Machine Learning

Logistic Regression - Theory

Dept. SW and Communication Engineering

Prof. Giseop Noh (kafa46@hongik.ac.kr)

Contents

- **Data for Logistic Regression**

- **Logistic Regression Model**

Study Goals

- Represent input data in a **feature space**
- Classify discrete actual values using **decision boundaries**
- Understand the **sigmoid function** and learn **classification methods**
- **Loss function** used in logistic regression
- Apply **gradient descent** to minimize it.

Data for Logistic Regression

Concept of Binary Classification

Classification

- The task of predicting the class/category to which an object belongs, based on a set of features.
- An observed object is described by a set of shared features.
- The training data for a classification problem is given:

$$D = \{ (x_i, y_i) \}_{i=1}^{N}$$

$$y_i = \{ C_1, C_2, \cdots, C_K \}$$

- Each data point consists of:
  - Input feature vector
  - Discrete label: y_i (from a set of k possible class labels)
- If only two classes exist, we typically assume:
  - The labels y_i represent binary outcomes (e.g., 1 for positive, 0 for negative)

Binary Classification

- **A classification problem where an object represented by a feature vector belongs to one of two classes.**

- **Typically expressed as $D = \{ (x_i, y_i) \}_{i=1}^{N}$, where $\boldsymbol{y_i \in \{0, 1\}}$.**

  [Example]

  - If we classify emails into spam and non-spam,
    - we assign 1 for spam and 0 otherwise.

- **The data consists of collections of observations with specific feature values.**

  [Example]

  - If a student is majoring in AI and has a GPA of 3.8, you can represent this as:
    - Major (categorical): AI
    - GPA (numerical): 3.8

Binary Classification - Example

- **Categorical information must be converted into numerical values for machine learning.**

[Example]

- Category Map
  - AI = 1
  - Mechanical Engineering = 2
  - Math = 3
  - Physics = 4,
- GPA is 3.8

➔ Feature vector might look like $[\ 1,\ \ 3.8\ ]^T$

Feature Vector

- **Students currently enrolled using their major and GPA,**

- **Each student can be represented as a feature vector with two attributes.**
  - When all objects are described using the same features, each individual observation is expressed in vector form.
  - This vector is called a feature vector

[Example]

- Student 1: majoring in Artificial Intelligence with a GPA of 3.8

  ➔ feature vector $[\ 1,\ \ 3.8\ ]^T$

- Student 2: majoring in Mathematics with a GPA of 4.0

  ➔ feature vector $[\ 3,\ \ 4.0\ ]^T$

Feature Space

Feature Space

- When using feature vectors, each observation can be represented as a point in a space where each dimension corresponds to one component of the feature vector.

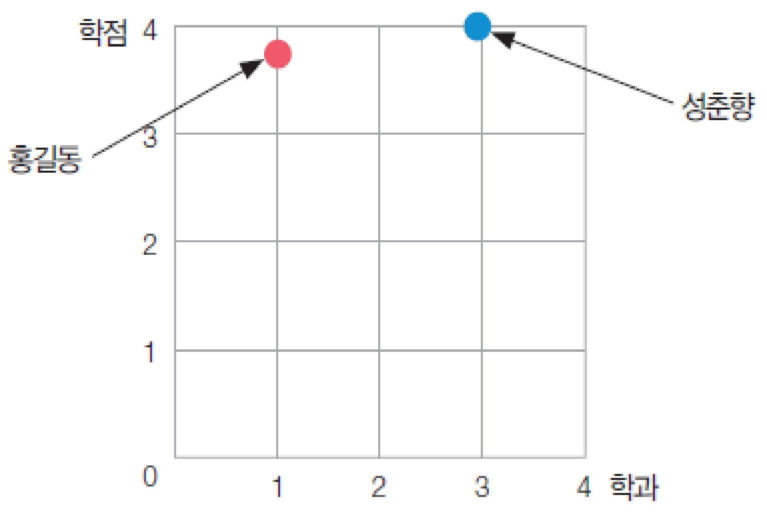
[Example]

- We represent a student's activity and performance

➔ $[1, 3.8]^T$ and $[3, 4.0]^T$ (Two feature vectors can be plotted in a 2D feature space)

- Feature vector has *d* components
- Each observation can be represented as a point in a *d*-dimensional space.

➔ This space is called the feature space.



Classification in Feature Space

- **Feature vectors of 3 students majoring in AI**

$$x_1 = [1 \;\; 2.2]^T, \qquad x_2 = [1 \;\; 3.8]^T, \qquad x_3 = [1 \;\; 3.9\,]^T$$

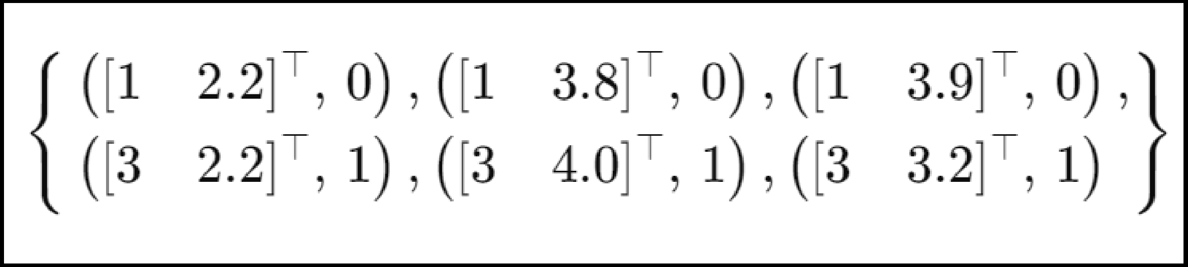
- **Feature vectors of 3 students majoring in Mathematics**

$$x_1 = [3 \;\; 2.2]^T, \qquad x_2 = [3 \;\; 4.0]^T, \qquad x_3 = [3 \;\; 3.3\,]^T$$

- **Suppose after surveying 6 students,**

  we find that all AI majors took the course and all Math majors did not.

