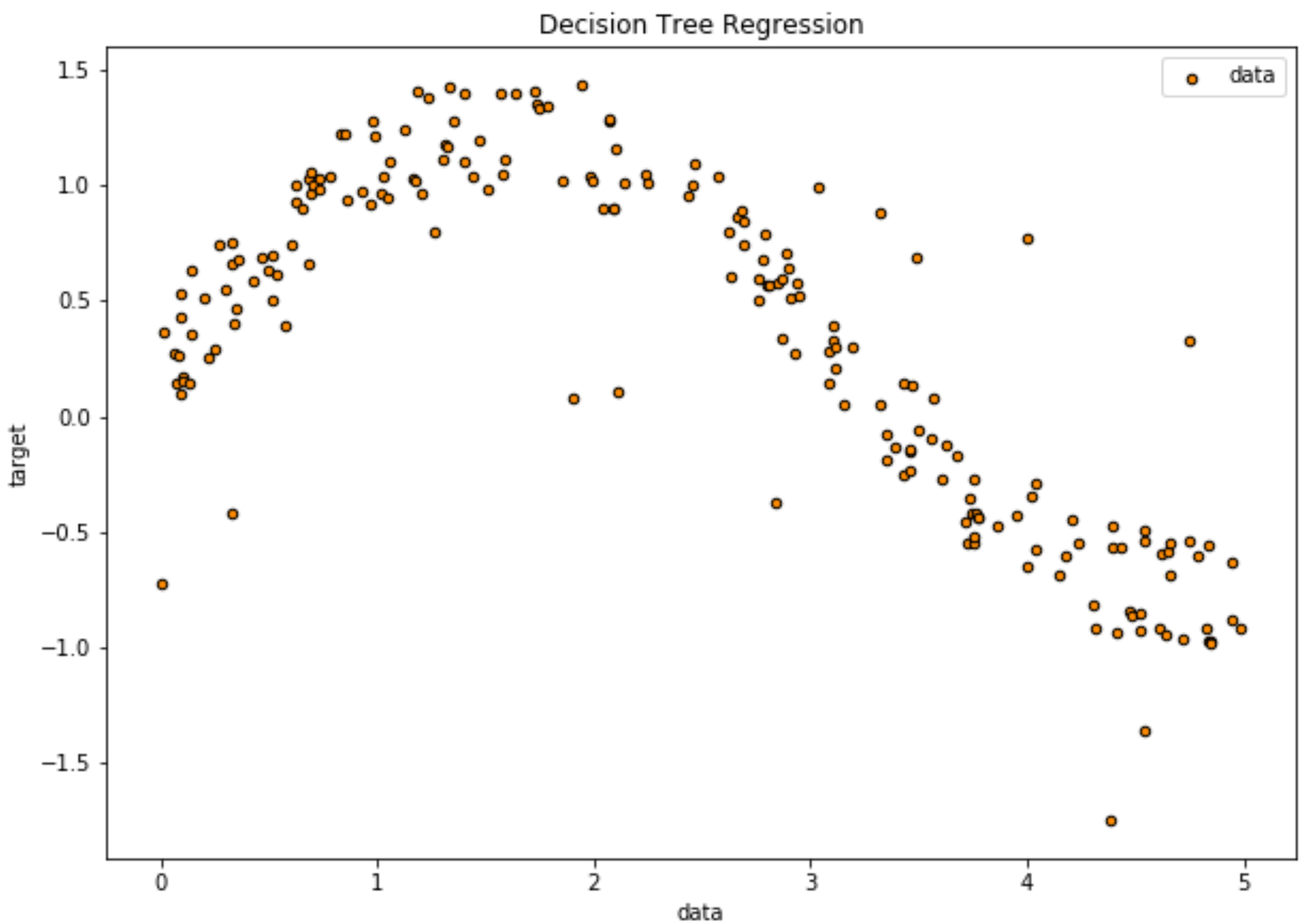
Why Ensemble is better?

Decision Tree의 한계점

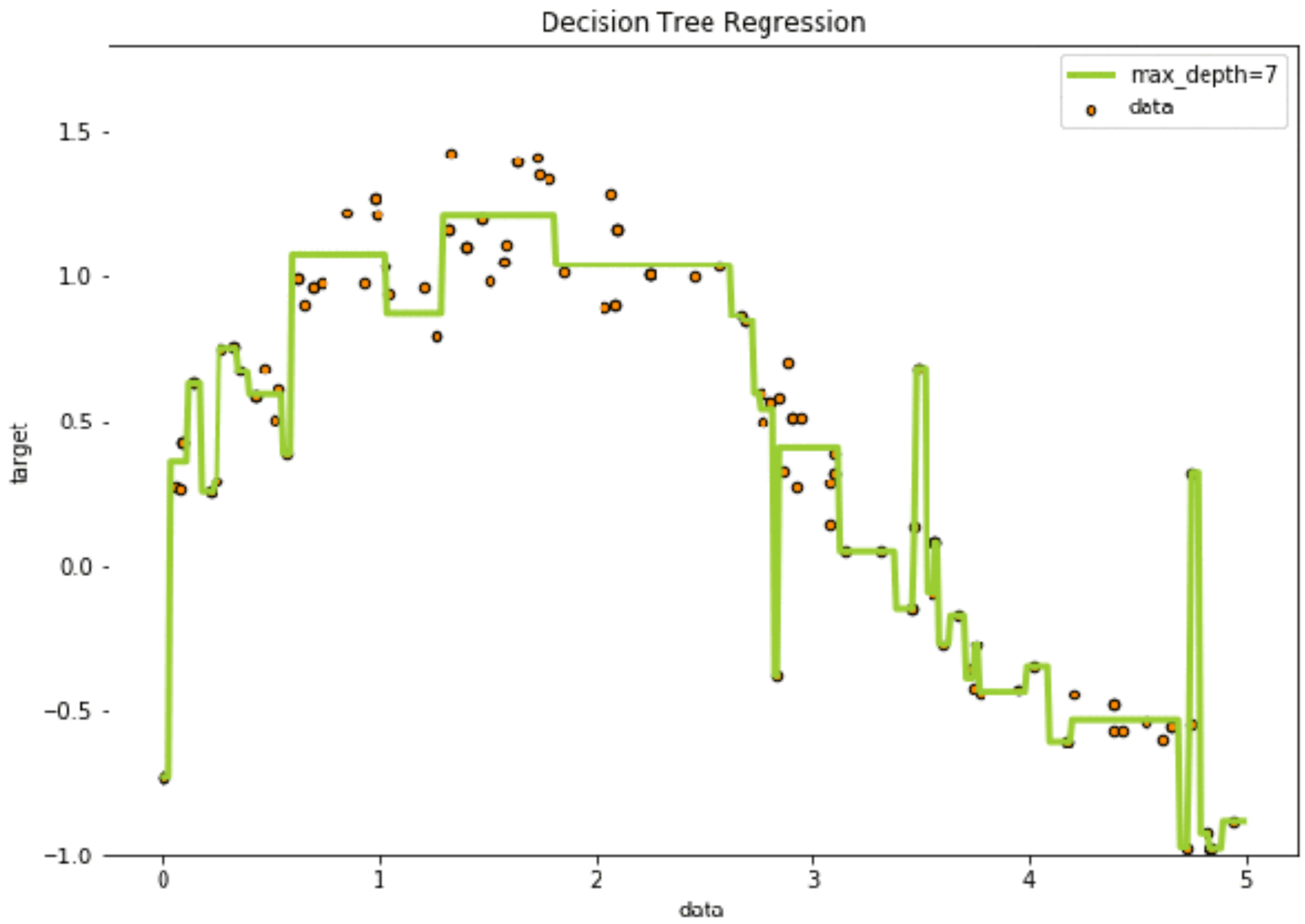
Decision Tree는 Variance가 큰 모델입니다
Input이 조금이라도 변하면 모델이 크게 변합니다



