather than a straight line.



Logistic Regression: Cut-off

- Determine the cut-off for the binary classification

$$\frac{1}{1+e^{-a}}$$



Cut-off when the majority class is the positive class

Most commonly used cut-off

Cut-off when the minority class is the positive class (ex: fault detection in manufacturing)

✓ 0.50 is popular initial choice

✓ Additional considerations: max. classification accuracy, max. sensitivity (subject to min. level of specificity), min. false positives (subject to max. false negative rate), min. expected cost of misclassification (need to specify costs)


ANY
questions?