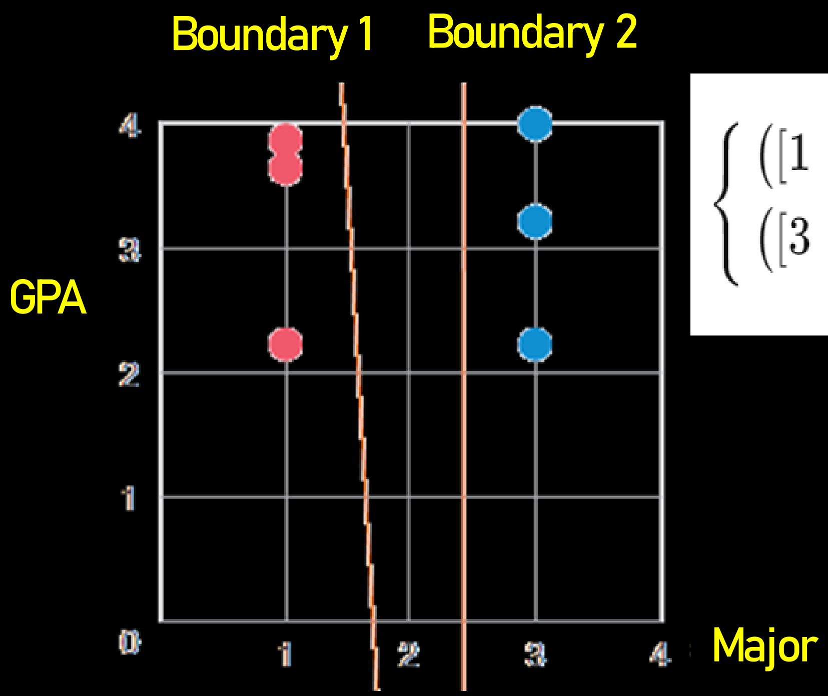
$$D = \{(x_i, y_i)\}_{i=1}^{6}$$



$$\left\{ \begin{matrix} \left([1 \quad 2.2]^\top, 0\right), \left([1 \quad 3.8]^\top, 0\right), \left([1 \quad 3.9]^\top, 0\right), \\ \left([3 \quad 2.2]^\top, 1\right), \left([3 \quad 4.0]^\top, 1\right), \left([3 \quad 3.2]^\top, 1\right) \end{matrix} \right\}$$

Visualization of feature space

- **Students in Number Theory course ➔ Blue dots**
- **Those who have not taken the course ➔ Red dots.**
- **Decision Boundary 1 or 2**
  - Separate students who took the Number Theory or not



$$\left\{ \begin{array}{l} \left([1 \quad 2.2]^\top, 0\right), \left([1 \quad 3.8]^\top, 0\right), \left([1 \quad 3.9]^\top, 0\right), \\ \left([3 \quad 2.2]^\top, 1\right), \left([3 \quad 4.0]^\top, 1\right), \left([3 \quad 3.2]^\top, 1\right) \end{array} \right\}$$

Binary Classification Using a Linear Discriminant Function

- **Discriminant Function**
  - A function that assigns a discrete predicted value to a feature vector given as input

- **Linear Discriminant Function**

$$g(X) = W^T X$$

parameter vector $W = [w_0, w_1, \cdots, w_d]^T$

Feature vector $X = [x_0, x_1, , \cdots, x_d]$

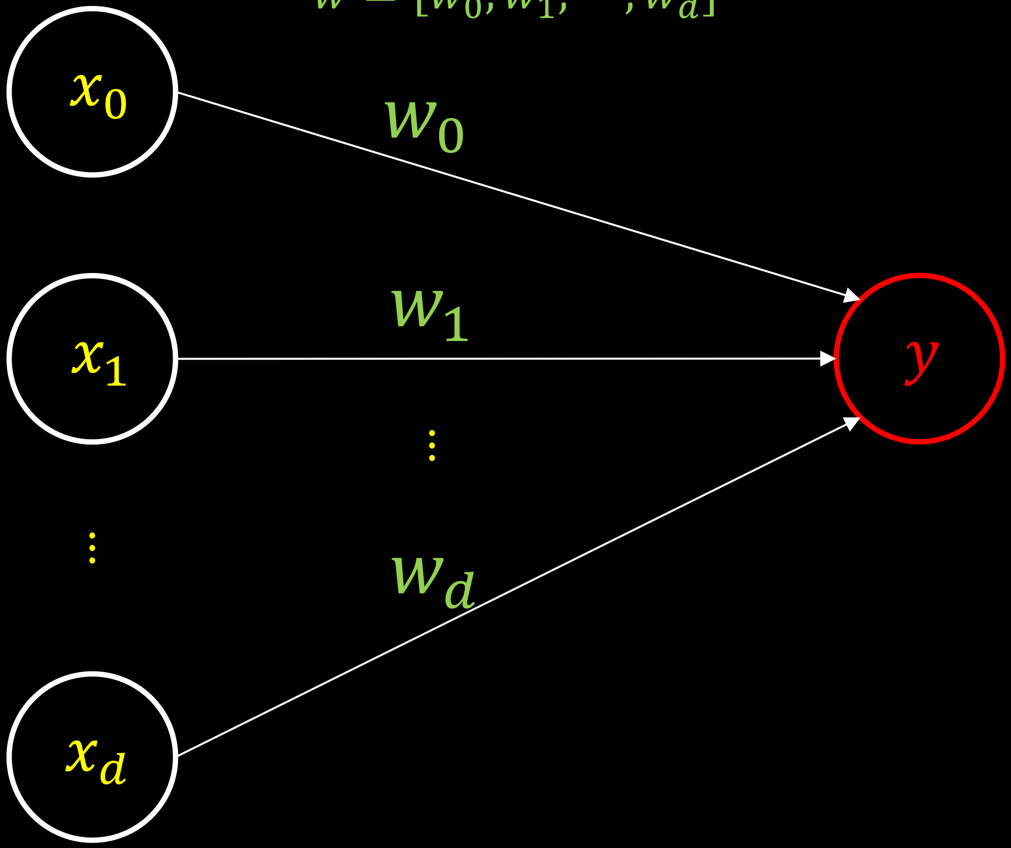
- **Once the discriminant function g(*x*) is determined,**

  the sign of *g(x)* for a given feature vector *x* is used

  to determine the discrete predicted value.

Visual understanding

$w = [w_0, w_1, \cdots, w_d]^T$

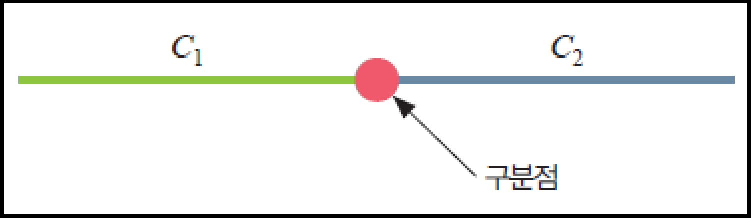


$x = [x_0, x_1,, \cdots, x_d]$

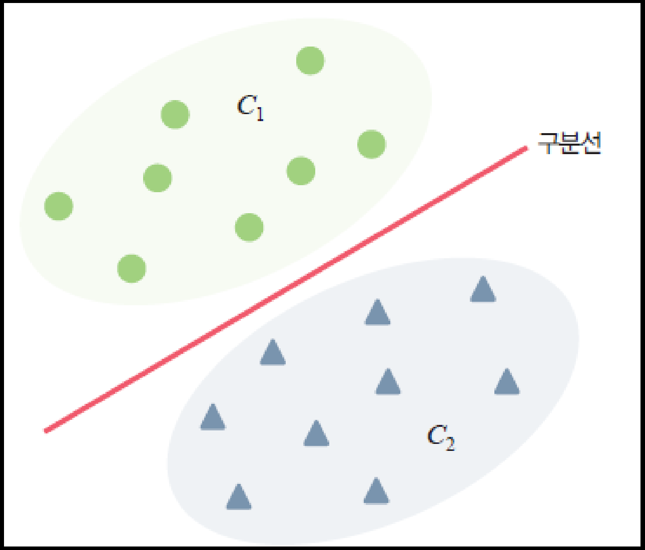
Linear Discriminant Function and Classification

- **In machine learning models used for classification, the feature space is divided in a way that supports accurate classification.**
  - When the feature space is 1-dimensional, it is divided by a point.
  - When the feature space is 2-dimensional, it is divided by a line.
  - When the feature space is 3-dimensional, it is divided by a plane.
  - When the feature space has 4 or more dimensions, it is divided by a hyperplane.