아웃라이어가 있는 위와 같은 2차원 데이터가 있다고 해봅시다

Decision Tree의 한계점

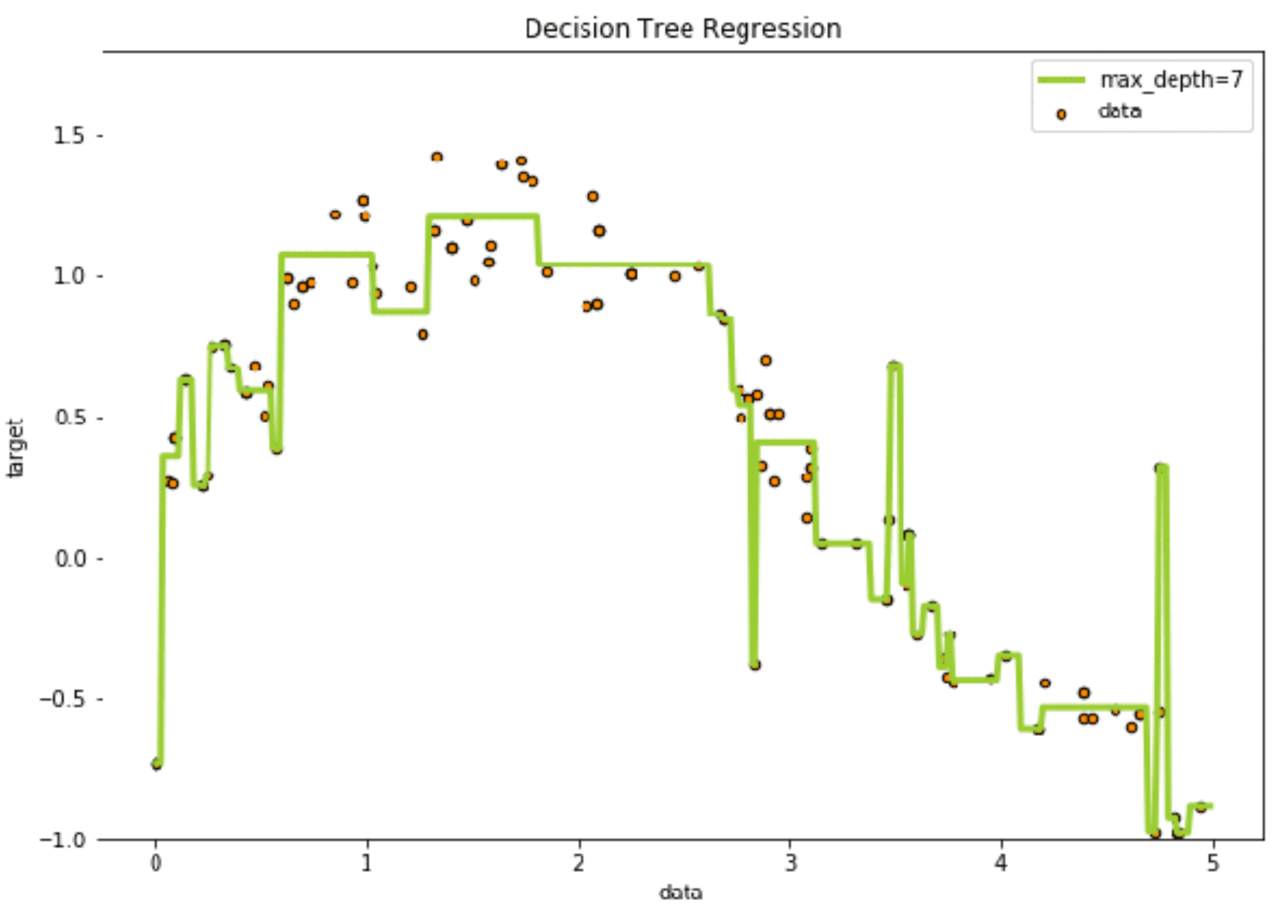
위의 원본데이터에서 70% 샘플링을 진행한 뒤 Decision Tree를 적합
그리고 각각의 x값에 대해 y값을 예측하면 다음과 같은 그래프가 나옵니다

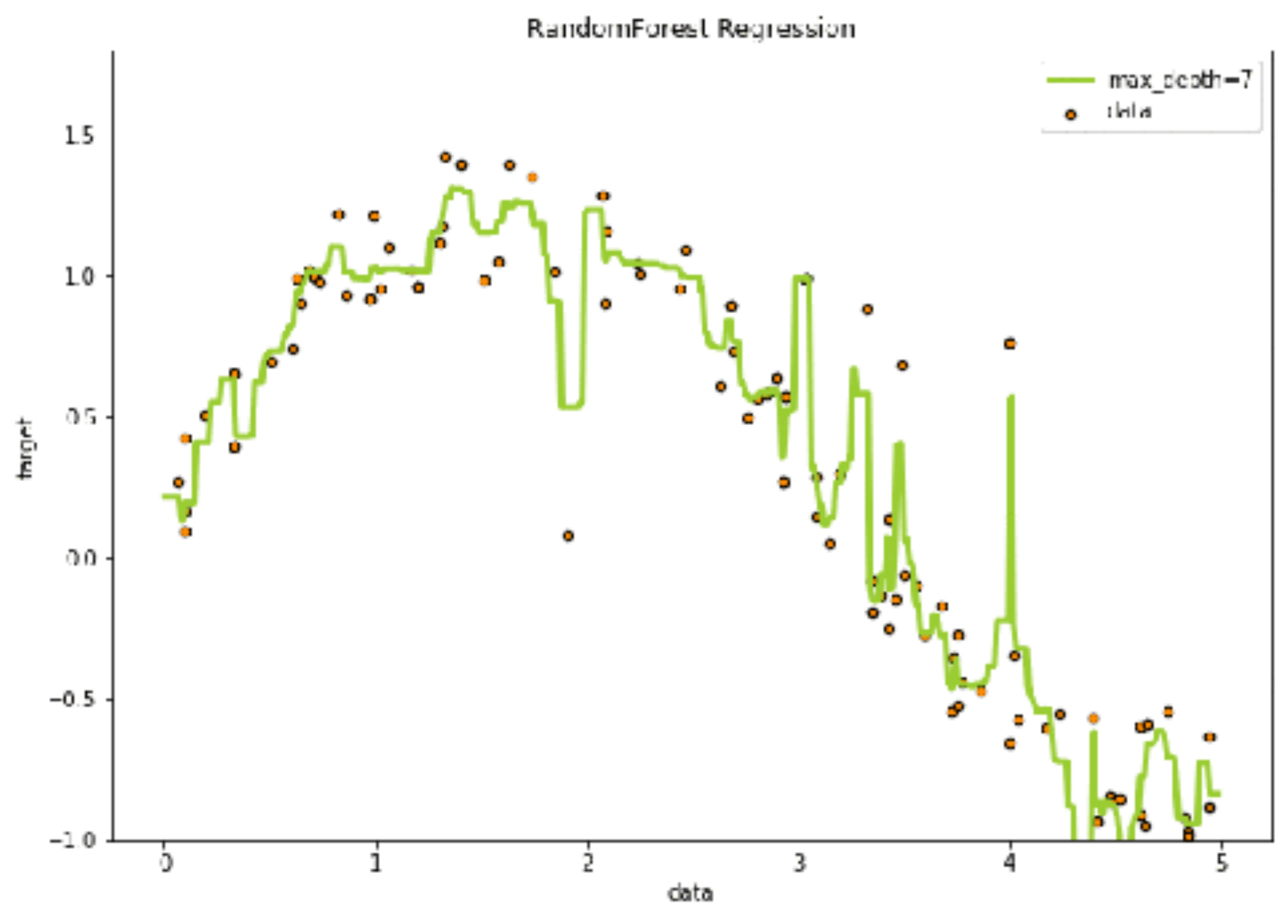


Input 데이터에 큰 영향을 받음
= 모델의 Variance가 큼

Random Forest가 이를 극복하는 방법

Decision Tree 여러개를 결합하여 Variance (변동성)을 낮춘다
tree처럼 outlier의 값을 그대로 따라가지 않고 모함수와 더 비슷해지는 것을 볼 수 있다



