Basic of Data Analytics

Data Scientist
안건이

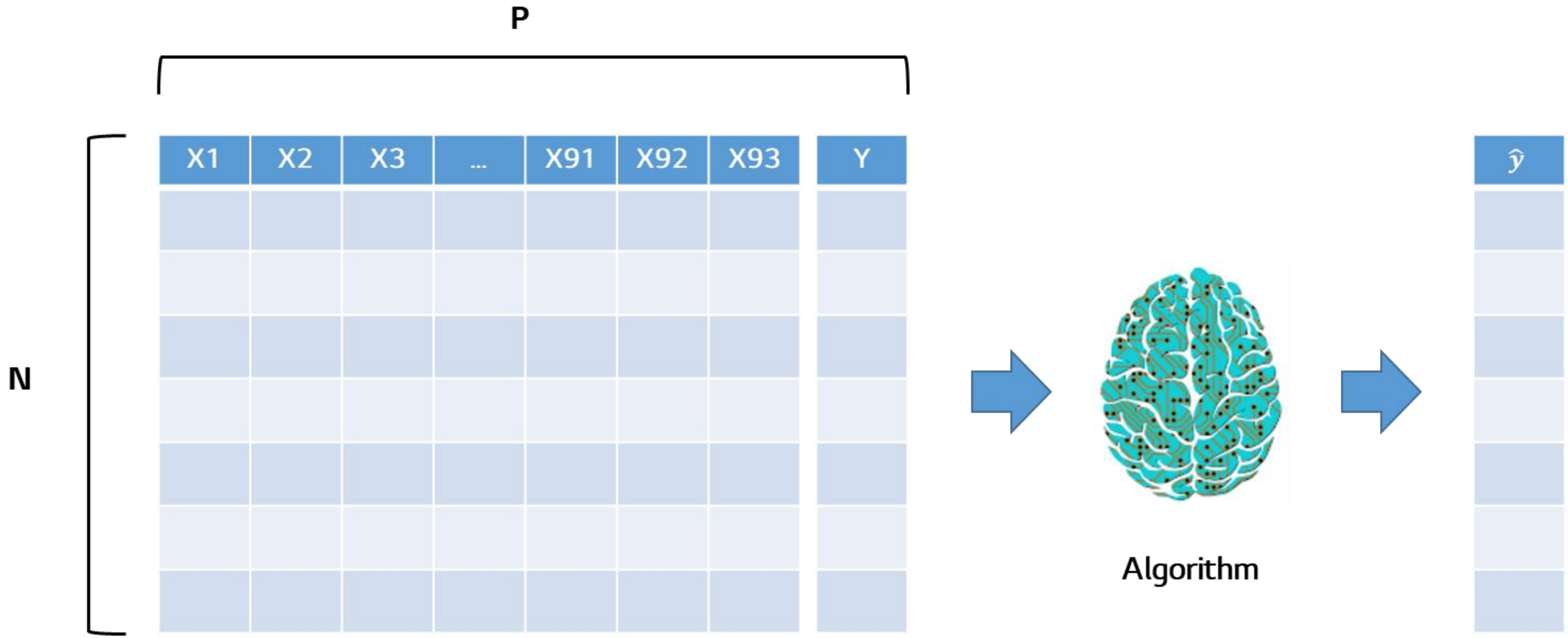
목차

Error = Bias vs Variance

- 정형 데이터 : N x P, Table 데이터

- X : Variables, Features, Columns, 독립변수, 설명변수
  - Numerical : 연속적인 변수 (2는 1보다 크다, 변수에 대/소가 의미 있음)
  - Categorical : 이분적인 변수 (2는 1과 다름, 변수에 대/소가 의미 없음)

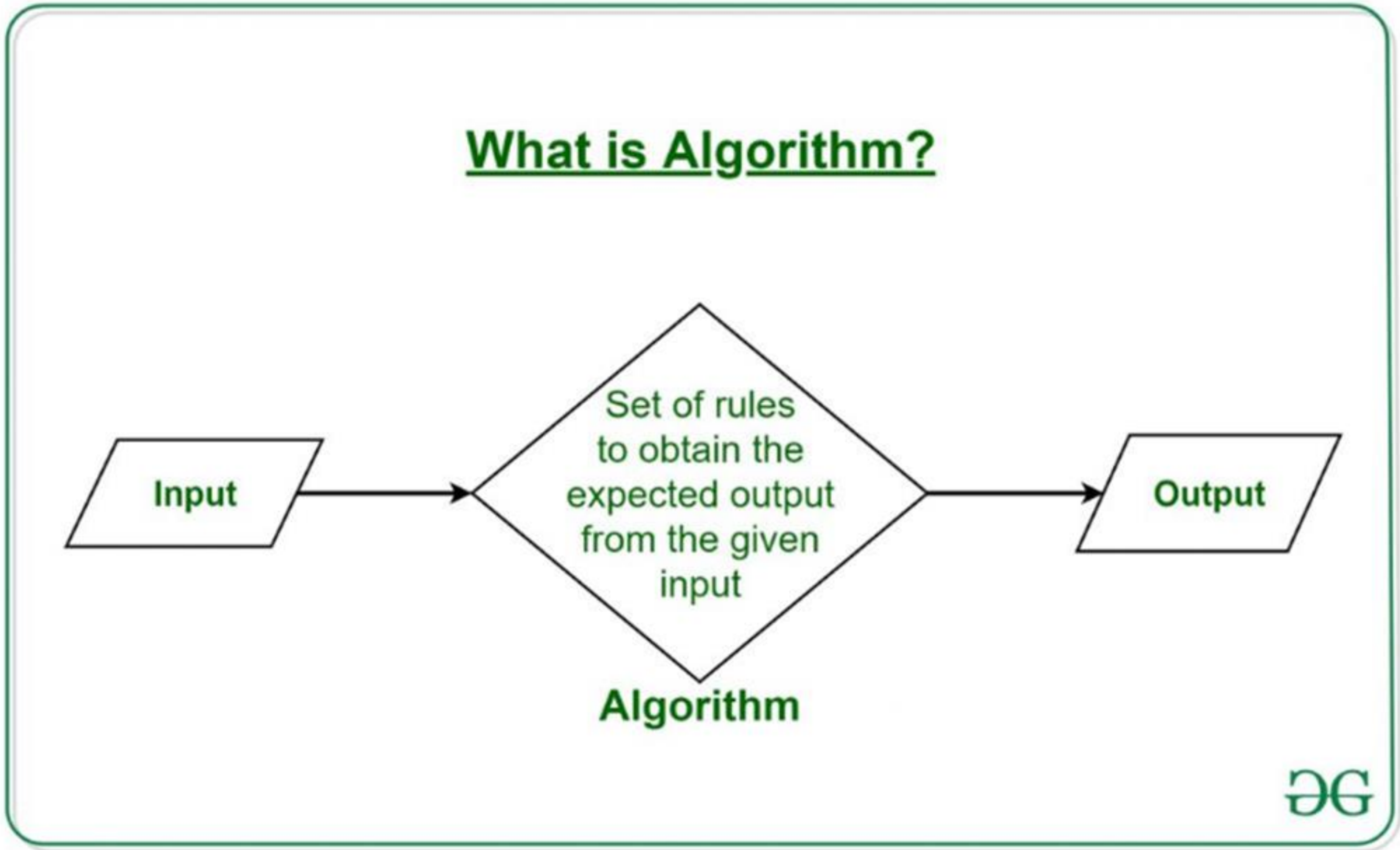
- Y : Labels, 종속변수
  - Numerical : Regression, 회귀모형
  - Categorical : Classification, 분류모형



Error = Bias vs Variance

- Algorithm = Input → Process → Output
  - Model

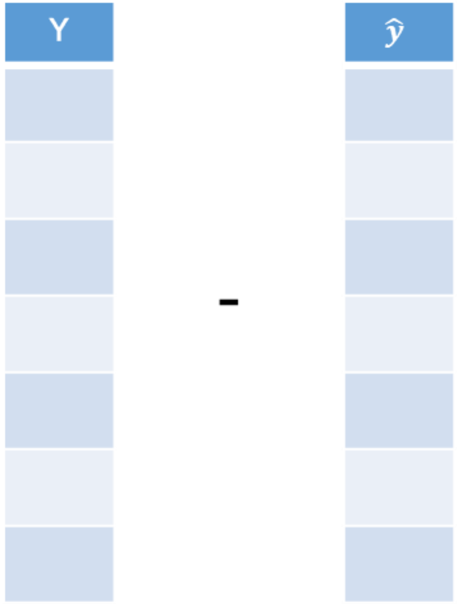
- 좋은 알고리즘의 기준은 ?
  - Error가 낮은 Algorithm