1 개의 성분으로 특징 벡터가 구성될 때 이진 분류



2 개의 성분으로 특징 벡터가 구성될 때 이진 분류



Example: Students taking the Number Theory course

Example: Students taking the Number Theory course

- Features: department and GPA
- Let the feature vector be $x = [x_1, x_2]^T$, where

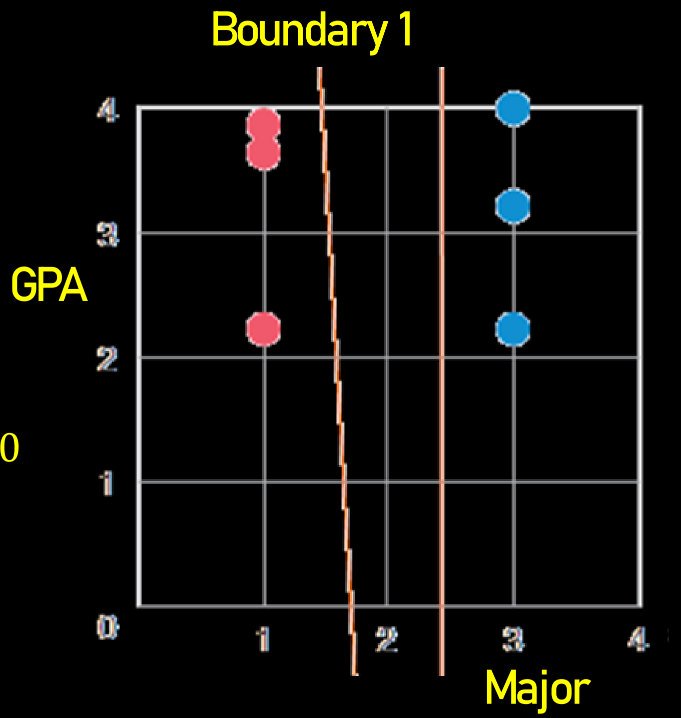
  x_1: department

  x_2: GPA

- If we define the decision boundary using

$$g(x) = 20x_1 + 3x_2 - 36 = 0$$

$$w = [-36 \quad 20 \quad 3]^T$$



- If we evaluate $x_2 = [1 \quad 3.8]^T$ and $x_4 = [3 \quad 2.2]^T$
  - $g(x_2) = 20 \cdot 1 + 3 \cdot 3.8 - 36 > 0$
  - $g(x_4) = 20 \cdot 3 + 3 \cdot 2.2 - 36 < 0$

This allows us to distinguish students who took the course from those who did not using the sign of $g(x)$

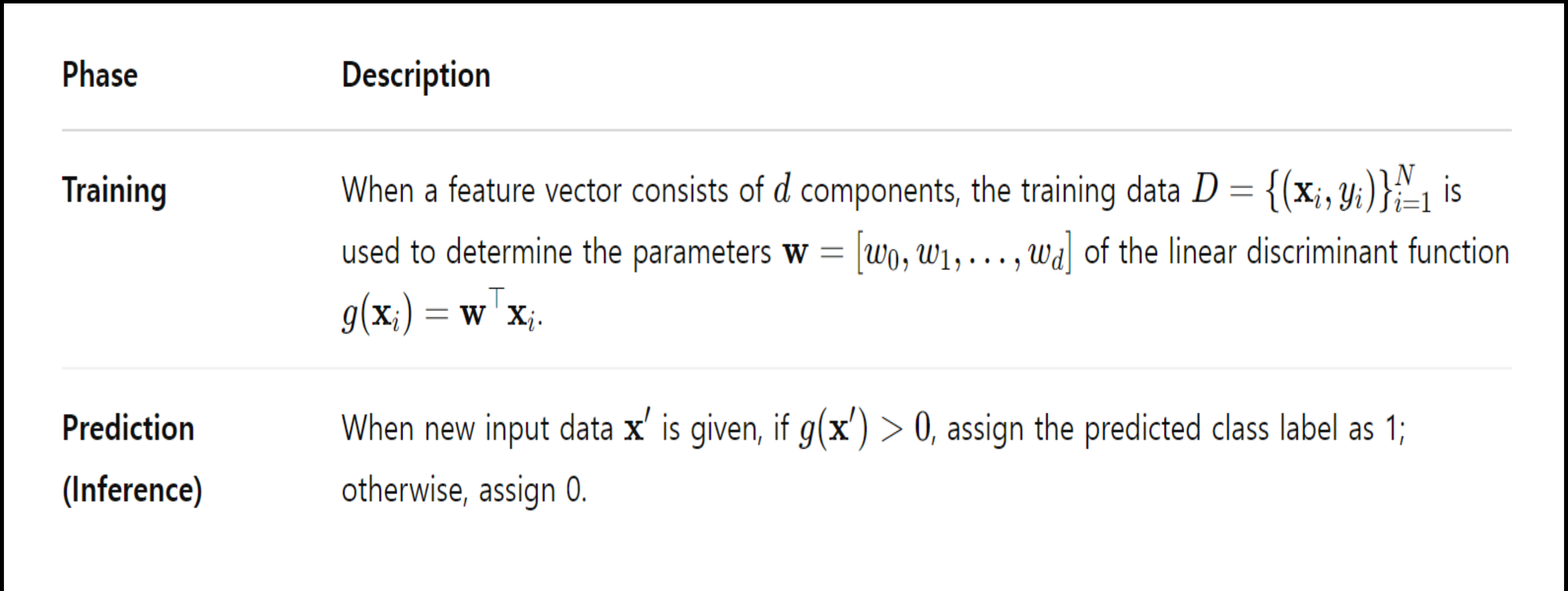
Summary on Decision Boundary

If $g(x) = 0$ defines the decision boundary,

- the sign of $g(x)$ determines whether a student is
  - Course taker

    or
  - Non-taker

  by dividing the feature space accordingly.

