가천대 회화·조소과 AI 특강

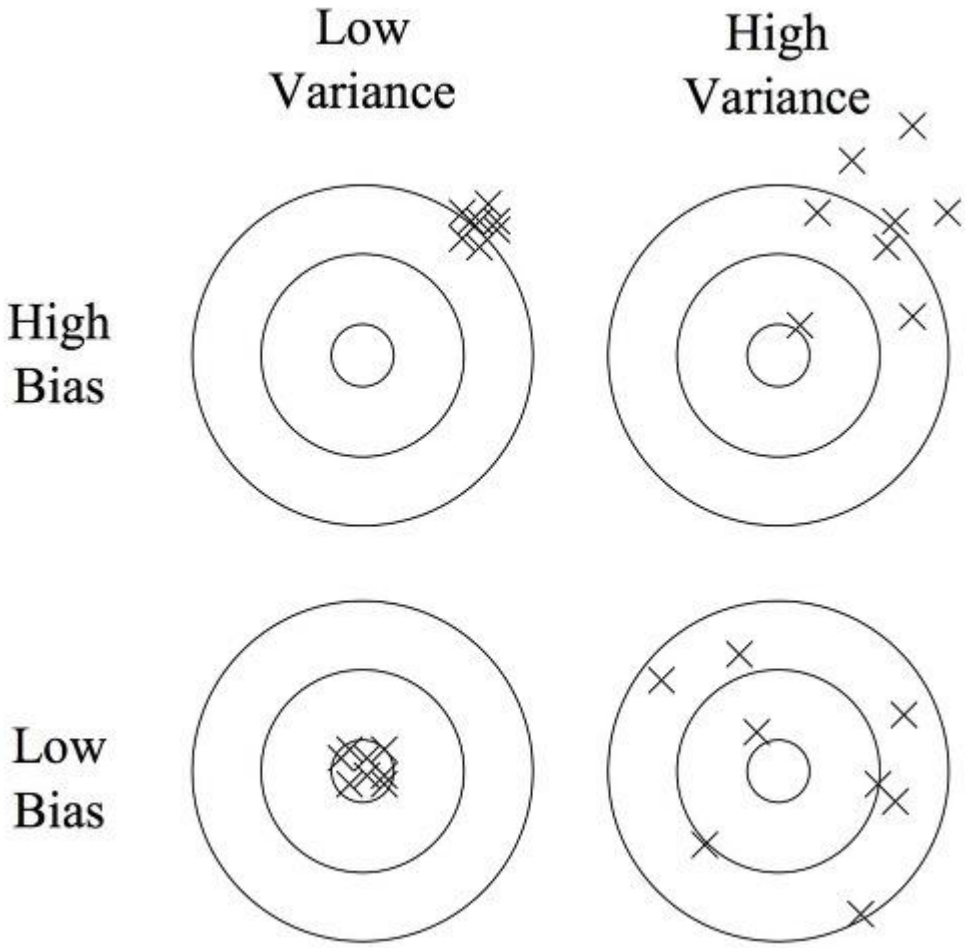
2021-06-19

조형래

Bias-Variance tradeoff

Overfitting과 Underfitting 정의 및 해결 방법

Bias-Variance



바이어스 :
" 훈련 데이터"와 관련된 전반적인 오류를 편향이라고 함
높은 편향: 훈련 데이터 오류가 증가하거나 훈련 데이터 정확도가 감소

낮은 편향: 훈련 데이터 오류가 감소하거나 훈련 데이터 정확도가 증가

분산 :
테스트 데이터와 관련된 전체 오류

높은 분산 : 높은 테스트 데이터 오류 / 낮은 테스트 데이터 정확도.
낮은 분산 : 낮은 테스트 데이터 오류 / 높은 테스트 데이터 정확도.

Bias-Variance

Error(X) = noise(X) + bias(X) + variance(X)

(noise ; 데이터가 가지는 본질적인 한계치, 변하지 않는 irreducible error

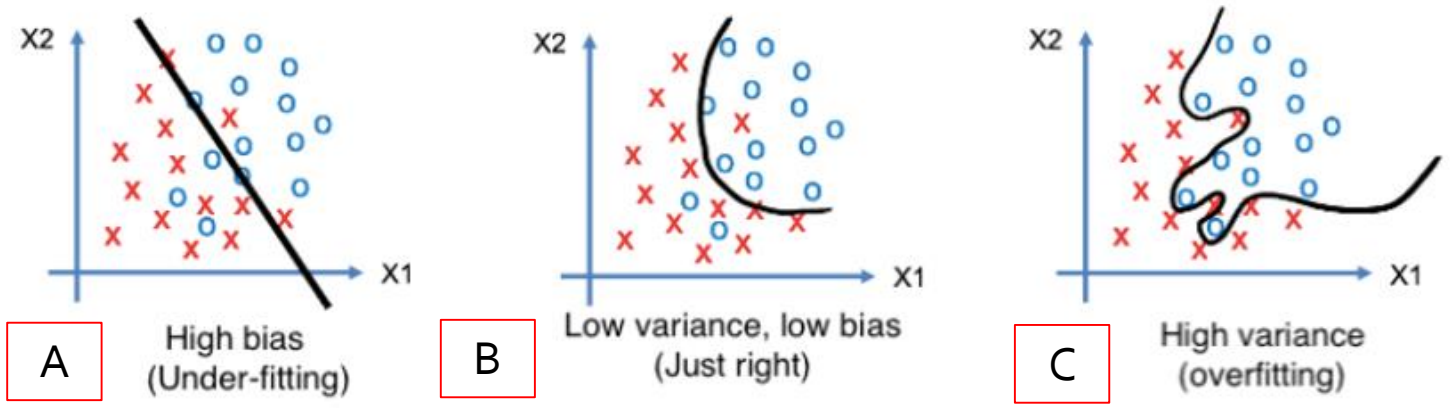
bias/variance; 모델에 따라 변하는 reducible error)