Error = Bias vs Variance

- Error가 낮은 Algorithm (Model)이 데이터의 패턴을 잘 학습한 것
- Square의 이유

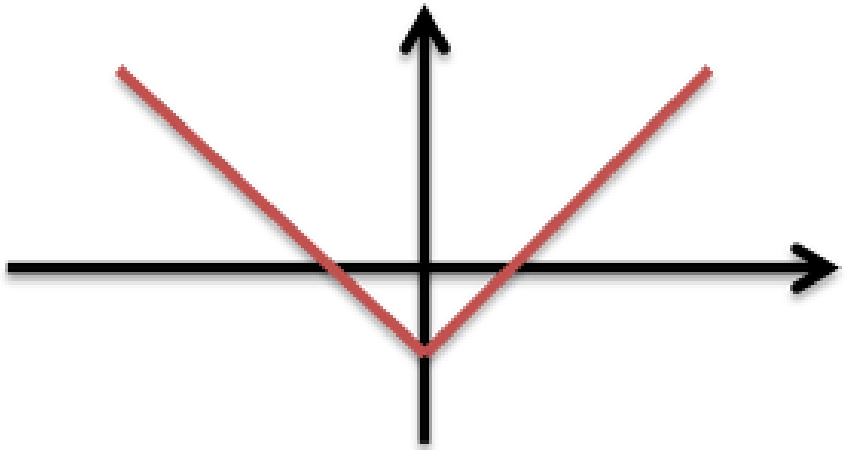
Error =

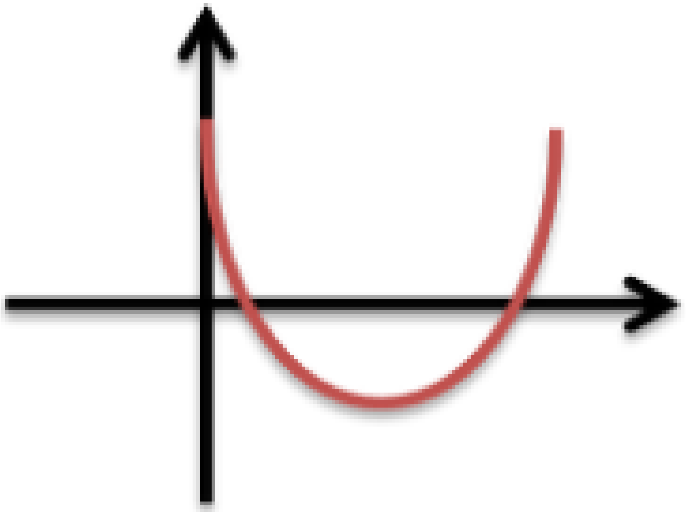


$$\text{Error} = \sum(\hat{y} - y)^2$$

Error = Bias vs Variance

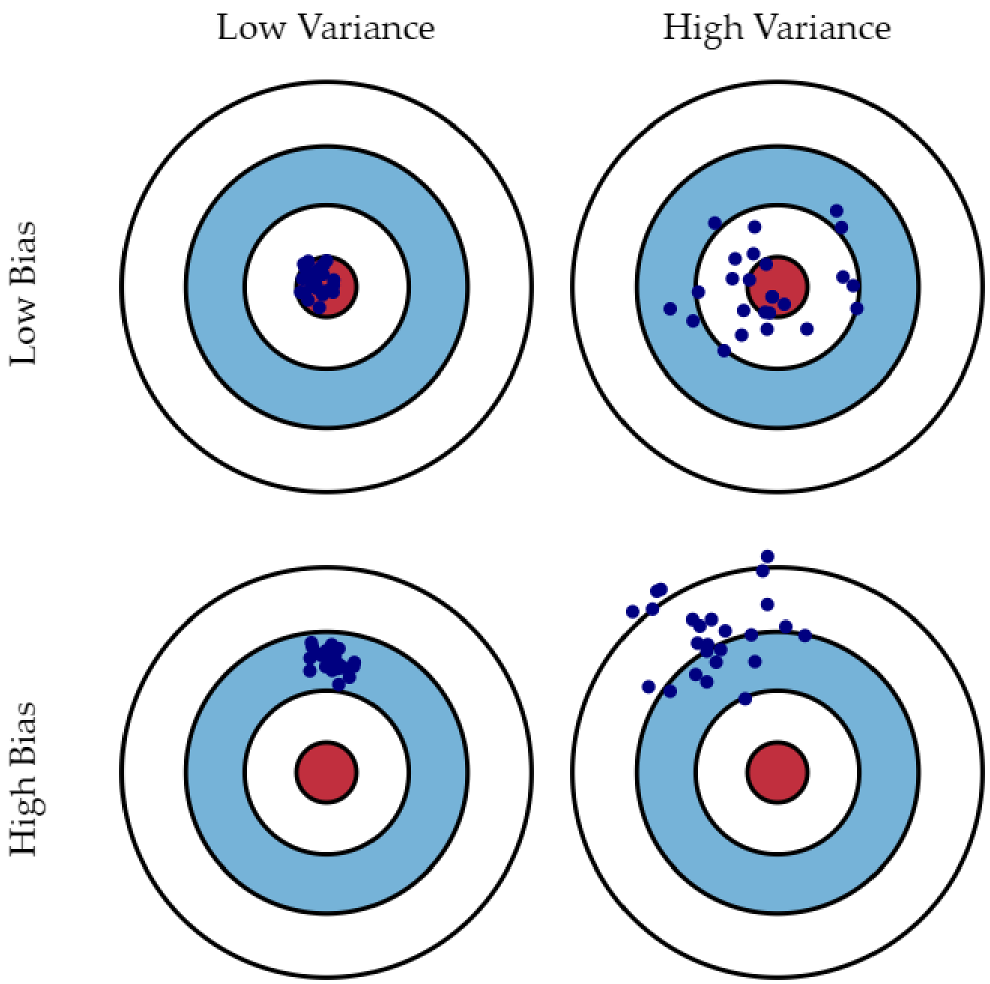
- Error가 낮은 Algorithm (Model)이 데이터의 패턴을 잘 학습한 것
- Square의 이유





Error = Bias vs Variance

- Error(X) = Noise(X) + Bias(X) + Variance(X)
  - Noise(X) : 데이터가 본질적으로 품고 있는 한계점
    - 극복 방법 : 정확한 Data Preprocessing
  - Bias(X) 와 Variance(X) : 모델에 따라 변하는 한계점
    - 극복 방법 : 상황에 맞는 알고리즘 선택, 정확한 검증 방법 선택



Error = Bias vs Variance

- Bias(X)
  - 추정 값의 평균과 참 값들 간의 차이
- Variance(X)
  - 추정 값의 평균과 추정 값들 간의 차이
- Bias는 참 값과 추정 값의 거리를 의미, Variance는 추정 값들의 흩어진 정도
  - Variance는 loss를 의미하지만, 참 값과는 관계없이 추정 값들의 흩어진 정도만을 의미함

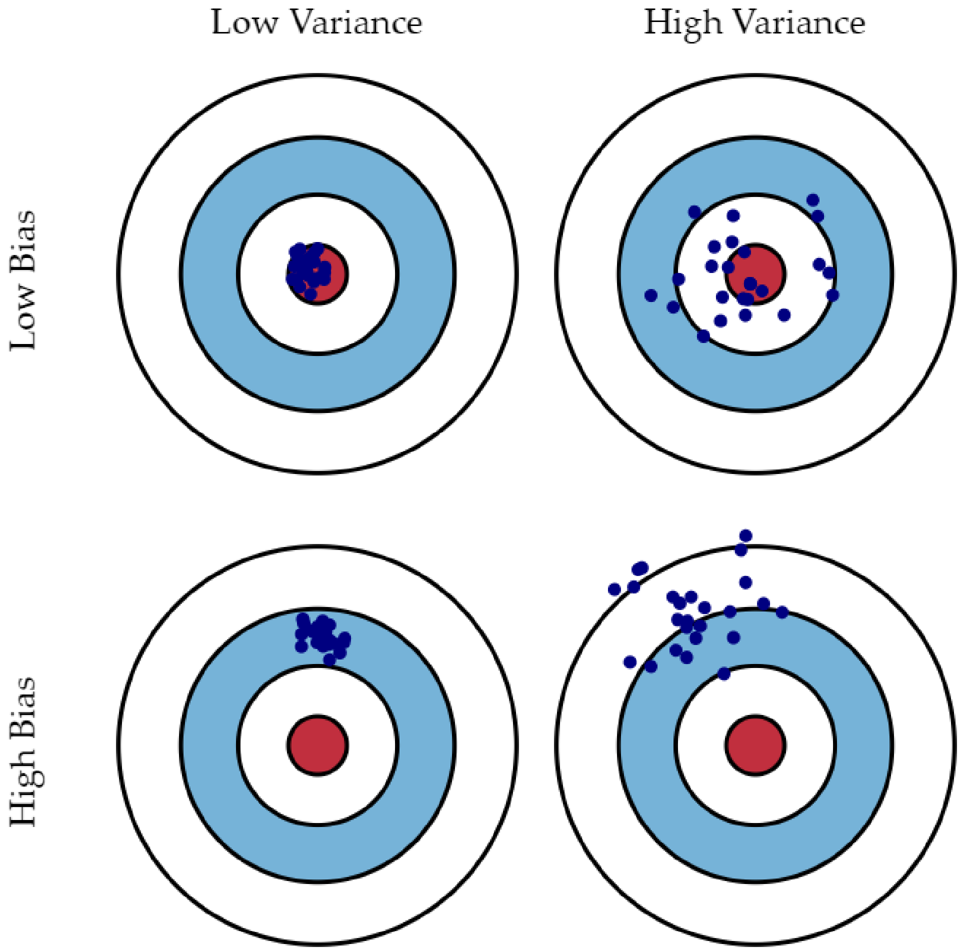
$$
\begin{aligned}
\mathrm{MSE}(\hat\theta) &= \mathrm{E}_\theta\left[(\hat\theta-\theta)^2\right] \\
&= \mathrm{E}_\theta\left[\left(\hat\theta-\mathrm{E}_\theta[\hat\theta]+\mathrm{E}_\theta[\hat\theta]-\theta\right)^2\right] \\
&= \mathrm{E}_\theta\left[\left(\hat\theta-\mathrm{E}_\theta[\hat\theta]\right)^2+2\left(\hat\theta-\mathrm{E}_\theta[\hat\theta]\right)\left(\mathrm{E}_\theta[\hat\theta]-\theta\right)+\left(\mathrm{E}_\theta[\hat\theta]-\theta\right)^2\right] \\
&= \mathrm{E}_\theta\left[\left(\hat\theta-\mathrm{E}_\theta[\hat\theta]\right)^2\right]+\mathrm{E}_\theta\left[2\left(\hat\theta-\mathrm{E}_\theta[\hat\theta]\right)\left(\mathrm{E}_\theta[\hat\theta]-\theta\right)\right]+\mathrm{E}_\theta\left[\left(\mathrm{E}_\theta[\hat\theta]-\theta\right)^2\right] \\
&= \mathrm{E}_\theta\left[\left(\hat\theta-\mathrm{E}_\theta[\hat\theta]\right)^2\right]+2\left(\mathrm{E}_\theta[\hat\theta]-\theta\right)\mathrm{E}_\theta\left[\hat\theta-\mathrm{E}_\theta[\hat\theta]\right]+\left(\mathrm{E}_\theta[\hat\theta]-\theta\right)^2 && \mathrm{E}_\theta[\hat\theta]-\theta=\text{const.} \\
&= \mathrm{E}_\theta\left[\left(\hat\theta-\mathrm{E}_\theta[\hat\theta]\right)^2\right]+2\left(\mathrm{E}_\theta[\hat\theta]-\theta\right)\left(\mathrm{E}_\theta[\hat\theta]-\mathrm{E}_\theta[\hat\theta]\right)+\left(\mathrm{E}_\theta[\hat\theta]-\theta\right)^2 && \mathrm{E}_\theta[\hat\theta]=\text{const.} \\
&= \mathrm{E}_\theta\left[\left(\hat\theta-\mathrm{E}_\theta[\hat\theta]\right)^2\right]+\left(\mathrm{E}_\theta[\hat\theta]-\theta\right)^2 \\
&= \boxed{\mathrm{Var}_\theta(\hat\theta)+\mathrm{Bias}_\theta(\hat\theta,\theta)^2}
\end{aligned}
$$