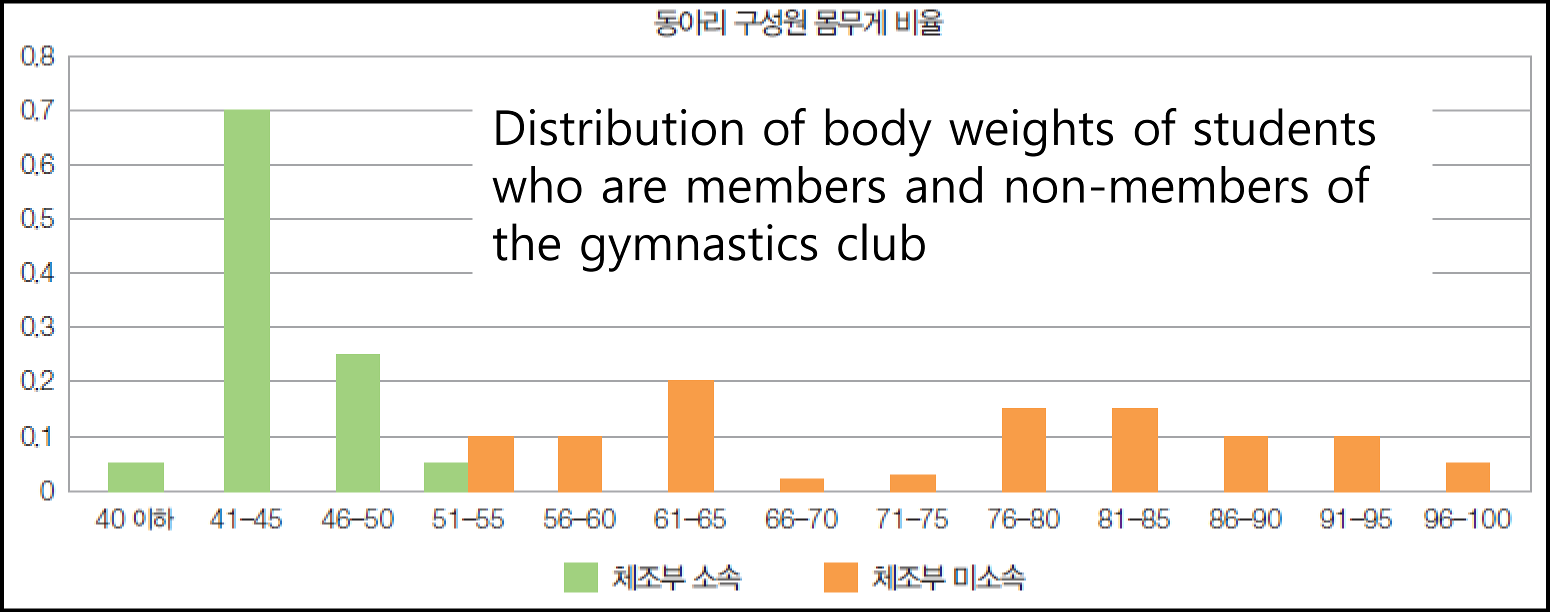


Phase	Description
Training	When a feature vector consists of d components, the training data $D = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N}$ is used to determine the parameters $\mathbf{w} = [w_0, w_1, \ldots, w_d]$ of the linear discriminant function $g(\mathbf{x}_i) = \mathbf{w}^\top \mathbf{x}_i$.
Prediction (Inference)	When new input data $\mathbf{x}'$ is given, if $g(\mathbf{x}') > 0$, assign the predicted class label as 1; otherwise, assign 0.

More Practical Example on Logistic Regression Data

Binary Classification Using Posterior Probability

- **Members and non-members of a gymnastics club at university A**
  - Students who are members of the “gymnastics” club
  - or Not
- **We recorded their body weights and obtained the distribution in histogram**



Distribution of body weights of students who are members and non-members of the gymnastics club

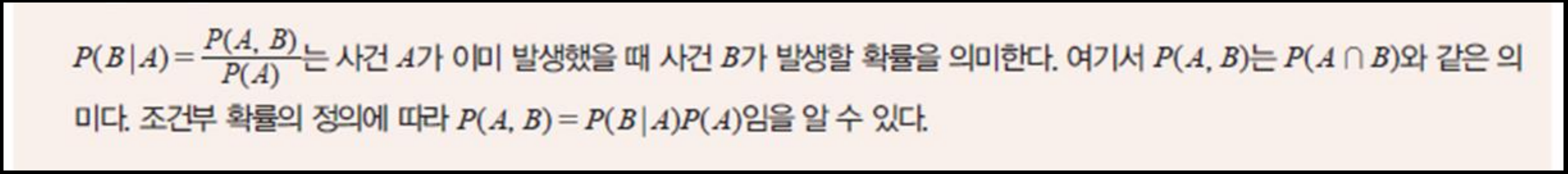
(Example) Data for Logistic Regression

- **How can we determine whether a student weighing 52 kg is a member of the gymnastics club?**
  - Let's assume
    - y = 1 for members of the gymnastics club
    - y = 0 for non-members

- **Let the observed value (evidence) be that the student weighs 52 kg**
  - Calculate $P(y = 1 \mid x)$ and $P(y = 0 \mid x)$
  - If $P(y = 1 \mid x) > P(y = 0 \mid x)$
    - the student is a member of the gymnastics club
    - otherwise, they are not.

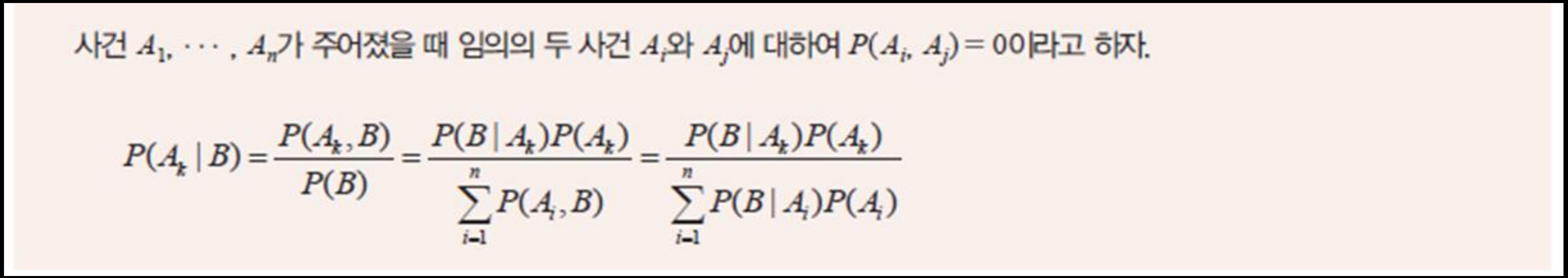
Bayes' Theorem

Conditional Probability



$P(B|A)=\frac{P(A, B)}{P(A)}$는 사건 A가 이미 발생했을 때 사건 B가 발생할 확률을 의미한다. 여기서 $P(A, B)$는 $P(A \cap B)$와 같은 의미다. 조건부 확률의 정의에 따라 $P(A, B)=P(B|A)P(A)$임을 알 수 있다.