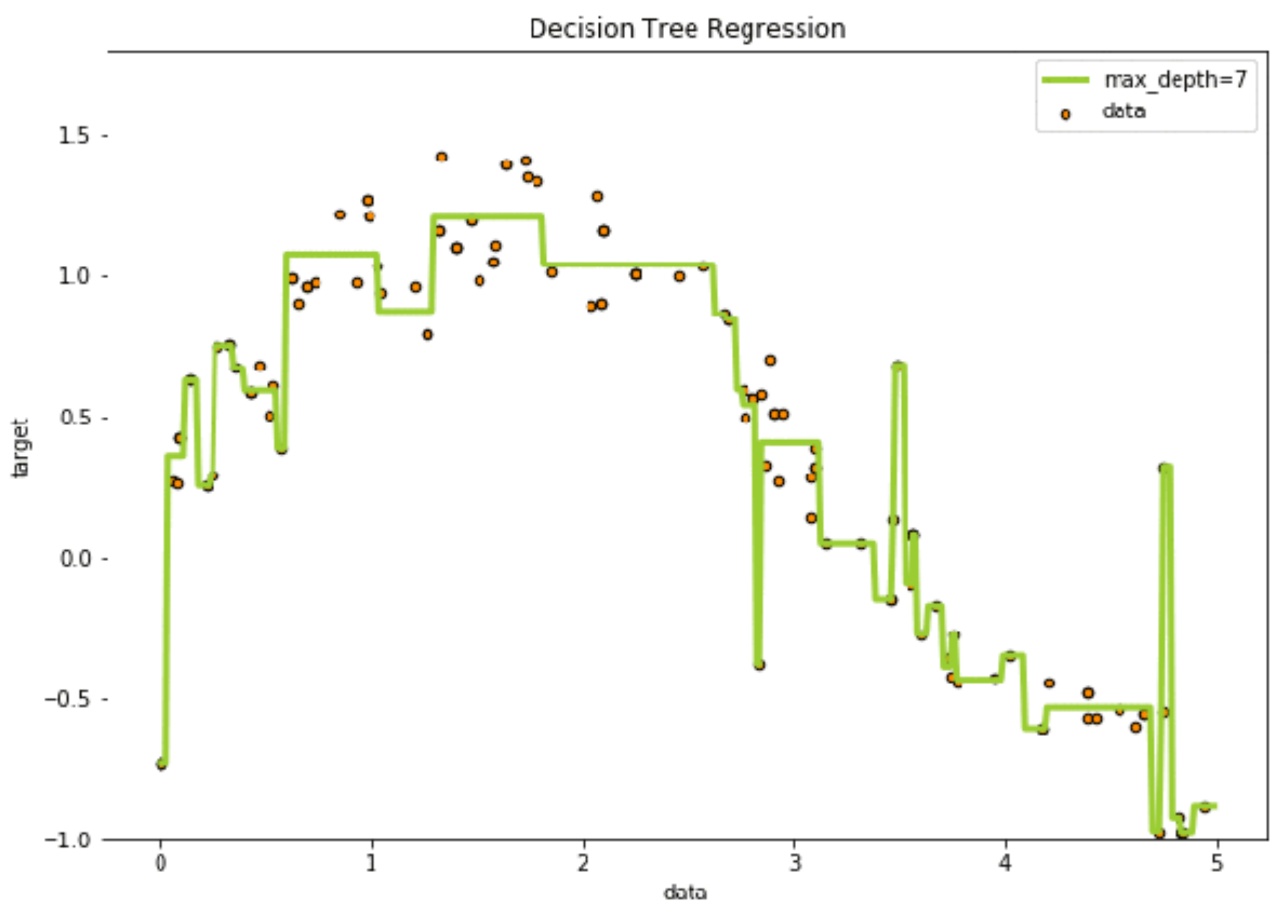


High Variance

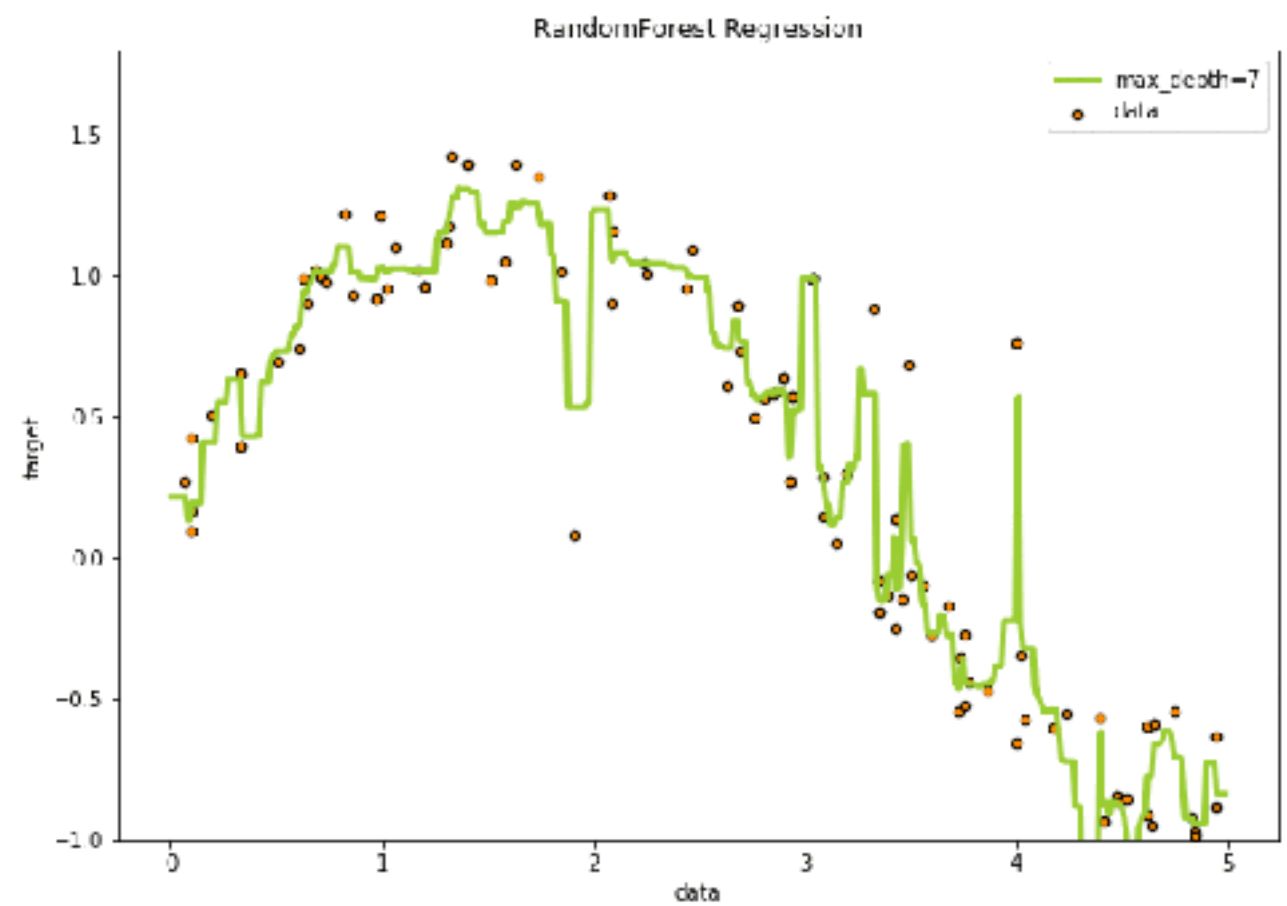
Low Variance

Random Forest가 가진 한계

Decision Tree 여러개를 결합하여 Variance (변동성)을 낮추지만
Decision Tree 자체가 가진 한계(=bias)인 선형 decision boundary를 극복하지 못한다