Alternatively, we have

$$
\begin{aligned}
\mathbb{E}(\theta-\hat\theta)^2 &= \mathbb{E}(\hat\theta^2)+\mathbb{E}(\theta^2)-2\theta\mathbb{E}(\hat\theta) \\
&= \mathrm{Var}(\hat\theta)+(\mathbb{E}\hat\theta)^2+\theta^2-2\theta\mathbb{E}(\hat\theta) \\
&= \mathrm{Var}(\hat\theta)+(\mathbb{E}\hat\theta-\theta)^2 \\
&= \boxed{\mathrm{Var}(\hat\theta)+\mathrm{Bias}^2(\hat\theta)}
\end{aligned}
$$

Error = Bias vs Variance

- Bias(X)
  - 추정 값의 평균과 참 값들 간의 차이
- Variance(X)
  - 추정 값의 평균과 추정 값들 간의 차이
- Bias는 참 값과 추정 값의 거리를 의미, Variance는 추정 값들의 흩어진 정도
  - Variance는 loss를 의미하지만, 참 값과는 관계없이 추정 값들의 흩어진 정도만을 의미함

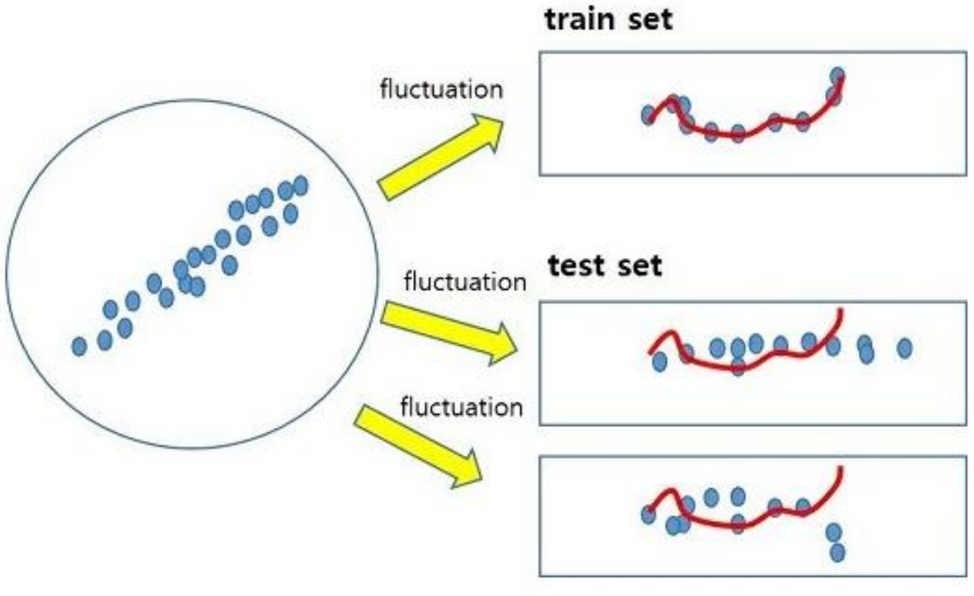


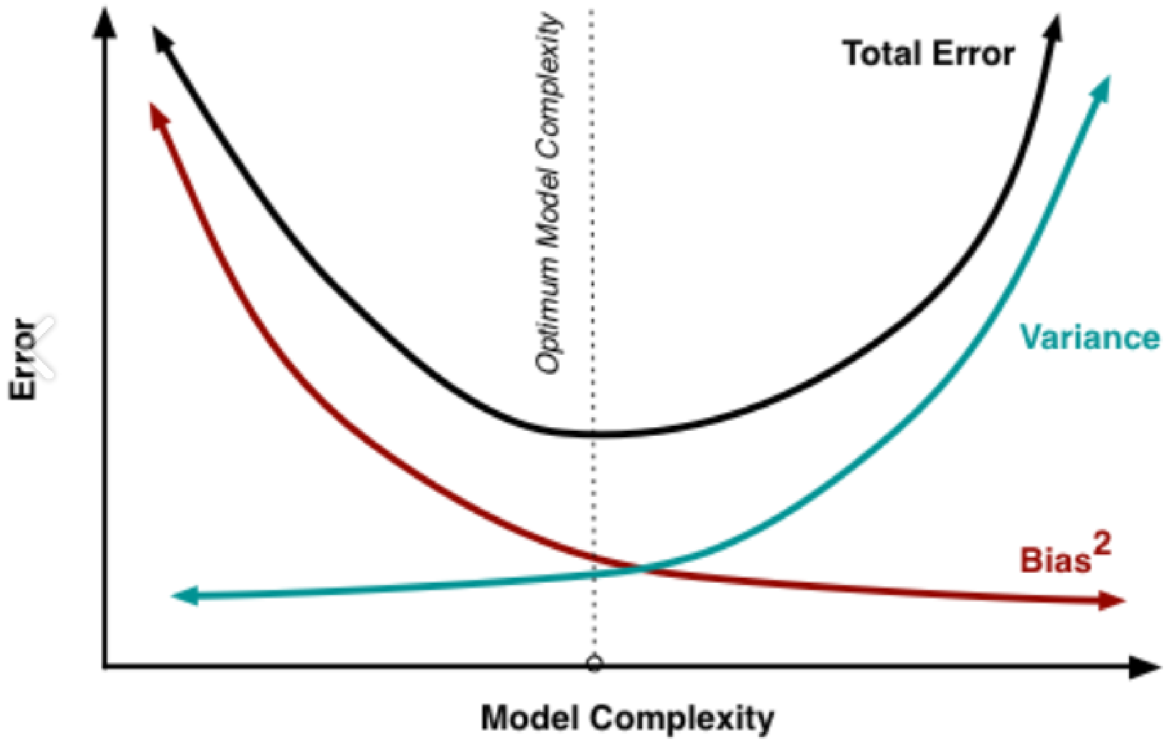
Overfitting vs Underfitting

- Train Data vs Test Data
    - Bias와 Variance의 Trade-Off 관계

$$\text{Error}(X) = \underline{\text{Noise}(X)} + \underline{\text{Bias}(X)} + \underline{\text{Variance}(X)}$$

Data Preprocessing　　Model Complexity ↑　　Model Complexity ↓



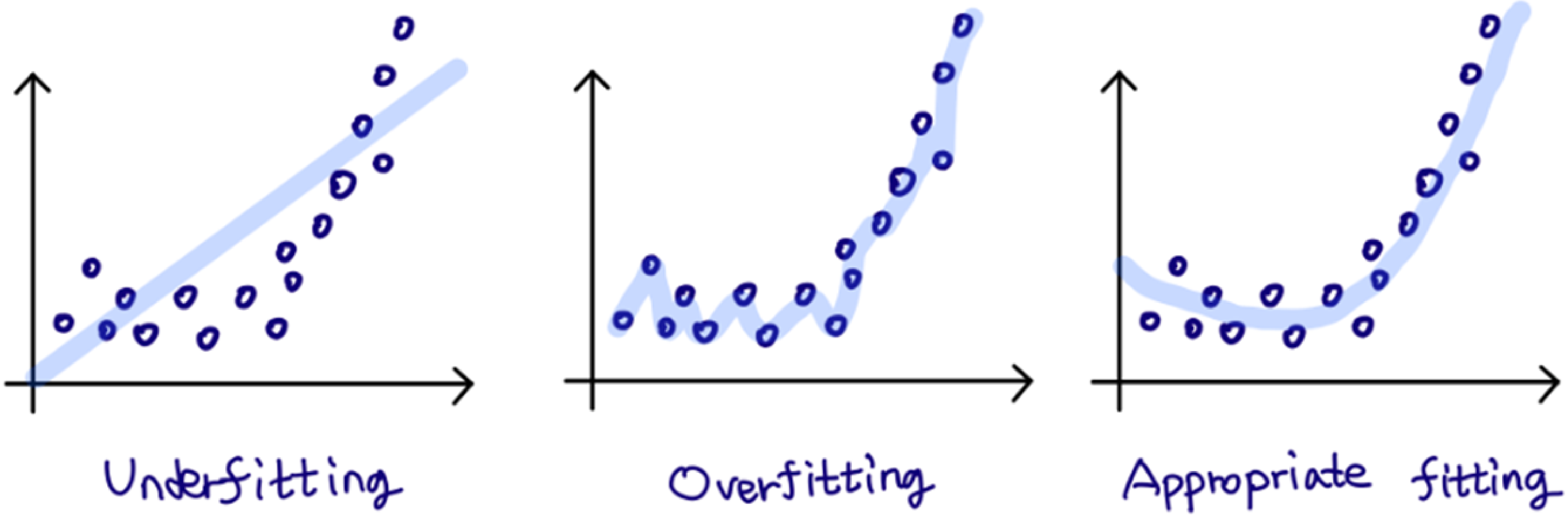


Overfitting vs Underfitting

- Train Data vs Test Data
  - Bias와 Variance의 Trade-Off 관계

$$\underline{\mathrm{Error}(X)} = \underline{\mathrm{Noise}(X)} + \underline{\mathrm{Bias}(X)} + \underline{\mathrm{Variance}(X)}$$