Bayes' Theorem



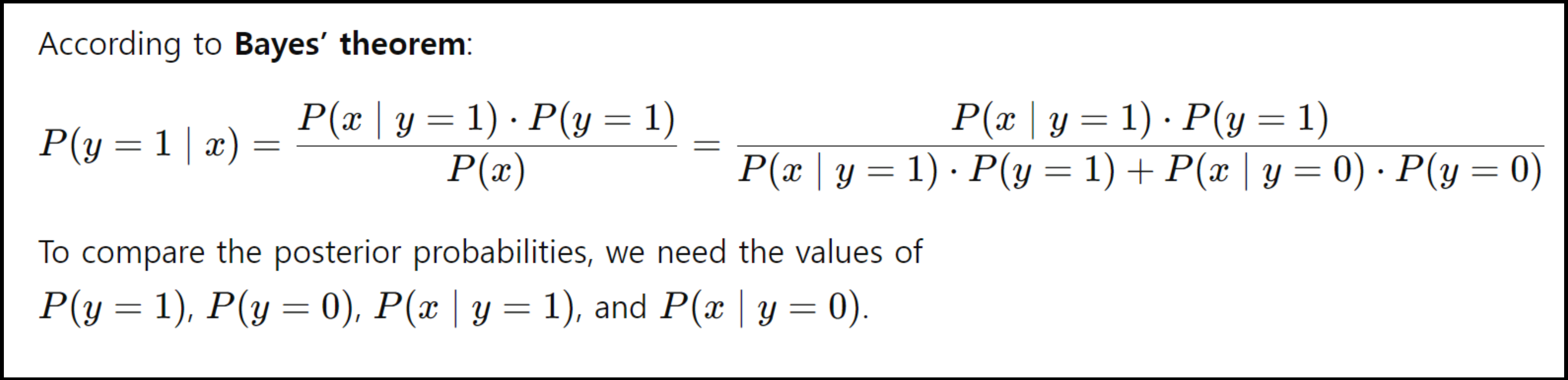
사건 $A_1, \cdots, A_n$가 주어졌을 때 임의의 두 사건 A_i와 A_j에 대하여 $P(A_i, A_j)=0$이라고 하자.

$$P(A_k \mid B)=\frac{P(A_k, B)}{P(B)}=\frac{P(B \mid A_k)P(A_k)}{\sum_{i=1}^{n} P(A_i, B)}=\frac{P(B \mid A_k)P(A_k)}{\sum_{i=1}^{n} P(B \mid A_i)P(A_i)}$$

Simple Version

$$P(A|B)=\frac{P(B|A)\cdot P(A)}{P(B)}$$

(Example) Data for Logistic Regression



According to **Bayes' theorem**:

$$P(y=1 \mid x) = \frac{P(x \mid y=1) \cdot P(y=1)}{P(x)} = \frac{P(x \mid y=1) \cdot P(y=1)}{P(x \mid y=1) \cdot P(y=1) + P(x \mid y=0) \cdot P(y=0)}$$

To compare the posterior probabilities, we need the values of $P(y=1)$, $P(y=0)$, $P(x \mid y=1)$, and $P(x \mid y=0)$.

사후 확률(Posterior probability)은 조건부 확률

$P(y = 1|x)$는 x를 관찰하였을 때(혹은 x가 주어졌을 때)

$y = 1$이 성립할 정도를 수치로 표현한 값

Logistic Regression Model

Sigmoid Function

Properties of the Sigmoid Function

- Denoted as $\sigma(z)$
- Maps any real-valued number into a value between 0 and 1
- Differentiable for any input

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

$$\frac{d}{dz}\sigma(x)$$??

Let, $y = \dfrac{1}{1+e^{-x}}$

$$y = (1+e^{-x})^{-1}$$

Differentiate

$$\frac{dy}{dx} = -1 \cdot (1+e^{-x})^{-2} \cdot \frac{d}{dx}(1+e^{-x})$$

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

So, $\dfrac{dy}{dx} = \dfrac{e^{-x}}{(1+e^{-x})^{2}}$

Therefore,

$$\frac{d}{dx}\sigma(x) = \frac{e^{-x}}{(1+e^{-x})^{2}} = \frac{1}{1+e^{-x}} \cdot \frac{e^{x}}{1+e^{x}}$$

$$= \sigma(x) - (1 - \sigma(x))$$

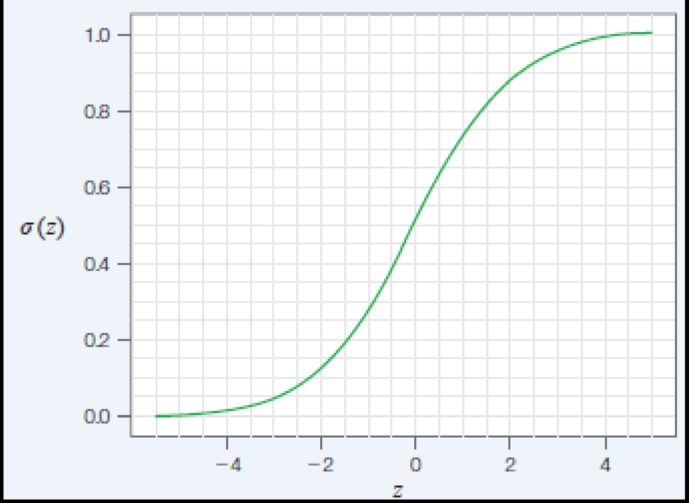
Usage of Sigmoid Function

Application 1

- Represent the probability of a certain event occurring
- Since the sigmoid function outputs values between 0 and 1, it can be used to represent the probability of a binary outcome based on the value of the input variable(as in logistic regression)

Application 2

- Activation function in the computation process of artificial neural networks
- Both the sigmoid function and its derivative are important and widely used.



Sigmoid Function

Concept of Logistic Function

Concept of Logistic Regression

- A model used to solve classification problems by predicting the probability of an event occurring.