High Variance

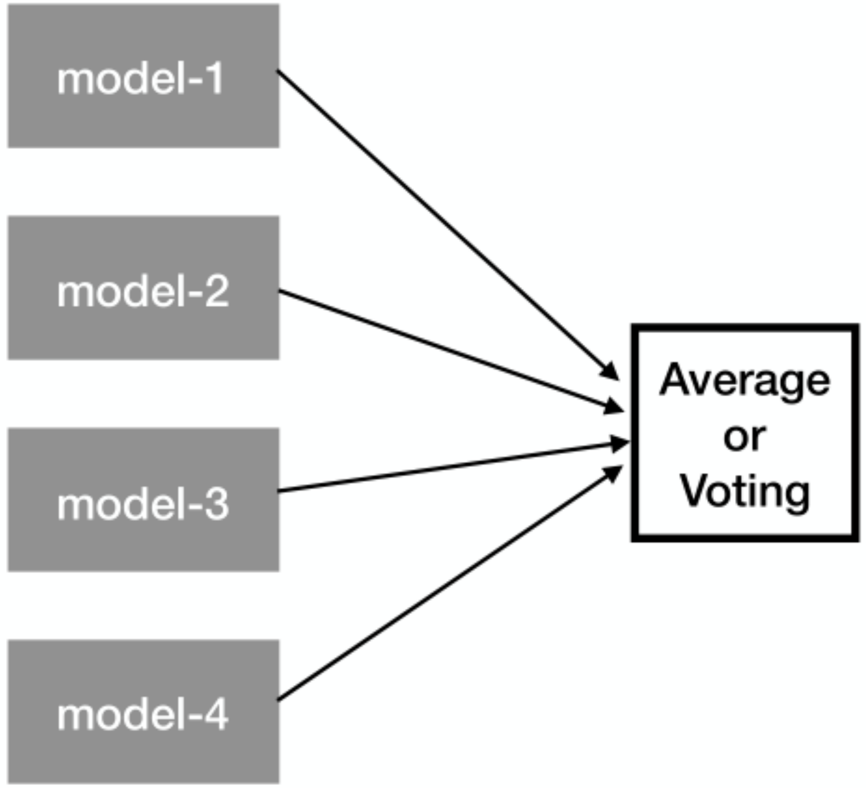


Low Variance

Boosting

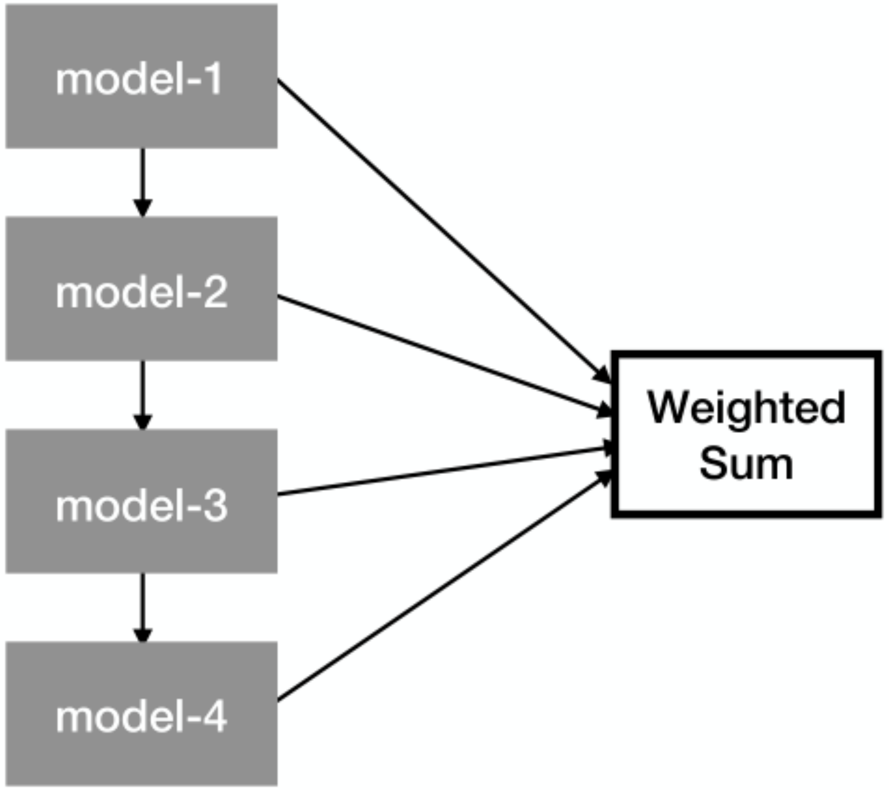
Decision Tree의 한계점

Bagging



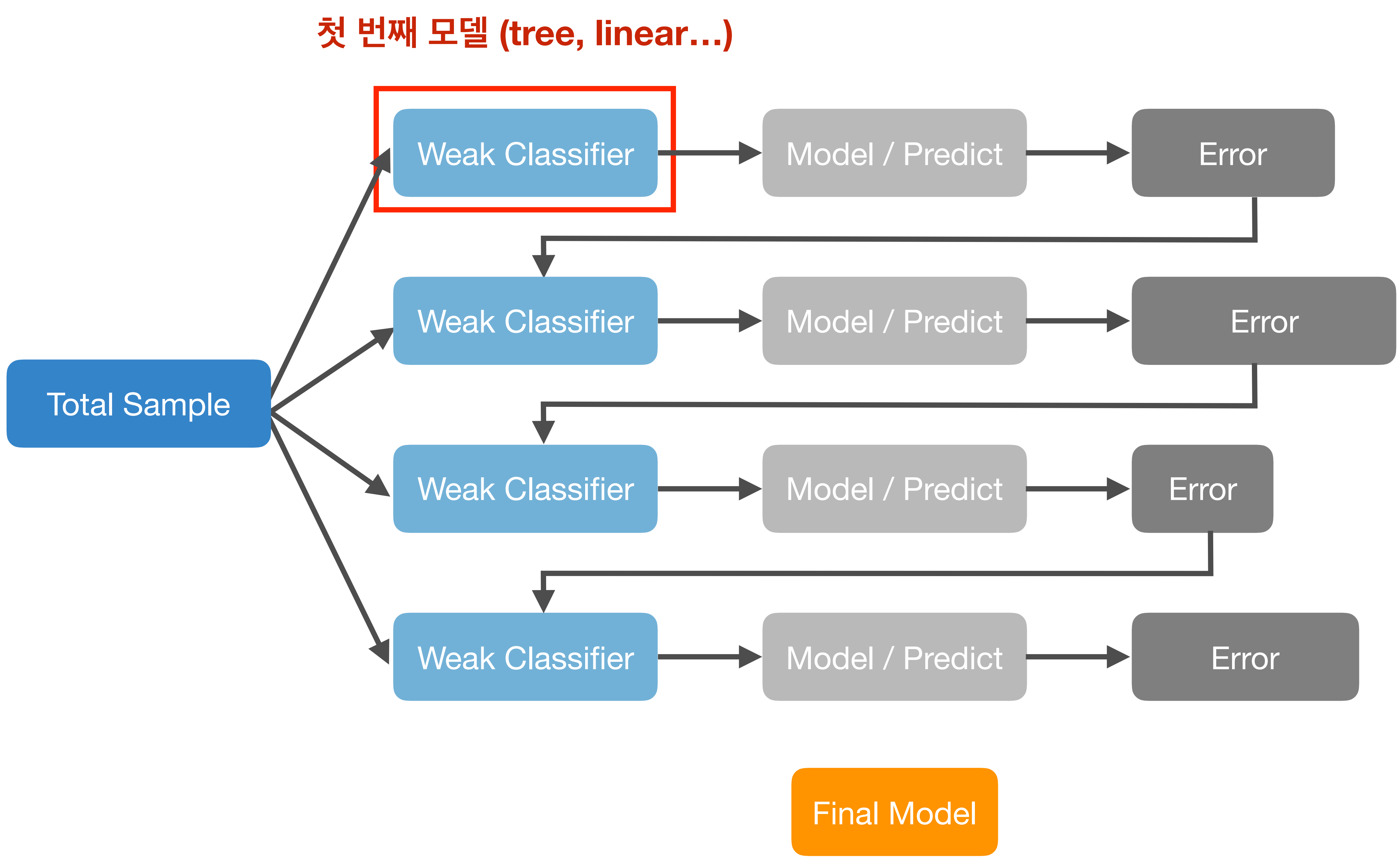
각 모델들은 병렬적으로 학습
모델끼리 영향을 주지 않음

Boosting

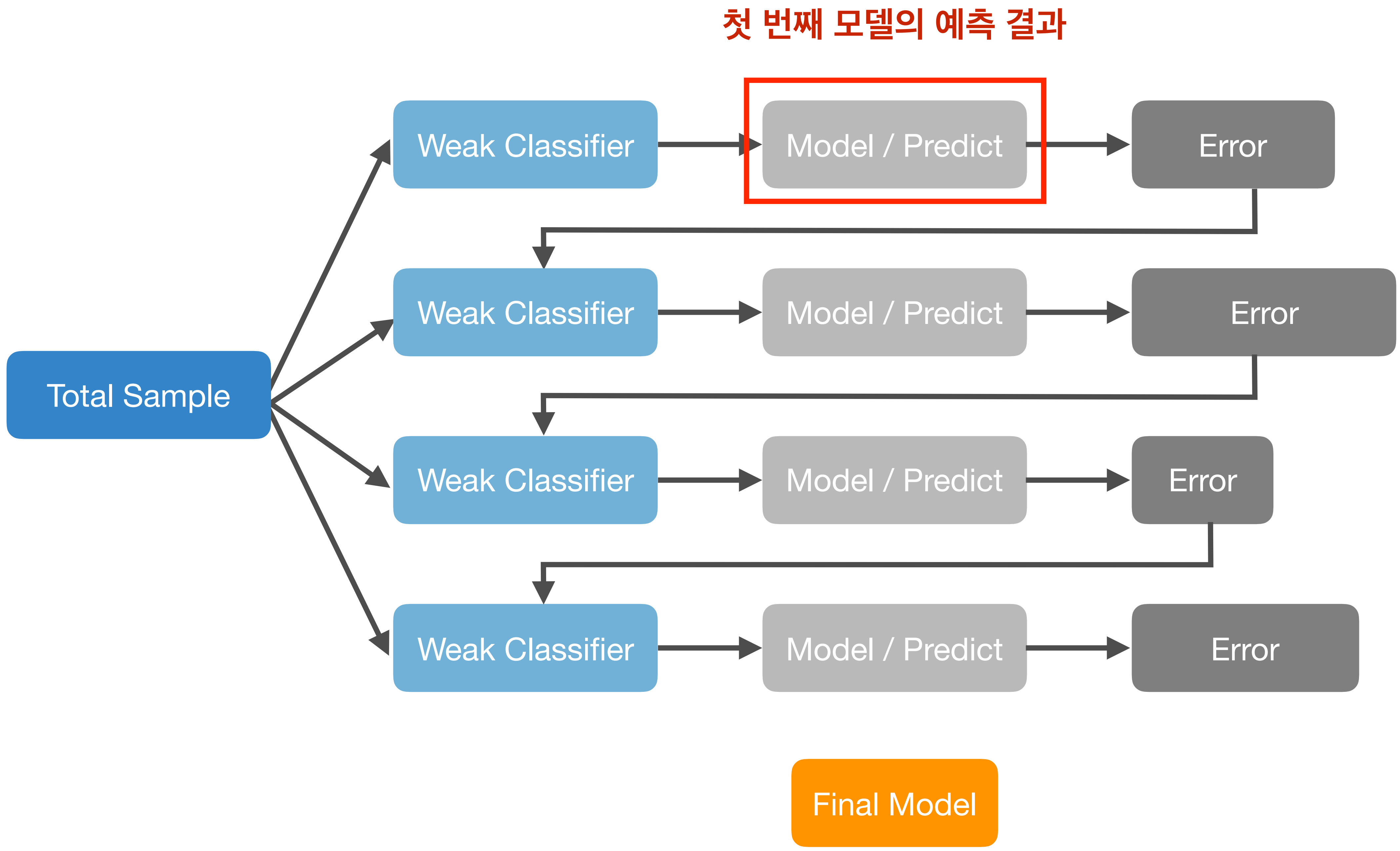


각 모델들은 순차적으로 학습
이전 모델의 학습 결과를 바탕으로 다음 모델을 학습

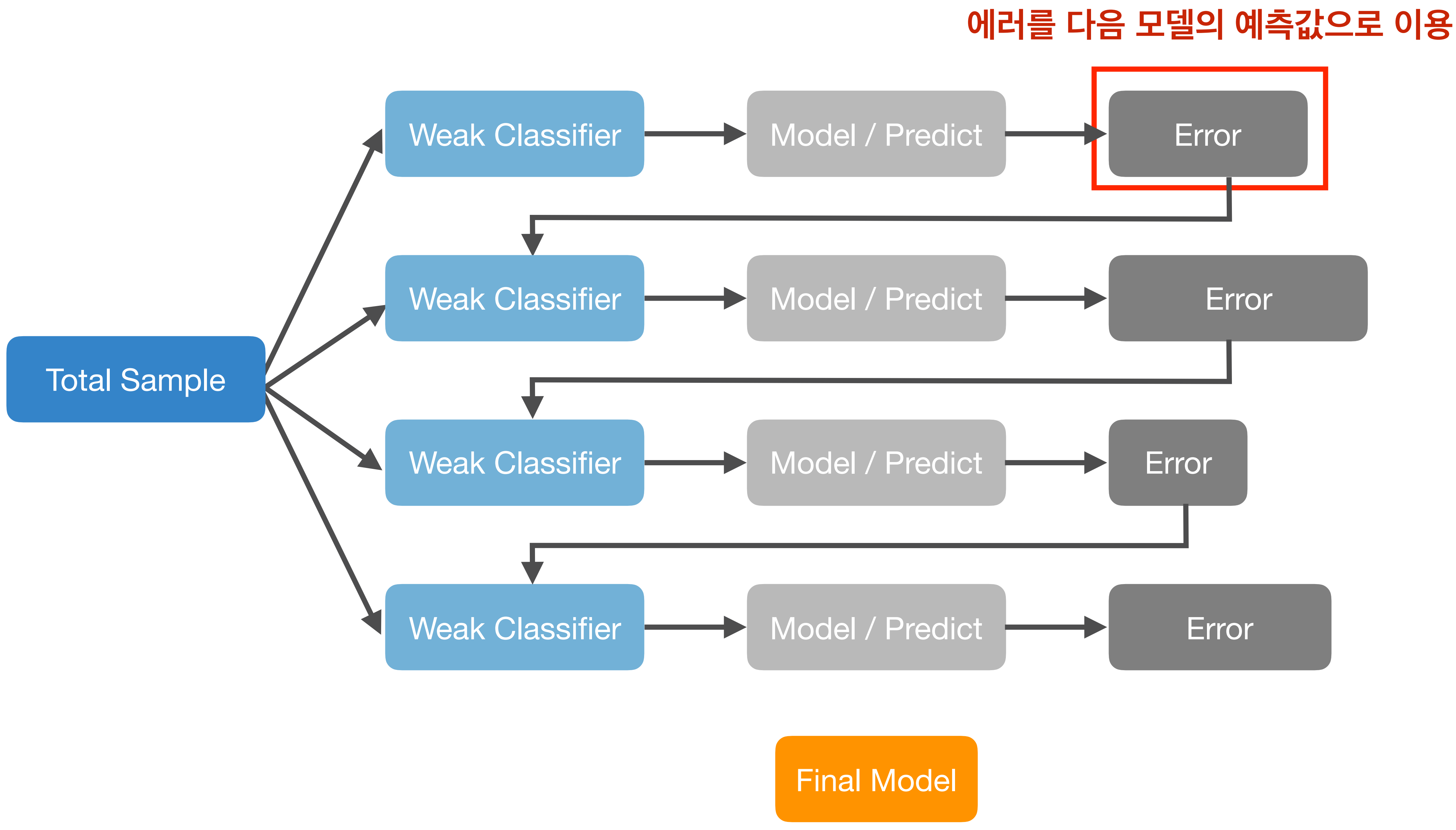