Data Preprocessing | Model Complexity ↑ | Model Complexity ↓



Bias ↑ Variance ↓ | Bias ↓ Variance ↑

Overfitting vs Underfitting

- Underfitting
  - 학습 데이터 내의 모든 정보를 고려하지 못하고 있음 (High Bias)
  - 검증 데이터에 대해 모델의 형태는 크게 변하지 않음 (Low Variance)

- Overfitting
  - 학습 데이터 내의 정보를 잘 설명하고 있음(Low Bias)
  - 검증 데이터에 대해 완전히 다른 형태로 변하게 되고, Generality를 잃게 됨 (High Variance)

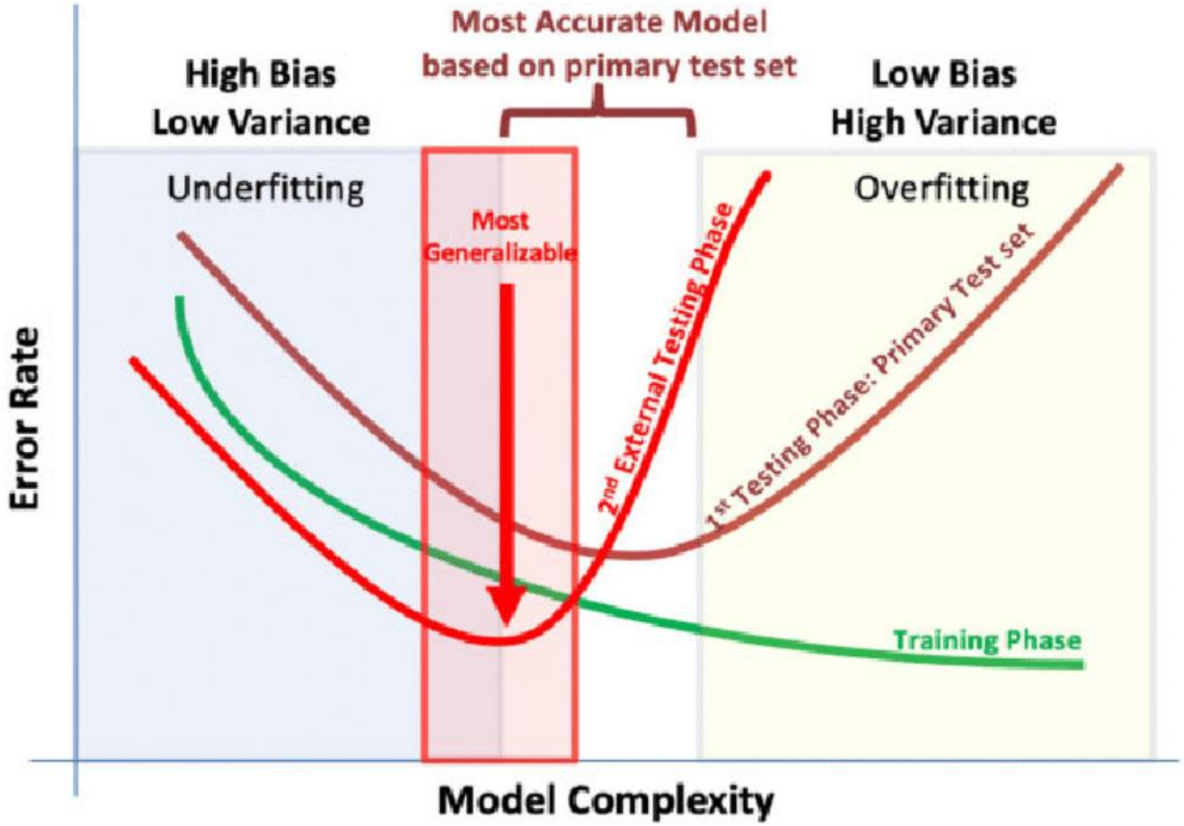
Bias ↑ | Bias ↓
Variance ↓ | Variance ↑

Overfitting vs Underfitting

- Train Data vs Test Data
  - Bias와 Variance의 Trade-Off 관계

$$\text{Error}(X) = \underline{\text{Noise}(X)} + \underline{\text{Bias}(X)} + \underline{\text{Variance}(X)}$$

Data Preprocessing　　Model Complexity ↑　　Model Complexity ↓



Overfitting vs Underfitting

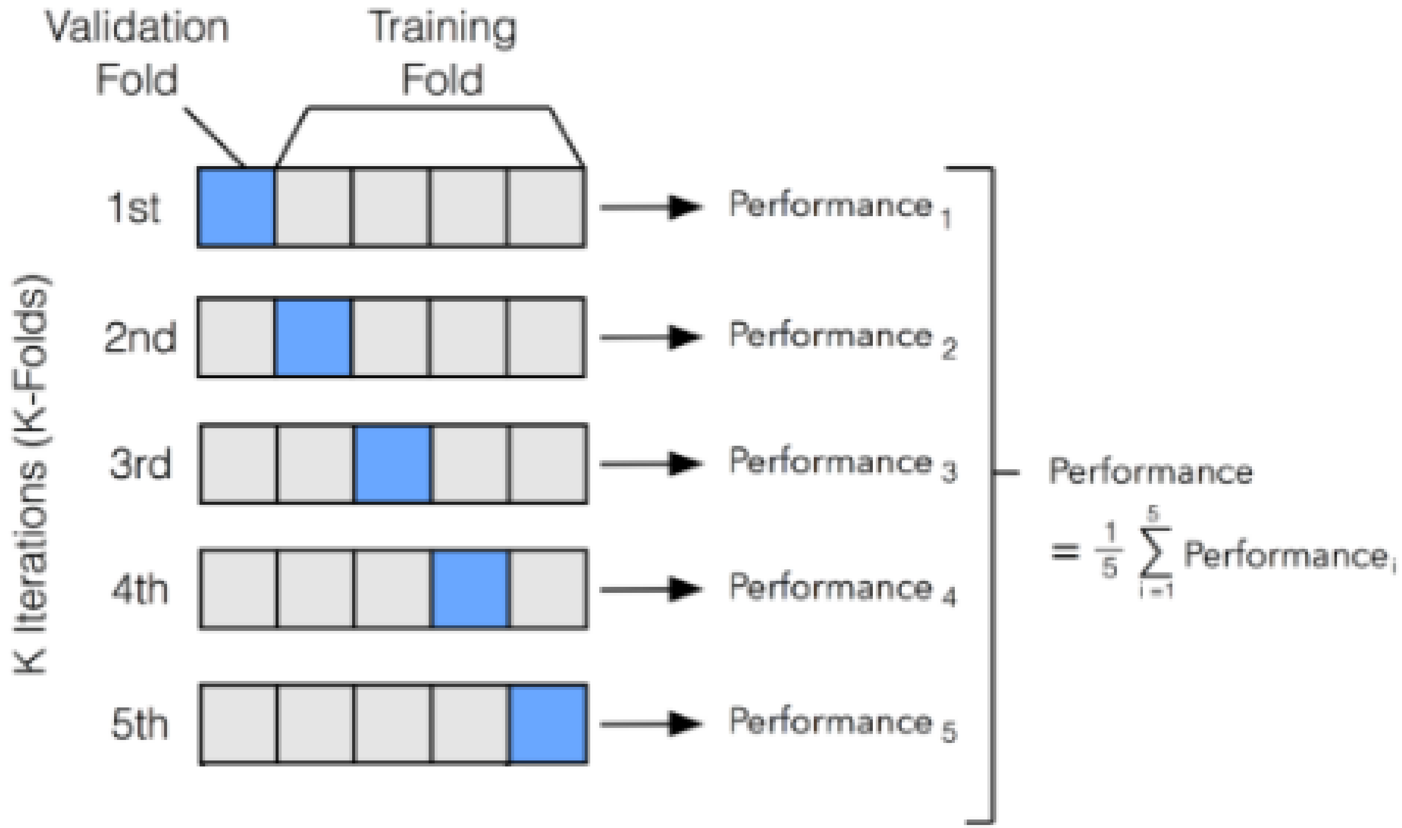
- Train Data vs Test Data
    - Bias와 Variance의 Trade-Off 관계

그래서 Overfitting 나쁜거야 ???