Expression of Event Probability

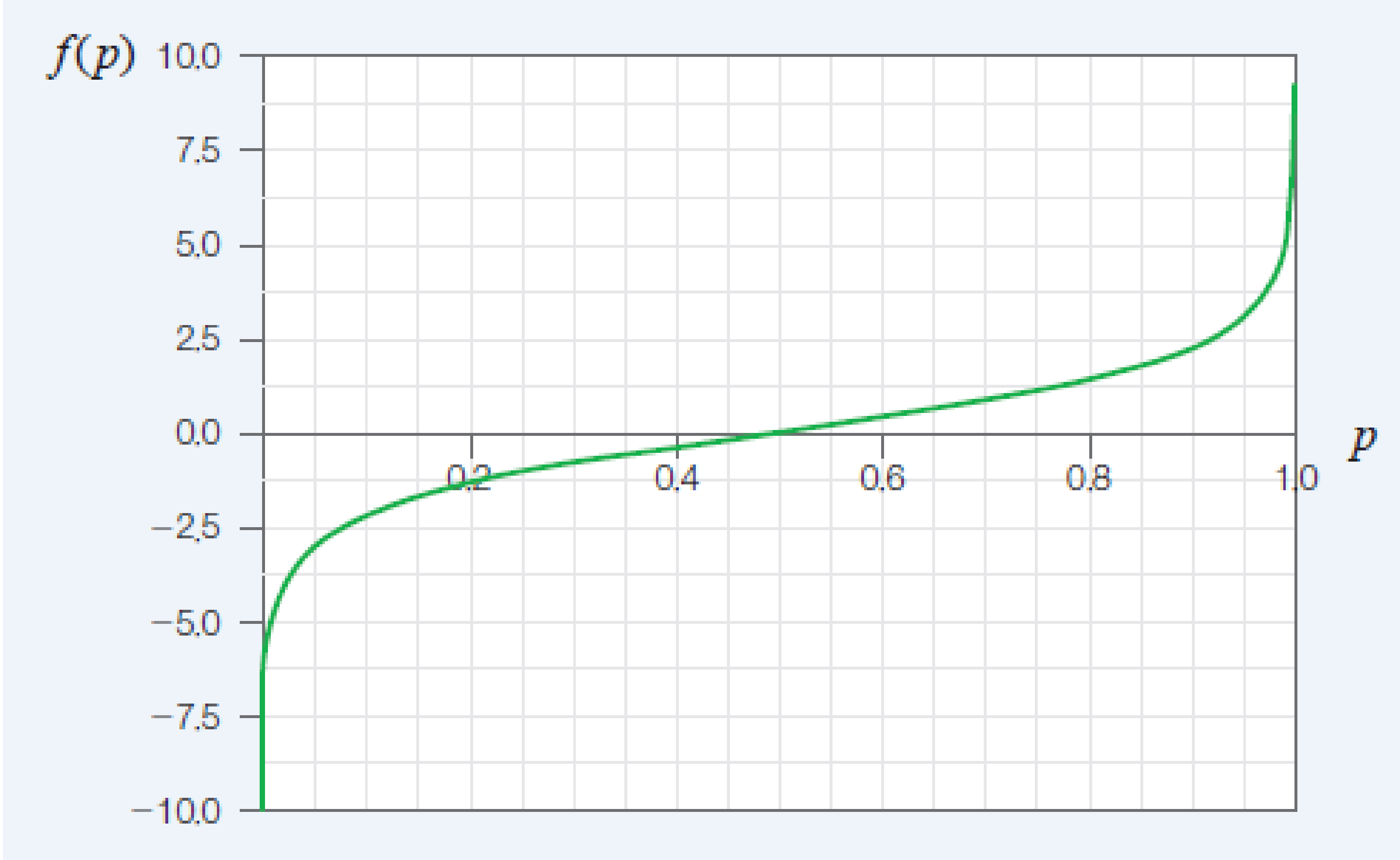
- Probability
  - "The chance of the Korean team to the semi-finals is less than 1 in 5."
- Odds: The ratio of the probability that an event will occur to the probability that it will not occur

$$Odds = \frac{p}{1-p}, where\ p\ is\ the\ probability\ of\ the\ event\ occurring$$

- Log-odds: The logarithm of the odds

$$\log odds = \ln \frac{p}{1-p}$$

The Shape of log odds



Binary Classification

- **A situation where all objects belong to one of two classes**

(Example)

- Whether the Korean national soccer team advances to the semi-finals
- Binary labels can be represented as:
  - $y = 1$ or $y = 0$
- If we can calculate the posterior probability,
  - we can compare $P(y = 1 \mid x)$ and $P(y = 0 \mid x)$ for classification
  - If $P(y = 1 \mid x) > 0.5$, then the predicted value for input x is 1

- **The log-odds of the probability**

- Object with feature vector x has the binary label 1

$$\ln \frac{P(y = 1|x)}{1 - P(y = 1|x)} = w_0 + w_1 x$$

Log odds in Logistic Regression

- **In logistic regression,**

  the log-odds are expressed as a linear function.

$$\ln \frac{P(y=1|x)}{1-P(y=1|x)} = w_0 + w_1x_1 + w_2x_2 + \cdots + w_dx_d = W^TX$$

$$Weight\ Params: W = [w_0, w_1, \cdots, w_d]^T$$

Step-by-step derivation from log-odds to the sigmoid function

(commonly used in logistic regression)

$$\ln \frac{P(y=1|x)}{1-P(y=1|x)} = w^Tx$$

$Let \quad z = w^Tx$

$Also, let$

$$p = P(y=1|x)$$

$$\Rightarrow \quad \ln \frac{p}{1-p} = z$$

Go to next slide

Step-by-step derivation from Log-odds to Sigmoid

$$\ln \frac{p}{1-p} = z$$

Exponentiate both side

$$\frac{p}{1-p} = e^z$$

Multiply both sides by $1-p$

$$p = e^z(1-p)$$

Distribute

$$p = e^z - e^z p$$

Bring p terms together

$$p + e^z p = e^z$$

Factor p

$$p(1+e^z) = e^z$$

Divide both sides by $1+e^z$

$$p = \frac{e^z}{1+e^z}$$

Exactly same form of Sigmoid Function!

Divide both the numerator and the denominator by x

$$p = \frac{1}{\frac{1}{e^z}+1} = \frac{1}{1+e^{-z}} = \frac{1}{1+e^{-w^T x}}$$

Therefore,

$$P(y=1|x) = \frac{\exp(w^T x)}{1+\exp(w^T x)} = \sigma(w^T x) = \sigma(z)$$

Likelihood

Expression of Likelihood

Training Dataset:

$$D = \{ (x_i , y_i)\}_{i=1}^{N}$$

Maximum Likelihood Estimation (MLE)

Find parameters w_0, w_1

that maximize the likelihood.