Gradient Boosting 작동 원리



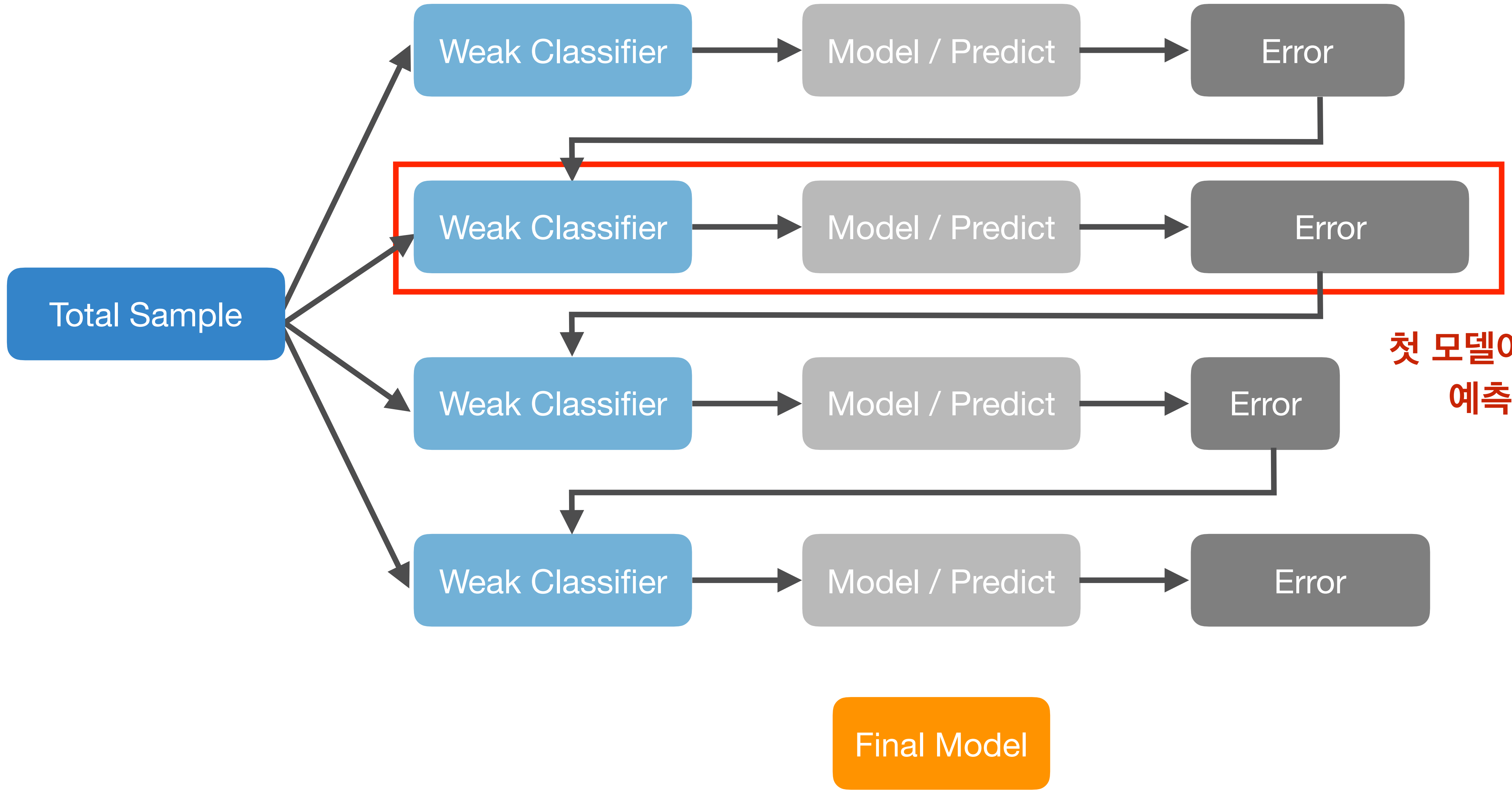
Gradient Boosting 작동 원리



Gradient Boosting 작동 원리

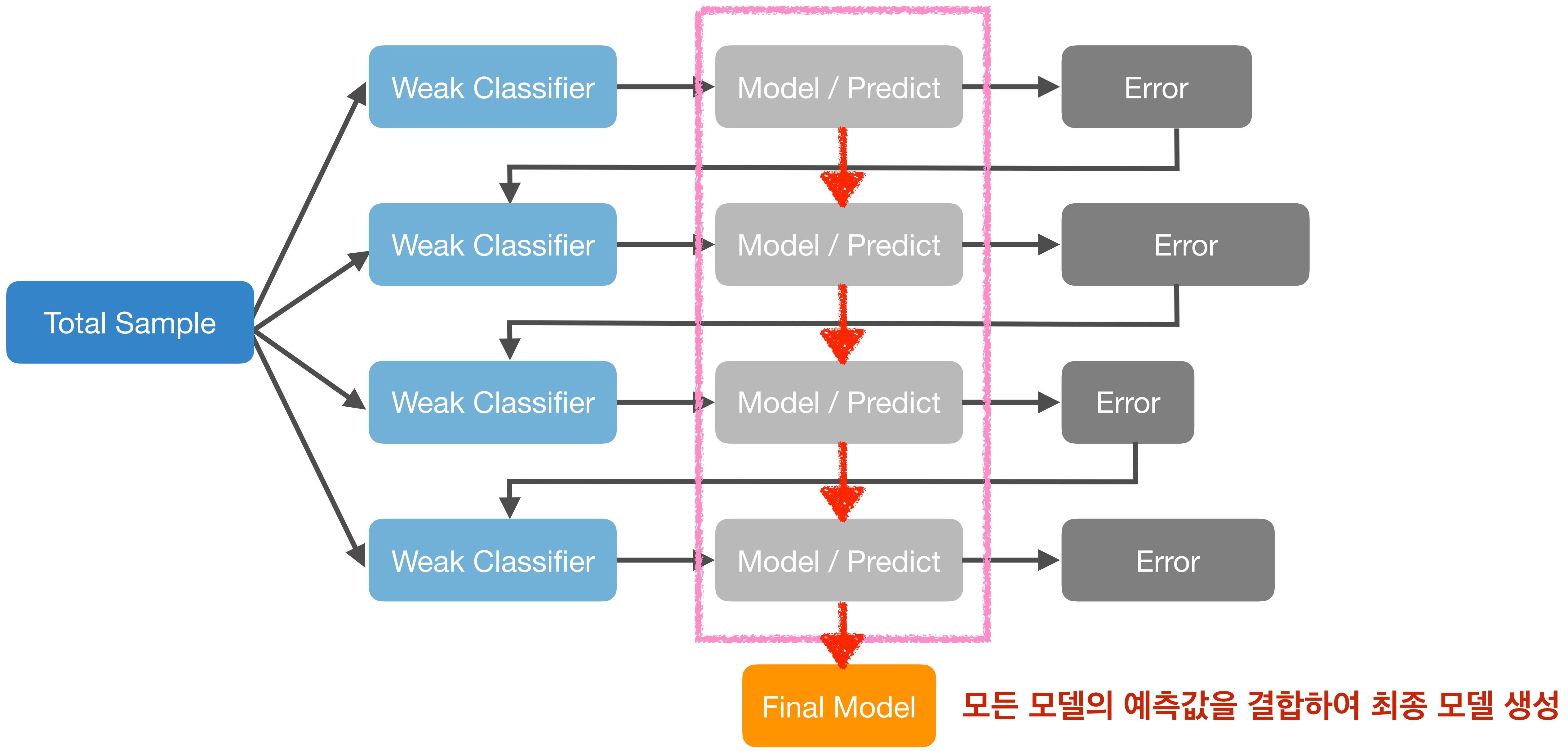


Gradient Boosting 작동 원리



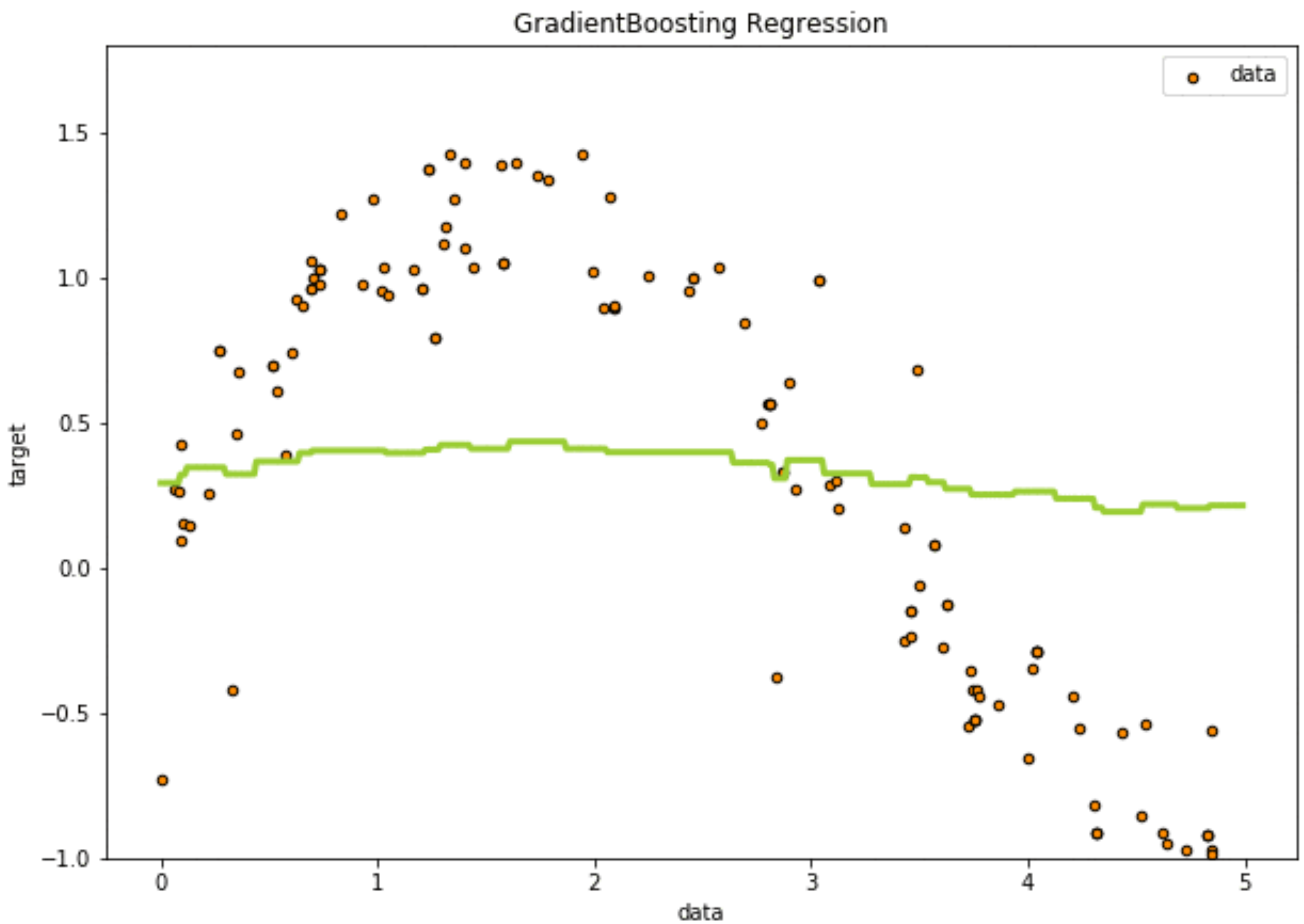
첫 모델이 맞추지 못한 부분을
예측하는 모델 만들기

Gradient Boosting 작동 원리



Boosting이 학습하는 과정

Weak model이 틀린점을 점차 개선해나가면서 여러개의 모델을 결합
Variance뿐만 아니라 Bias도 함께 줄일 수 있음