원인분석 : Overfitting Is Okay
예측문제 : Overfitting is not Okay
상황에 알맞게 진행

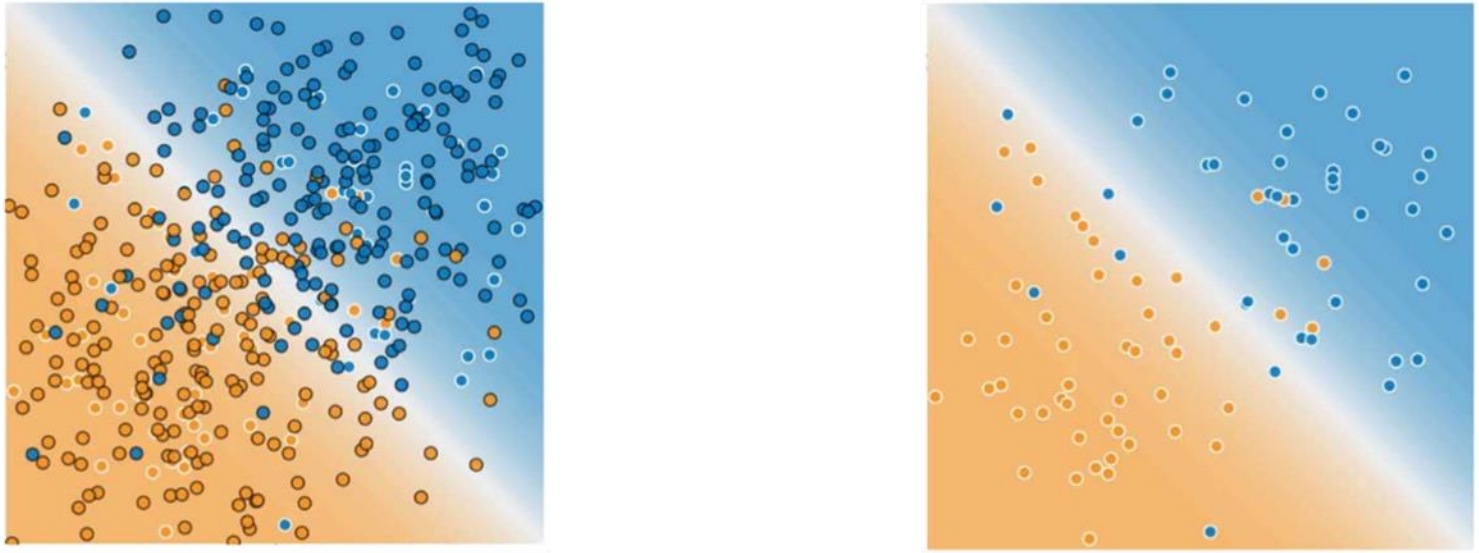
K-Fold Cross Validation

- K-Fold Cross Validation
  - Bias와 Variance의 Trade-Off 관계에서 최적의 Loss 값을 찾기 위함
  - 전체 데이터에 대해 검증할 수 있는 기법
  - Data Size가 크지 않을 때 사용하는 기법
  - Model의 Hyperparameter가 많지 않을 때 사용하는 기법
    - Hyperparameter는 Model의 Complexity를 조절할 수 있는 변수
    - 적게는 1개에서 많게는 10개 이상인 경우가 존재함



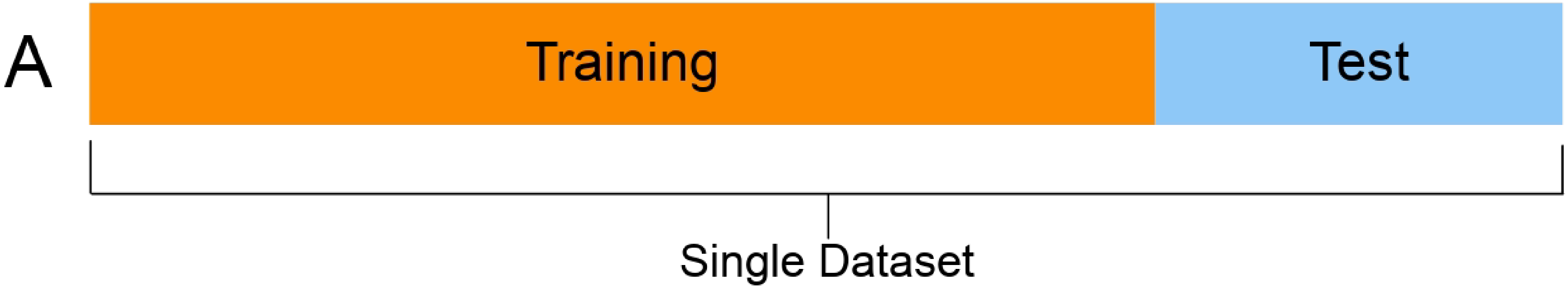
Model Validation

- 현실적인 “Big Data” + “# of Hyperparameter”이 많은 Model에서는 Random Sampling



Training　　　　　Test

- Time Series Data의 경우 Time으로 Sorting 한 후 시간대로 잘라서 검증함



Loss Function

- 모든 Model(Algorithm)은 Loss Function을 가지고 있음
  - 크게는 Regression, Classification

Regression VS **Classification**

Y : Numeric

Y : Categorical

X1	X2	X3	...	X91	X92	X93	Y

Loss Function

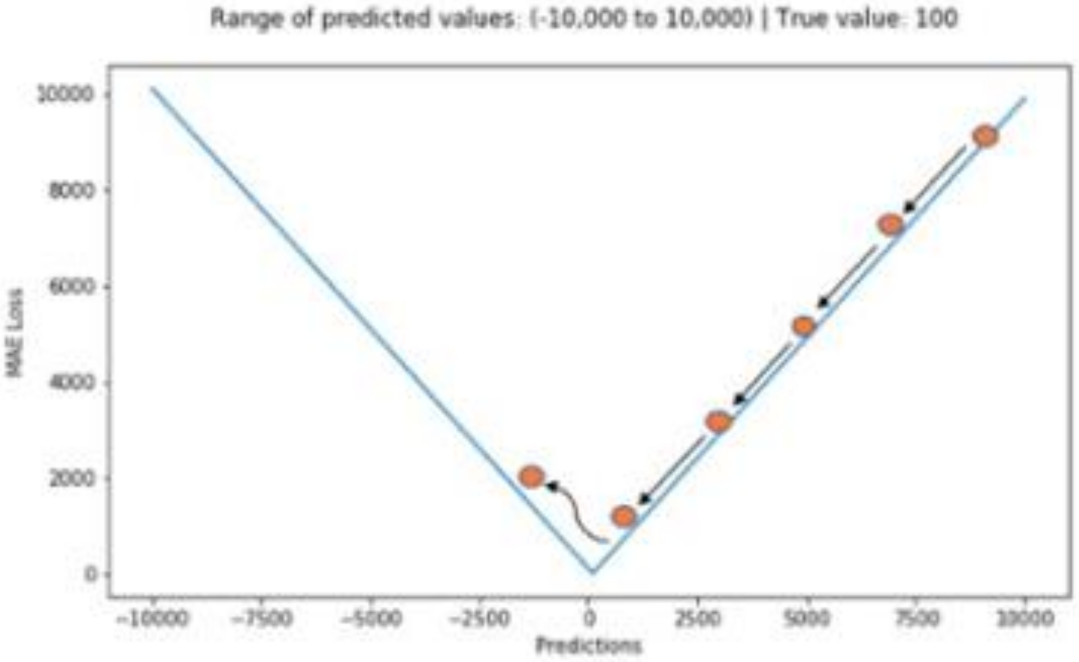
- Regression Loss Function

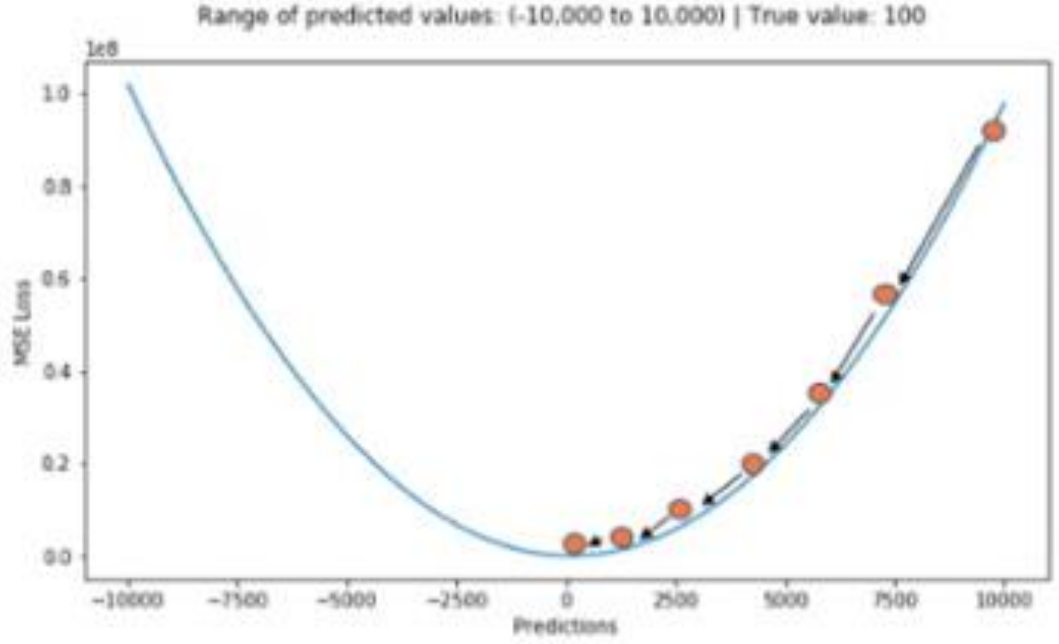
미분 불가능

$$MAE = \frac{1}{N}\sum_{i=1}^{n}\left|\hat{y_i} - y_i\right|$$

미분 가능

$$MSE = \frac{1}{N}\sum_{i=1}^{n}(\hat{y_i} - y_i)^2 \qquad RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{n}(\hat{y_i} - y_i)^2}$$