Expression of Posterior Probability:

$$P(y = 1|x_i) = \sigma(w_0 + w_1 x_i)$$

For given training data, the parameters w_0, w_1 determine the classification result:

$$P(y_i|x_i) = \sigma(w_0 + w_1 x_i)^{y_i}\big(1 - \sigma(w_0 + w_1 x_i)\big)^{1-y_i}$$

Interpretation

$$p(y_i|x_i) = \begin{cases} \sigma(w_0 + w_1 x_i), & if\ y_i = 1 \\ 1 - \sigma(w_0 + w_1 x_i) & if\ y_i = 0 \end{cases}$$

Exactly same to the Bernoulli Distribution (Probability Mass Function)

$f(k;p) = p^k(1-p)^{1-k}$ for possible outcome $k \in \{0, 1\}$ and

given probability $p = \sigma(w_0 + w_1 x_i)$

Likelihood of training dataset D

Simple case

$$y_i = w_0 + w_1 x_i$$

$$\prod_{i=1}^{N} P(y_i|x_i) = \prod_{i=1}^{N} [\sigma(w_0 + w_1 x_i)^{y_i}(1 - \sigma(w_0 + w_1 x_i)^{1-y_i})]$$

General case

$$y_i = w_0 + w_1 x_{i1} + w_2 x_{i2} + \cdots + w_d x_{id}$$

$$\prod_{i=1}^{N} P(y_i|x_i) = \prod_{i=1}^{N} [\sigma(w_0 + w_1 x_{i1} + \cdots + w_d x_{id})^{y_i}(1 - \sigma(w_0 + w_1 x_{i1} + \cdots + w_d x_{id})^{1-y_i})]$$

Interpretation of Likelihood

If $y_i = 1$

$$P(y_i = 1|x_i) = \sigma(w_0 + w_1 x_i)$$

If $y_i = 0$

$$P(y_i = 0|x_i) = 1 - \sigma(w_0 + w_1 x_i)$$

Likelihood function $L(\cdot)$ is the product of all $P(y_i|x_i)$

The higher $L(\cdot)$ is preferred!!

In practice, the negative log-likelihood $-\ln L(\cdot)$ is used for ease of computation

In this case, smaller value is preferred!!

$P(y_i|x_i)$: conditional probability, based on the model parameters (w_0, w_1)

Input x_i belongs to the correct class for the i-th observation

Learning Objective & NLL

Learning Objective

Find the parameter vector w that maximizes classification performance on the given training data D.

Negative Log Likelihood (NLL)

- Negative value of the log of the likelihood function
- Used to define the loss function $L(\cdot)$

$$L(w) = -\frac{1}{N}\sum_{i=1}^{N} \ln P(y_i|x_i)$$

$N: the\ number\ of\ samples\ in\ dataset\ D$
$w: parameter\ vector$

Learning Apporach

The posterior probability $P(y_i = 1|x_i)$ is determined by the feature vector x_i and parameters w.

Therefore, our goal is to find parameter w that maximizes the likelihood $L(w)$.

$$w^* = arg\max_w L(w) = \frac{1}{N}\sum_{i=1}^{N} P(y_i|x_i)$$

(Exactly same meaning)

In other words, our goal is to find parameter w that minimizes the NLL (Negative Log Likelihood $-\ln L(w)$).

$$w^* = arg\,\underset{w}{\text{mix}}\, L(w)$$

$$= -\frac{1}{N}\sum_{i=1}^{N} \ln P(y_i|x_i)$$

Binary Cross Entropy (BCE)

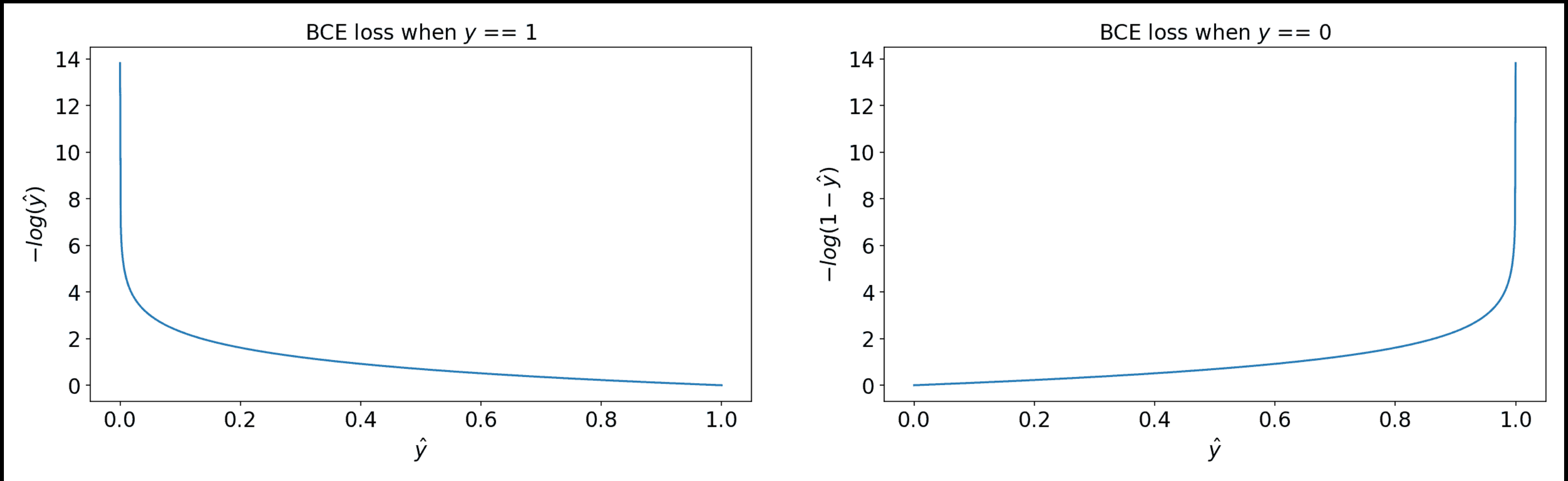
NLL (Negative Log Likelihood) and BCE (Binary Cross Entropy)

Commonly used loss functions in binary classification, especially in logistic regression and neural networks.

In binary classification with a sigmoid output, BCE and NLL are mathematically equivalent

Definition of BCE

$$BCE = -\frac{1}{N}\sum_{i=1}^{N}\{y_i \cdot \log \hat{y}_i + (1-y_i)\cdot \log(1-\hat{y}_i)\}$$