틀린 것을 어떻게 학습하는가?

기존 모델 학습 결과의 틀린 정도 (Loss Function에 의해 정의)를 학습

	y_true
0	1
1	2
2	3
3	4
4	5

틀린 것을 어떻게 학습하는가?

weak model에 의한 예측을 진행

	y_true	y_pred
0	1	3
1	2	3
2	3	3
3	4	3
4	5	3

틀린 것을 어떻게 학습하는가?

weak model이 틀린 정도를 다시 예측값으로 설정

	y_true	y_pred	error
0	1	3	-2
1	2	3	-1
2	3	3	0
3	4	3	1
4	5	3	2

→

	y_true
0	-2
1	-1
2	0
3	1
4	2

틀린 것을 어떻게 학습하는가?

첫 번째 모델의 error를 예측하는 두 번째 weak model 생성

	y_true	y_pred	error
0	1	3	-2
1	2	3	-1
2	3	3	0
3	4	3	1
4	5	3	2

→

	y_true	y_pred
0	-2	-1
1	-1	-1
2	0	0
3	1	1
4	2	1

틀린 것을 어떻게 학습하는가?

첫 번째 모델과 두 번째 모델의 예측 결과를 조합

	y_true	y_pred	error
0	1	3	-2
1	2	3	-1
2	3	3	0
3	4	3	1
4	5	3	2

	y_true	y_pred
0	-2	-1
1	-1	-1
2	0	0
3	1	1
4	2	1

	y_pred1	y_pred2
0	3	-1
1	3	-1
2	3	0
3	3	1
4	3	1

틀린 것을 어떻게 학습하는가?