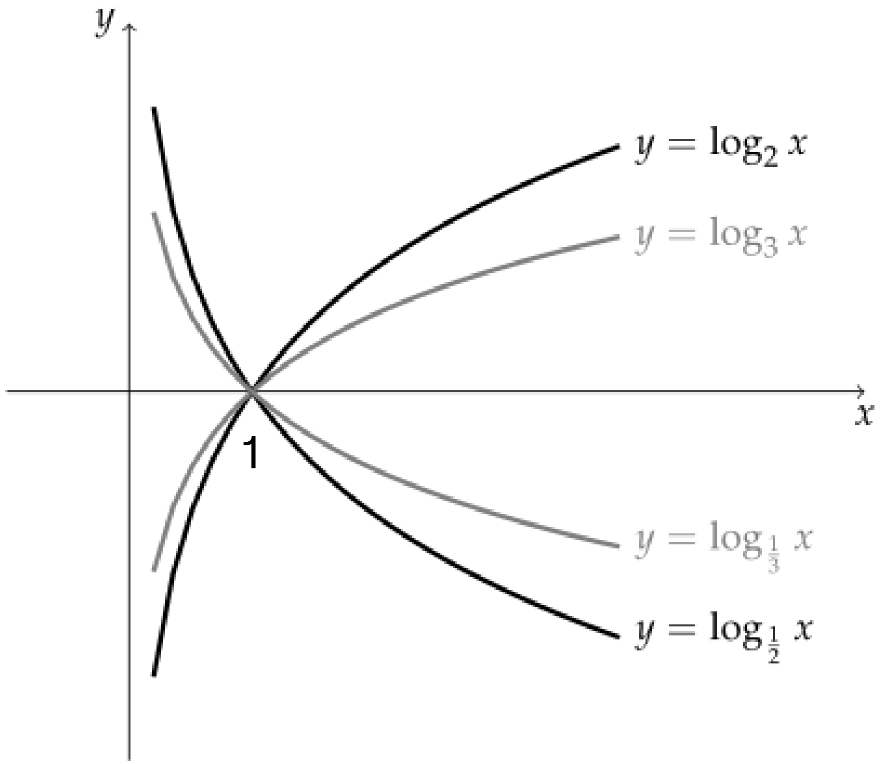


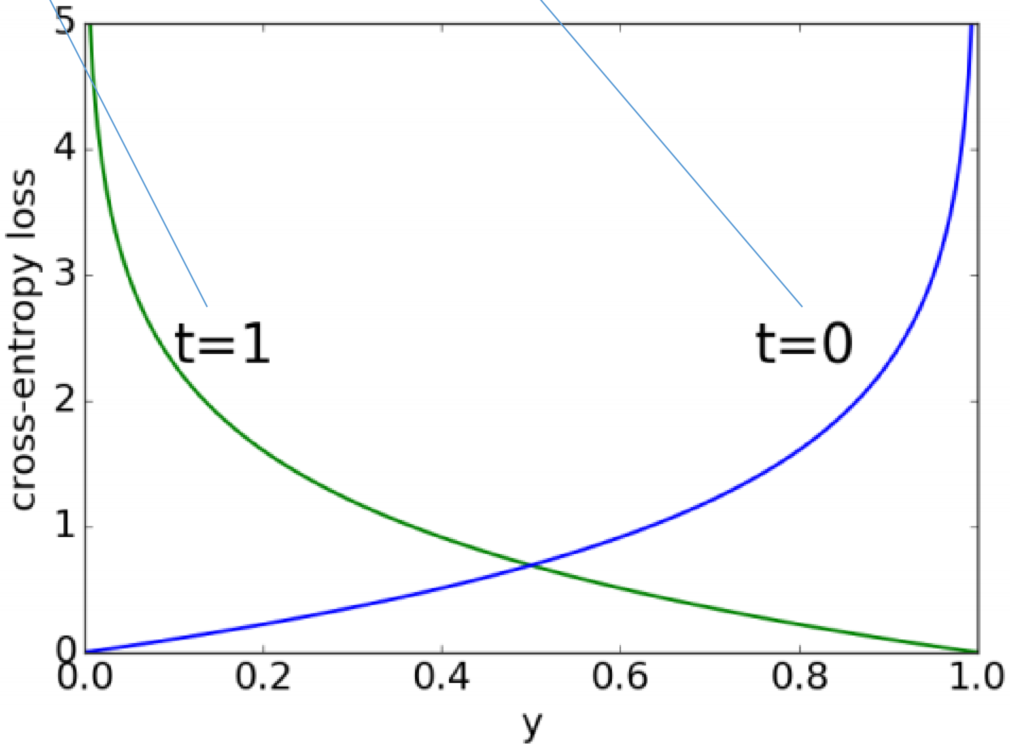


Loss Function

- Classification Loss Function
  - 예측 값을 확률로 뱉음

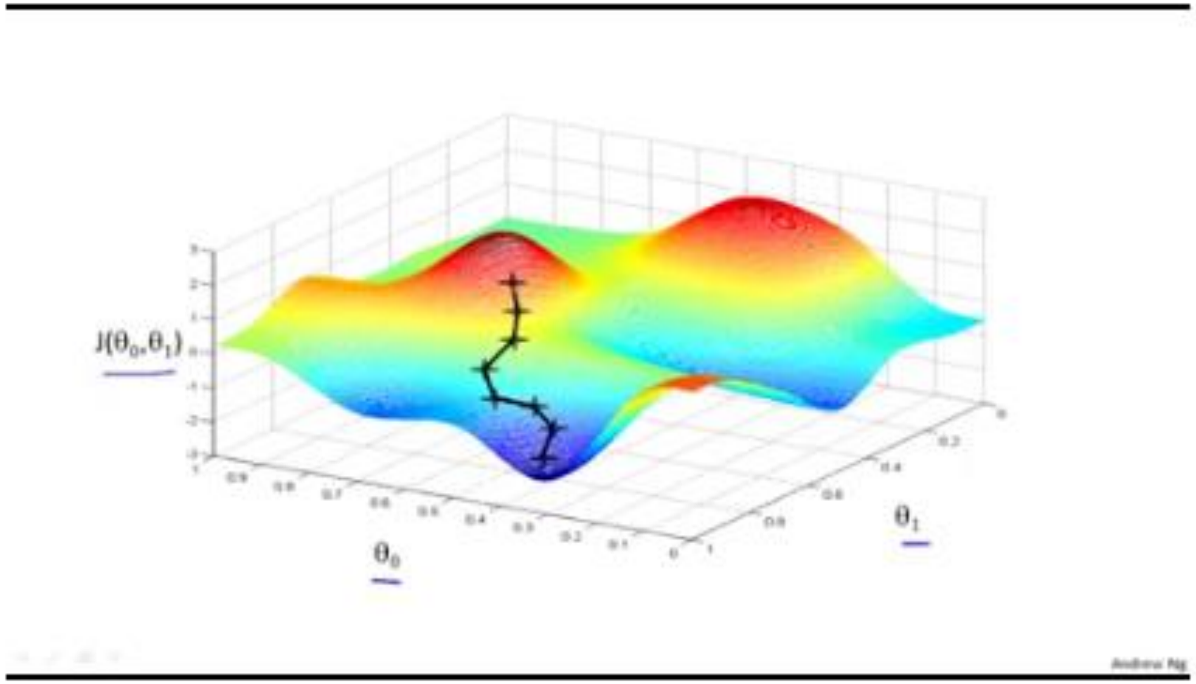
$$BCE = -\frac{1}{N}\sum_{i=0}^{N} y_i \cdot \log(\hat{y_i}) + (1-y_i) \cdot \log(1-\hat{y_i})$$

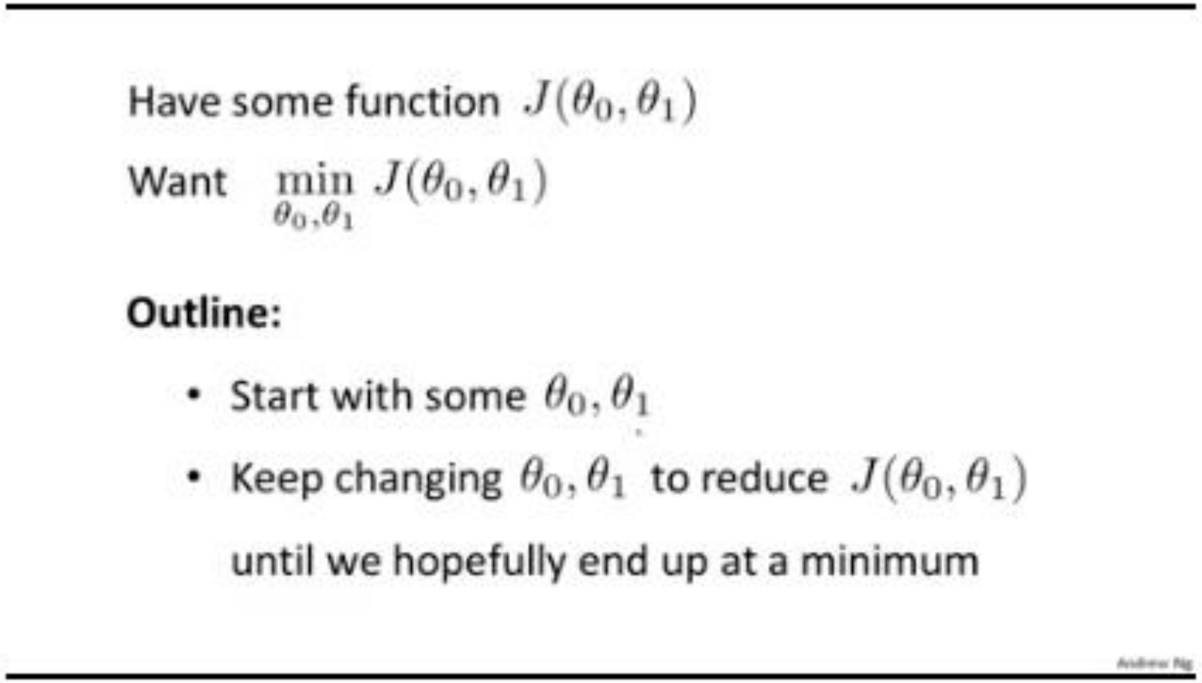




Gradient Descent

- Gradient Descent – 경사하강법
    - Non-convex 경우 Gradient Descent를 활용하여 해(Loss가 가장 낮은)를 찾아 감
    - 대부분의 non-linear regression 문제는 closed form solution이 존재하지 않음
    - Closed form solution이 존재해도 수많은 parameter가 있을때 GD로 해결하는 것이 효율적