Learning with Gradient Descent

Loss Function

Negative Log Likelihood (NLL) Loss

Procedure

1) Initialization: Randomly Initializate Parameters

2) Feed Input

3) Compute NLL Loss

3) Update Parameters

Repeat until reaching to the end condition

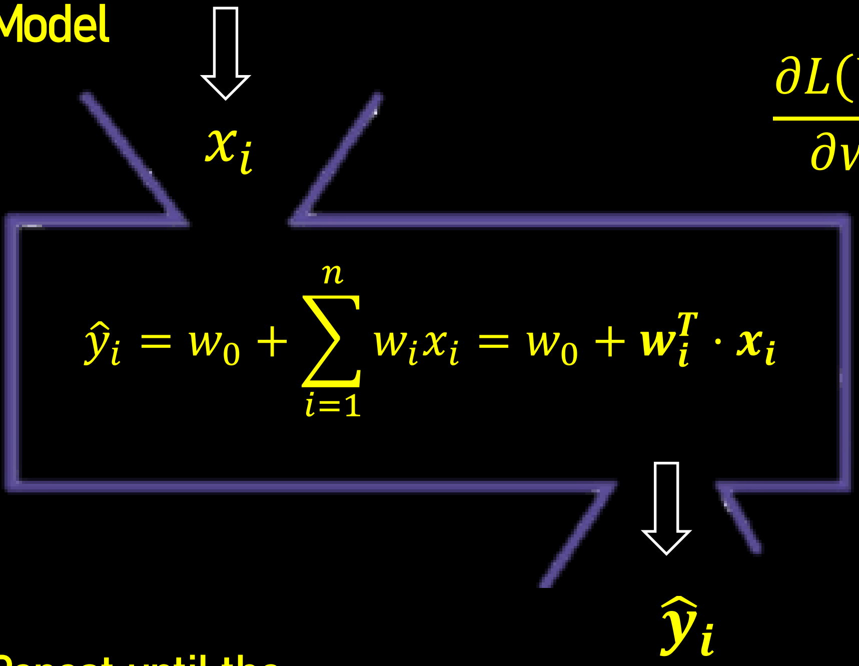
Feed-forward

$$NLL = L(W) = -\frac{1}{N}\sum_{i=1}^{N} \ln P(\hat{y}_i|X_i)$$

미분과정 생략
자세한 내용은
다음 슬라이드 참조

Feed input

to Model



$$x_i$$

$$\hat{y}_i = w_0 + \sum_{i=1}^{n} w_i x_i = w_0 + \boldsymbol{w}_i^T \cdot \boldsymbol{x}_i$$

$$\hat{\boldsymbol{y}}_i$$

$$\frac{\partial L(W)}{\partial w_j} = -\frac{1}{N}\sum_{i=1}^{N}\left(y_i - \sigma(W^T X_i)\right)x_{ij}$$

$$,where\ \ \boldsymbol{X}_i = [x_{i1}, \cdots, x_{id}]$$

$$w_j \leftarrow w_j - \alpha \boxed{\frac{\partial L(W)}{\partial w_j}}$$

Repeat until the

terminaltion condition

is satisfied

Derivative of Loss Function (1/3)

$$BCE = NLL = L(W)$$

$$= -\frac{1}{N}\sum_{i=1}^{N}\{y_i \cdot \log \hat{y}_i + (1-y) \cdot \log(1-\hat{y}_i)\}$$

$$= -\frac{1}{N}\sum_{i=1}^{N}\{y_i \cdot \sigma(z_i) + (1-y) \cdot \log(1-\sigma(z_i))\}$$

$$where\ z_i = W^T X_i$$

미분 목표: 모든 파라미터

즉, $w_0, w_1, \cdots, w_j, \cdots, w_d$ 에

대하여 각각 미분

$$\frac{\partial L(W)}{\partial w_j}$$

Again, 시그모이드 미분

$$\frac{d}{dx}\sigma(x) = \frac{e^{-x}}{(1+e^{-x})^2}$$

$$= \frac{1}{1+e^{-x}} \cdot \frac{e^x}{1+e^x}$$

$$= \sigma(x) - (1-\sigma(x))$$

Derivative of Loss Function (2/3)

합성함수 미분 (Chain Rule)

ln 함수 미분 × $\sigma(\cdot)$미분 × z_i 미분

$$\frac{\partial}{\partial w_j} \ln \sigma(z_i) = \frac{1}{\sigma(z_i)} \sigma'(z_i) \frac{\partial z_i}{\partial w_j}$$

ln 함수 미분 × 중간함수 $(1 - z_i)$미분 × $\sigma(\cdot)$미분 × z_i 미분

시그모이드 미분 대입

$$\frac{\partial}{\partial w_j} \ln \sigma(z_i) = \frac{1}{\sigma(z_i)} \sigma(x) - (1 - \sigma(x)) \frac{\partial z_i}{\partial w_j}$$

$$\frac{\partial}{\partial w_j} \ln \sigma(1 - z_i) = \frac{1}{\sigma(1 - z_i)} \cdot (-1) \cdot \sigma(x) - (1 - \sigma(x)) \cdot \frac{\partial z_i}{\partial w_j}$$

Derivative of Loss Function (3/3)

미분 결과를 손실 함수에 대입

$$L(W) = -\frac{1}{N}\sum_{i=1}^{N}[y_i \cdot \sigma(z_i) + (1 - y_\mathrm{i}) \cdot \log(1 - \sigma(z_i))]$$

$$\frac{\partial L(W)}{\partial w_j} = -\frac{1}{N}\sum_{i=1}^{N}\frac{\partial}{\partial w_j}[y_i \cdot \sigma(z_i) + (1 - y) \cdot \log(1 - \sigma(z_i))]$$

$$\frac{\partial L(W)}{\partial w_j} = -\frac{1}{N}\sum_{i=1}^{N}\left[y_i \cdot \frac{\sigma(\mathrm{z_i})\big(1 - \sigma(\mathrm{z_i})\big)}{\sigma(\mathrm{z_i})}\frac{\partial z_i}{\partial w_j} + (1 - y)\frac{\sigma(\mathrm{z_i})\big(1 - \sigma(\mathrm{z_i})\big)}{1 - \sigma(\mathrm{z_i})}\frac{\partial z_i}{\partial w_j}\right]$$

불필요 항 약분, 항을 간단히 정리

$$\frac{\partial L(W)}{\partial w_j} = -\frac{1}{N}\sum_{i=1}^{N}\big(y_i - \sigma(\mathrm{z_i})\big)\frac{\partial z_i}{\partial w_j} = -\frac{1}{N}\sum_{i=1}^{N}\big(y_i - \sigma(W^T X_i)\big)x_{ij}$$

Terminaltion Condition

Repeat until the terminaltion condition is satisfied

- Fix the number of updates.
- Update the parameters until the desired performance is achieved.
- Update the parameters until the norm of the gradient (partial derivative vector) falls below a threshold. (If the parameter updates are below a meaningful threshold, stop the optimization.)

Learning Method Comparison

Aspect	Batch Learning	Mini-batch Learning
Data Unit	Entire dataset	Small batches (e.g., 32, 64 samples)
Memory Usage	High	Moderate
Update Frequency	Once per epoch	Once per mini-batch
Convergence Speed	Slow but stable	Fast and efficient
Stability	Very stable	Relatively stable
Computational Efficiency	Lower (due to large dataset)	High (GPU-friendly)
Best Use Case	Small datasets that fit in memory	Standard deep learning
Examples	Linear regression, full batch training	CNN, RNN training

In the Next Lecture

- **We will explore real world problem**
  - Practice & Exercise!
  - Have a fun!

수고하셨습니다 ..^^..
Thank you!