두 예측 결과를 조합했을 때 첫 번째 모델의 예측 결과보다 성능이 개선되었음을 알 수 있다

	y_true	y_pred	error
0	1	3	-2
1	2	3	-1
2	3	3	0
3	4	3	1
4	5	3	2

	y_true	y_pred
0	-2	-1
1	-1	-1
2	0	0
3	1	1
4	2	1

	y_pred1	y_pred2	final_pred
0	3	-1	2
1	3	-1	2
2	3	+ 0	3
3	3	1	4
4	3	1	4

틀린 정도를 어떻게 정의하는가?

틀린 정도를 측정해주는 함수를 Loss Function이라고 합니다

y_i : **i 번째 데이터의 target 값**

x_i : **i 번째 데이터의 feature**

$f(x_i)$: **모델** f**가** x_i **를 입력 받았을 때의 예측값**

$$L(y, f(x)) = \frac{1}{2}(y - f(x))^2$$

Loss가 최소가 되는 지점을 어떻게 찾아나가는가?

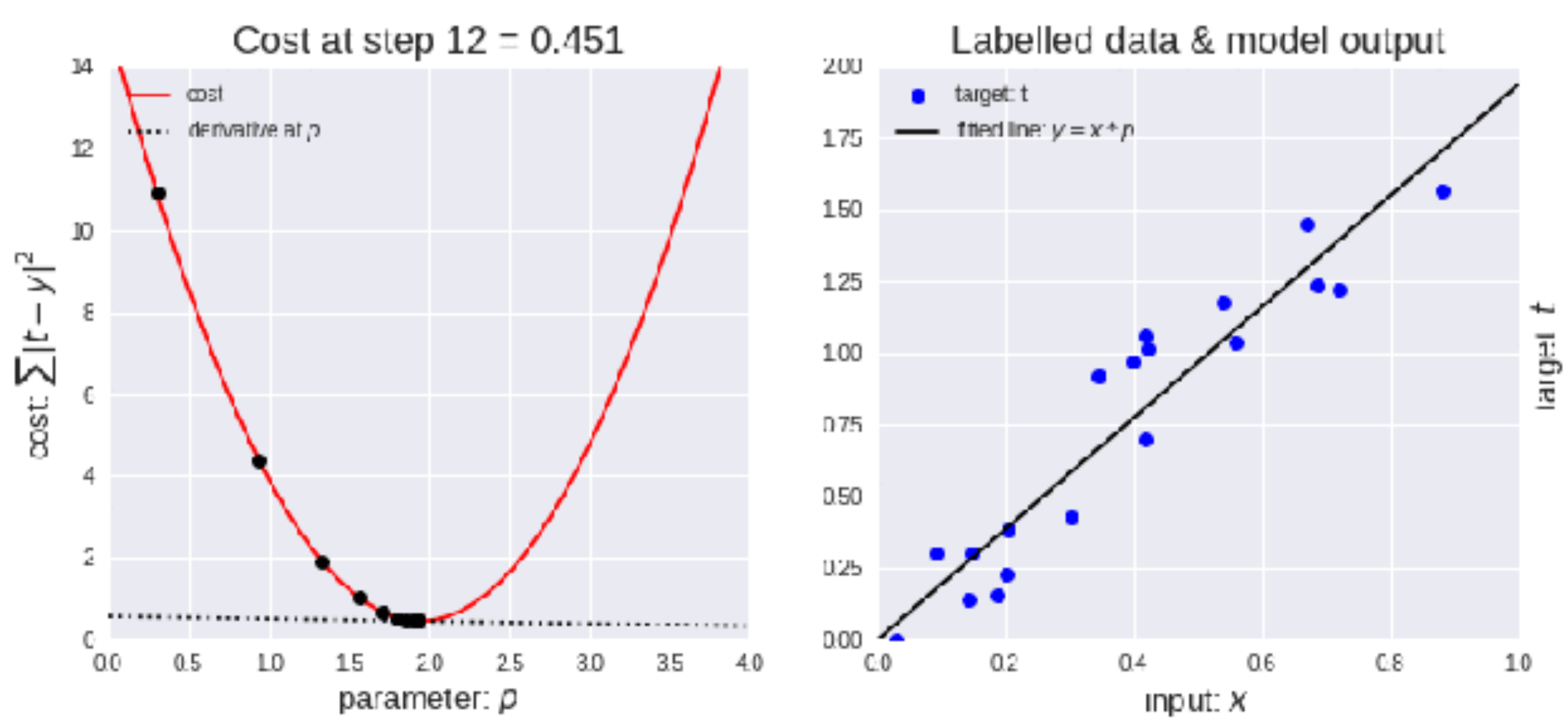
모델을 Loss Function의 기울기를 따라 수정해나가면 가장 빠르게 최소점을 찾을 수 있습니다

y_i : **i 번째 데이터의 target 값**

x_i : **i 번째 데이터의 feature**

$f(x_i)$: **모델 f가 x_i 를 입력 받았을 때의 예측값**

$$L(y, f(x)) = \frac{1}{2}(y - f(x))^2$$



Gradient Descent

Loss Function의 f(x)에 대한 기울기는 편미분을 통해 구할 수 있습니다

y_i : **i 번째 데이터의 target 값**

x_i : **i 번째 데이터의 feature**

$f(x_i)$: **모델** f**가** x_i **를 입력 받았을 때의 예측값**

$$\frac{\partial L(y, f(x))}{\partial f(x)} = \frac{\partial}{\partial f(x)} \frac{1}{2}(y - f(x))^2$$

$$L(y, f(x)) = \frac{1}{2}(y - f(x))^2$$

Gradient Descent

y를 상수취급하고 미분하면 다음과 같은 식을 얻게 됩니다

y_i : **i 번째 데이터의 target 값**

x_i : **i 번째 데이터의 feature**

$f(x_i)$: **모델** f**가** x_i **를 입력 받았을 때의 예측값**

$$\frac{\partial L(y, f(x))}{\partial f(x)} = \frac{\partial}{\partial f(x)} \frac{1}{2}(y - f(x))^2$$

$$\frac{\partial L(y, f(x))}{\partial f(x)} = f(x) - y$$

$$L(y, f(x)) = \frac{1}{2}(y - f(x))^2$$

Gradient Descent

따라서 Loss가 L2 (MSE)인 경우 error를 negative gradient라고 할 수 있습니다

y_i : **i 번째 데이터의 target 값**

x_i : **i 번째 데이터의 feature**

$f(x_i)$: **모델** f**가** x_i **를 입력 받았을 때의 예측값**

$$L(y, f(x)) = \frac{1}{2}(y - f(x))^2$$

$$\frac{\partial L(y, f(x))}{\partial f(x)} = \frac{\partial}{\partial f(x)} \frac{1}{2}(y - f(x))^2$$

$$\frac{\partial L(y, f(x))}{\partial f(x)} = f(x) - y$$

$$y_i - f(x_i) = -\frac{\partial L(y_i, f(x_i))}{\partial f(x_i)}$$

Gradient Descent

따라서 Loss가 L2 (MSE)인 경우 error를 negative gradient라고 할 수 있습니다

y_i : **i 번째 데이터의 target 값**

x_i : **i 번째 데이터의 feature**

$f(x_i)$: **모델 f가 x_i 를 입력 받았을 때의 예측값**

$$L(y, f(x)) = \frac{1}{2}(y - f(x))^2$$

$$\frac{\partial L(y, f(x))}{\partial f(x)} = \frac{\partial}{\partial f(x)} \frac{1}{2}(y - f(x))^2$$

$$\frac{\partial L(y, f(x))}{\partial f(x)} = f(x) - y$$

$$y_i - f(x_i) = -\frac{\partial L(y_i, f(x_i))}{\partial f(x_i)}$$

L2-loss 뿐만 아니라 미분가능한 모든 loss function에 대해
Gradient Boosting을 적용할 수 있습니다