Have some function $J(\theta_0, \theta_1)$

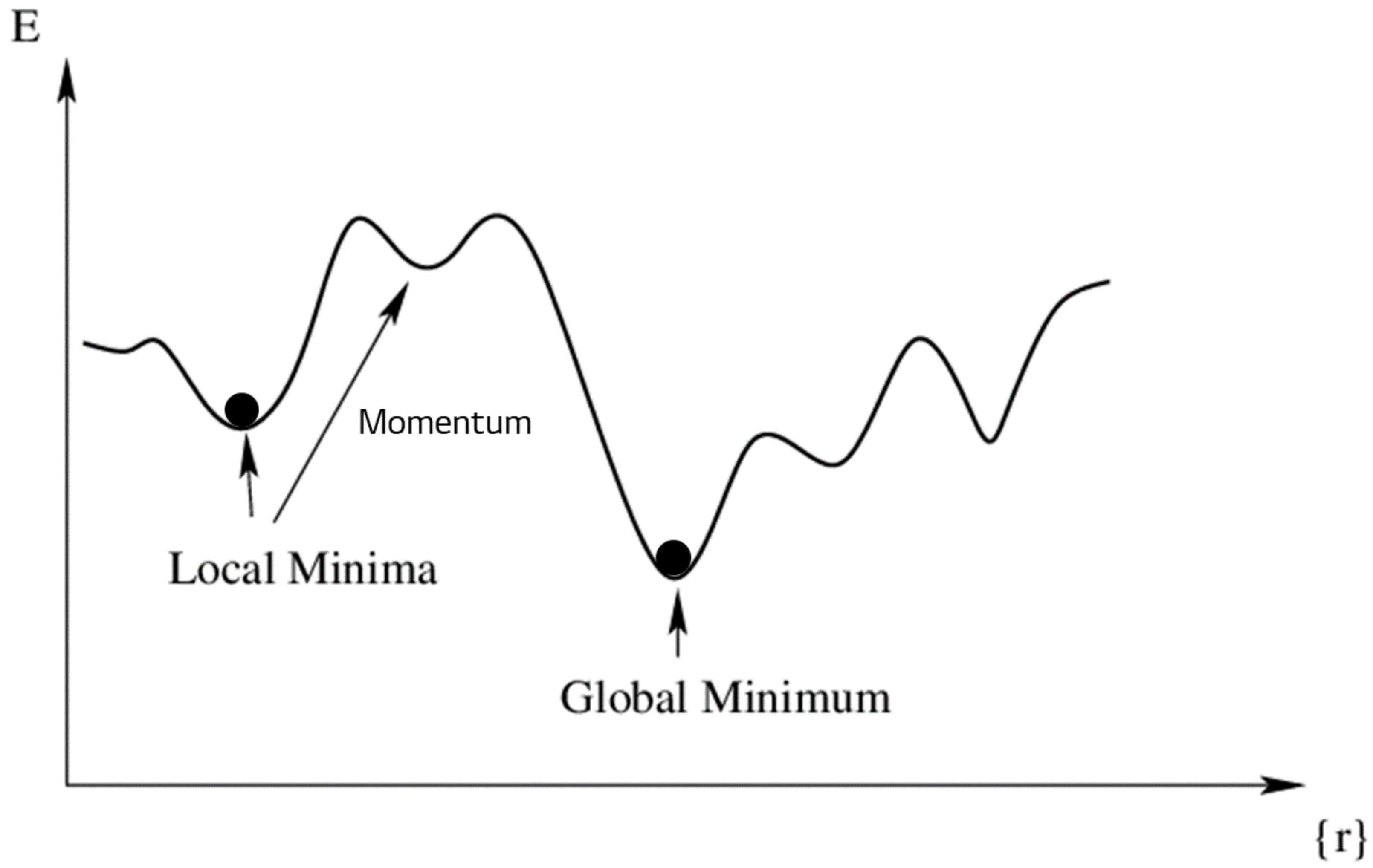
Want $\min_{\theta_0,\theta_1} J(\theta_0, \theta_1)$

Outline:

- Start with some θ_0, θ_1
- Keep changing θ_0, θ_1 to reduce $J(\theta_0, \theta_1)$ until we hopefully end up at a minimum

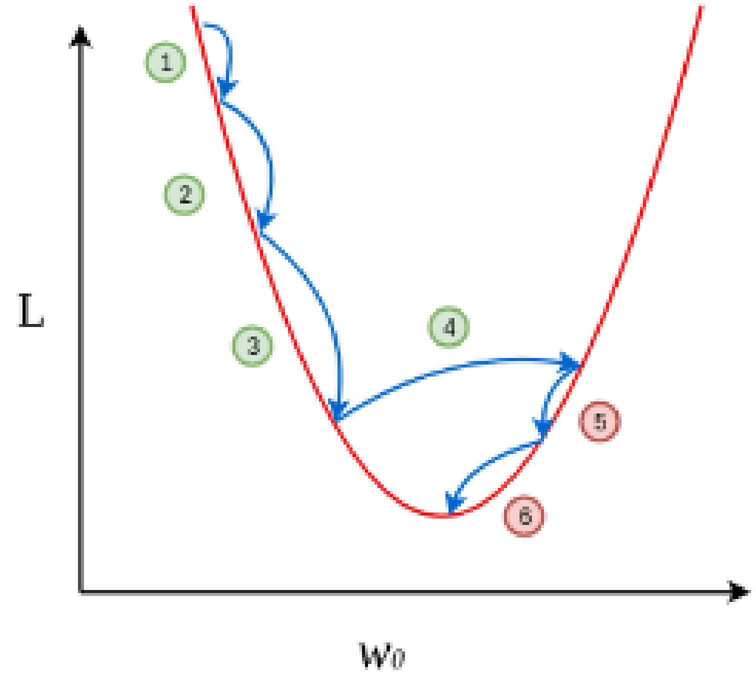
Gradient Descent

- Gradient Descent – 경사하강법
    - Non-convex 경우 Gradient Descent를 활용하여 해(Loss가 가장 낮은)를 찾아 감
    - 대부분의 non-linear regression 문제는 closed form solution이 존재하지 않음
    - Closed form solution이 존재해도 수많은 parameter가 있을때 GD로 해결하는 것이 효율적

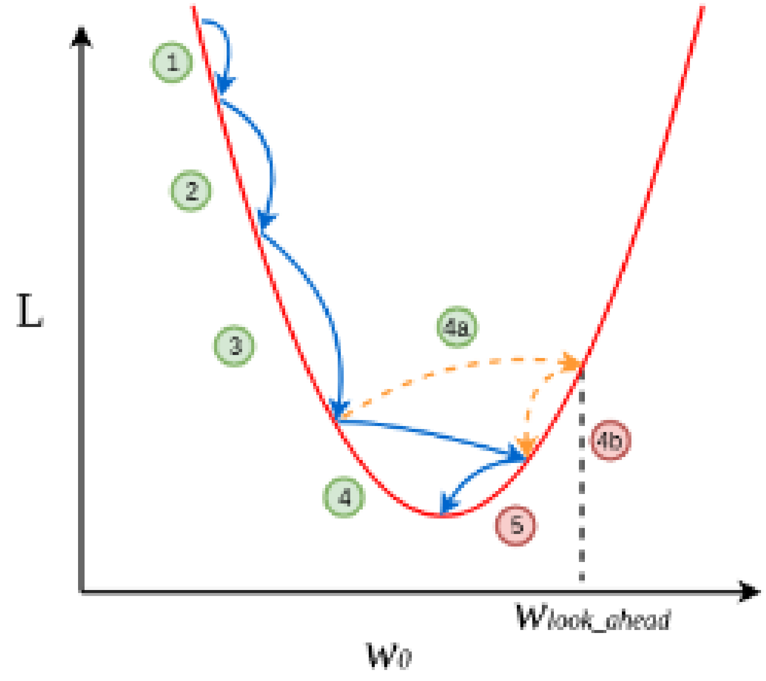


Gradient Descent

- Gradient Descent – 경사하강법
  - Non-convex 경우 Gradient Descent를 활용하여 해(Loss가 가장 낮은)를 찾아 감
  - 대부분의 non-linear regression 문제는 closed form solution이 존재하지 않음
  - Closed form solution이 존재해도 수많은 parameter가 있을때 GD로 해결하는 것이 효율적



(a) Momentum-Based Gradient Descent

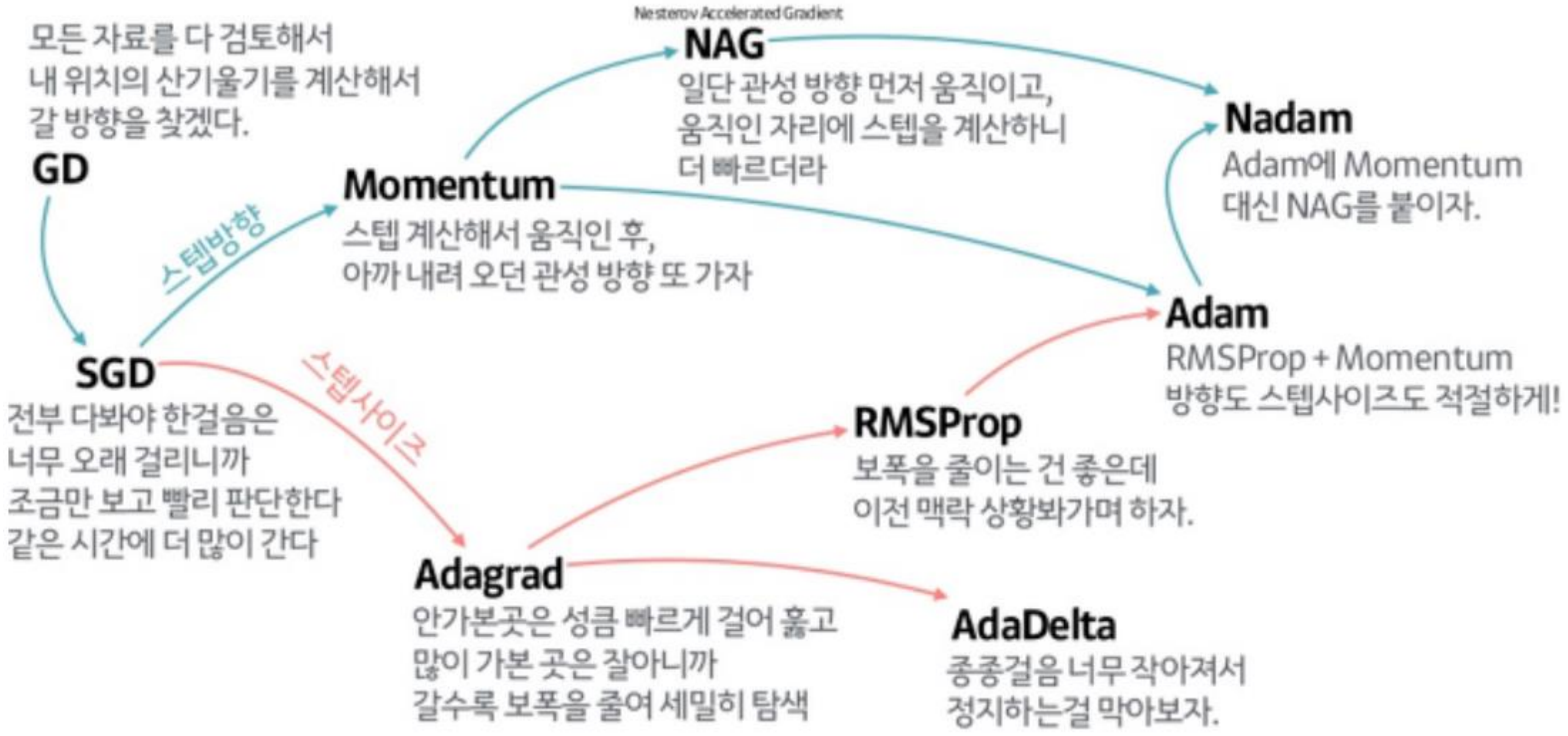


(b) Nesterov Accelerated Gradient Descent

$\Longrightarrow \frac{\partial L}{\partial w_0} = \frac{Negative(-)}{Positive(+)}$ $\quad\Longrightarrow \frac{\partial L}{\partial w_0} = \frac{Negative(-)}{Negative(-)}$

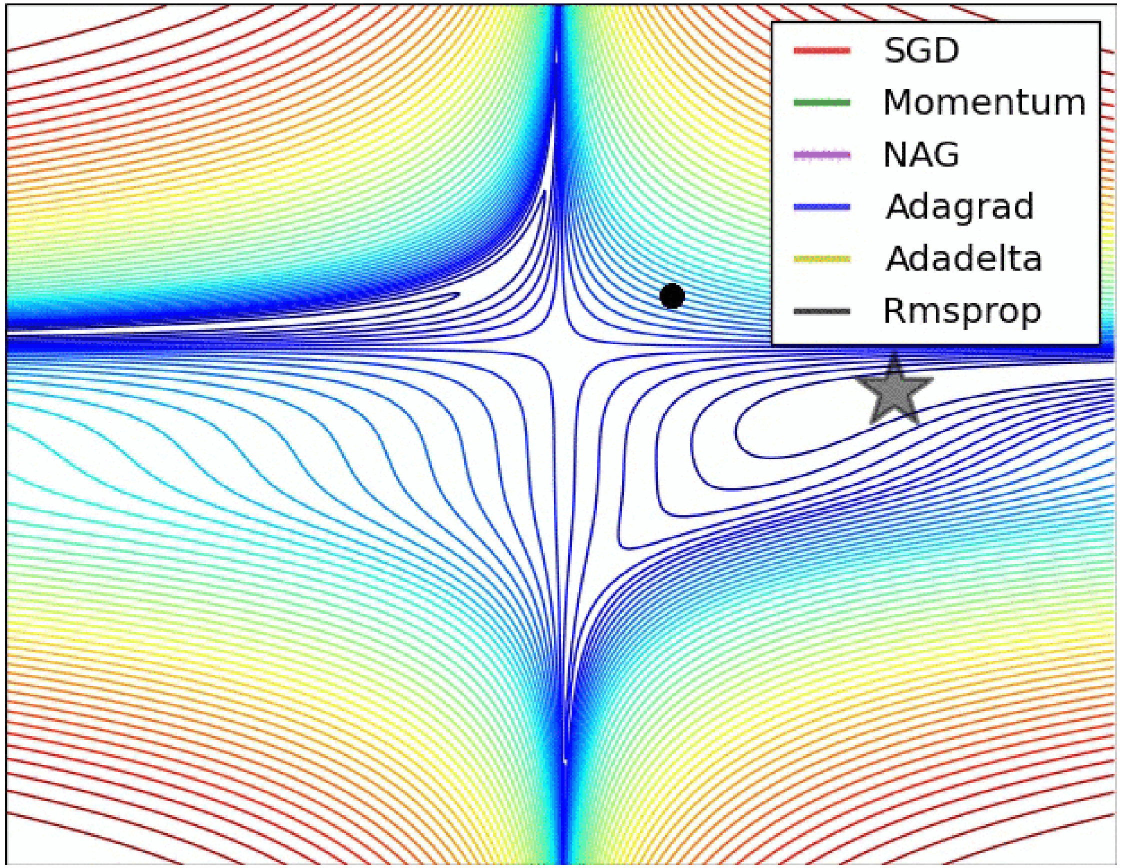
Gradient Descent

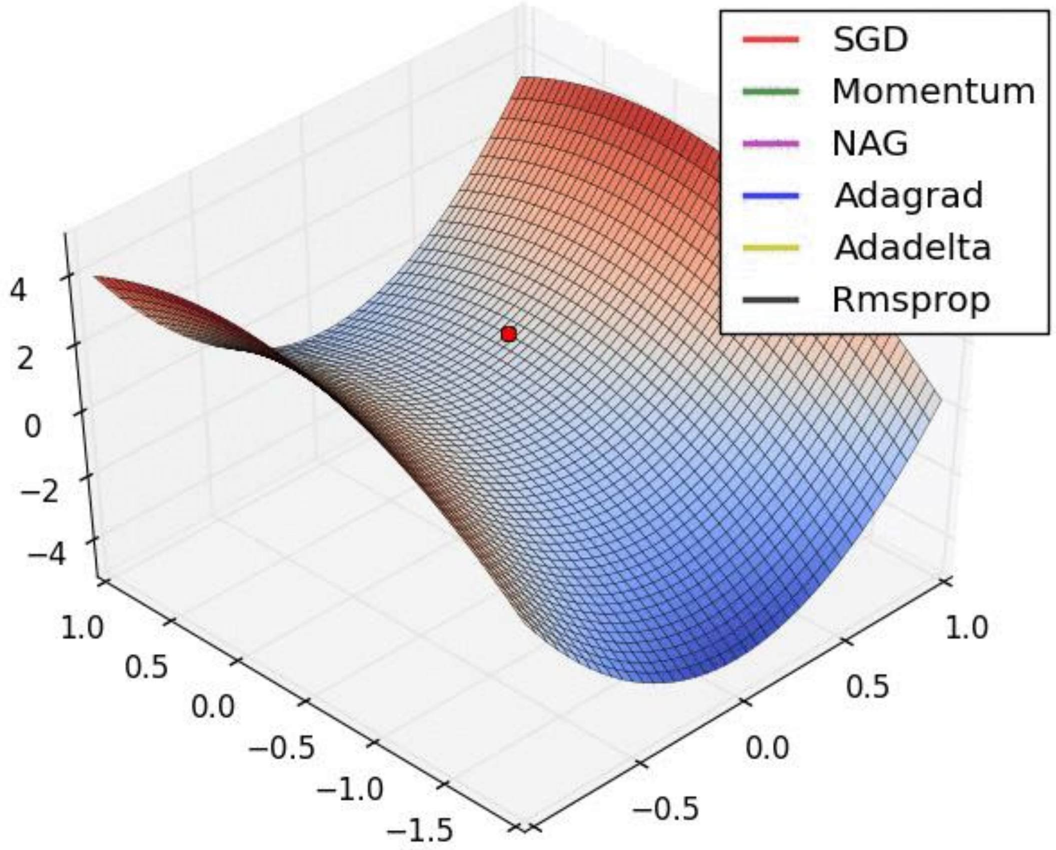
- Gradient Descent – 경사하강법
  - 종류



Gradient Descent

- Gradient Descent – 경사하강법
  - Non-convex 경우 Gradient Descent를 활용하여 해(Loss가 가장 낮은)를 찾아 감
  - Global Optimal을 보장할 수 없음





Model

<u>Regression</u> VS <u>Classification</u>

Y : Numeric   Y : Categorical

더 어려운 문제는 ?

eXplainable 한 건 ?

Q & A