Bias는 -데이터 내에 있는 모든 정보를 고려하지 않고
-지속적으로 잘못된 것들을 학습하는 경향이 있다. Underfitting
-트레이닝 데이터를 바꿈에 따라서 알고리즘의 평균 정확도가 얼마나 많이 변하는지를 보여준다.

Variance는 -데이터 내에 있는 에러나 노이즈까지 잘 잡아내는 highly flexible models 에 데이터를
-fitting시킴으로써, 실제 현상과 관계 없는 random한 것들까지 학습하는 알고리즘, overfitting
-특정 입력 데이터에 대해 알고리즘이 얼마나 민감한 지를 나타낸다.

이상적인 모델은 트레이닝 데이터에서 반복되는 규칙성을 정확하게 잡아내면서도 학습되지 않은 (unseen) 데이터를 잘 일반화 할 수 있는 모델이다.

Underfitting 과 Overfitting



A: 선형 모델은 under-fit

1) 데이터 내의 모든 정보를 고려하지 못한다 (high bias)
2) 새로운 데이터가 들어온다 하더라도 이 모델의 형태는 크게 변하지 않을 것이다 (low variance)

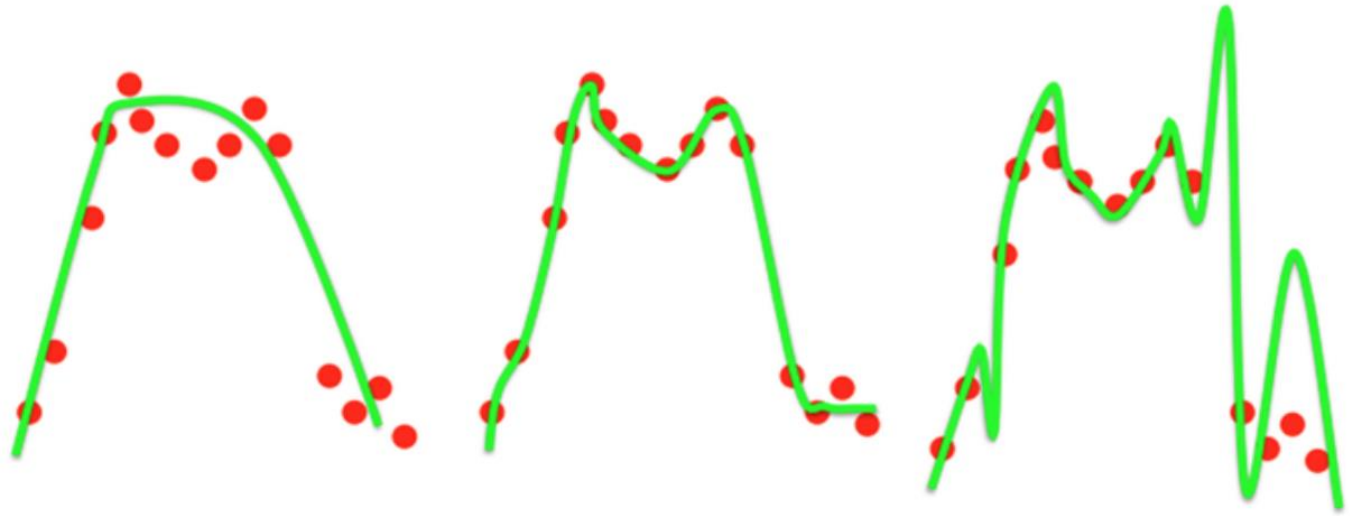
C: 고차 다항함수 모델은 over-fit

1) 모델은 주어진 데이터를 잘 설명하고 있다 (low bias)
2) 새로운 데이터가 들어왔을 때 완전히 다른 형태로 변하게 되고, generality를 잃게 된다(high variance)

B: 이상적인 모델은 데이터의 규칙성을 잘 잡아내어 정확하면서도 다른 데이터가 들어왔을 때도 잘 일반화할 수 있는 모델

이러한 문제를 bias-variance trade-off 라고 함

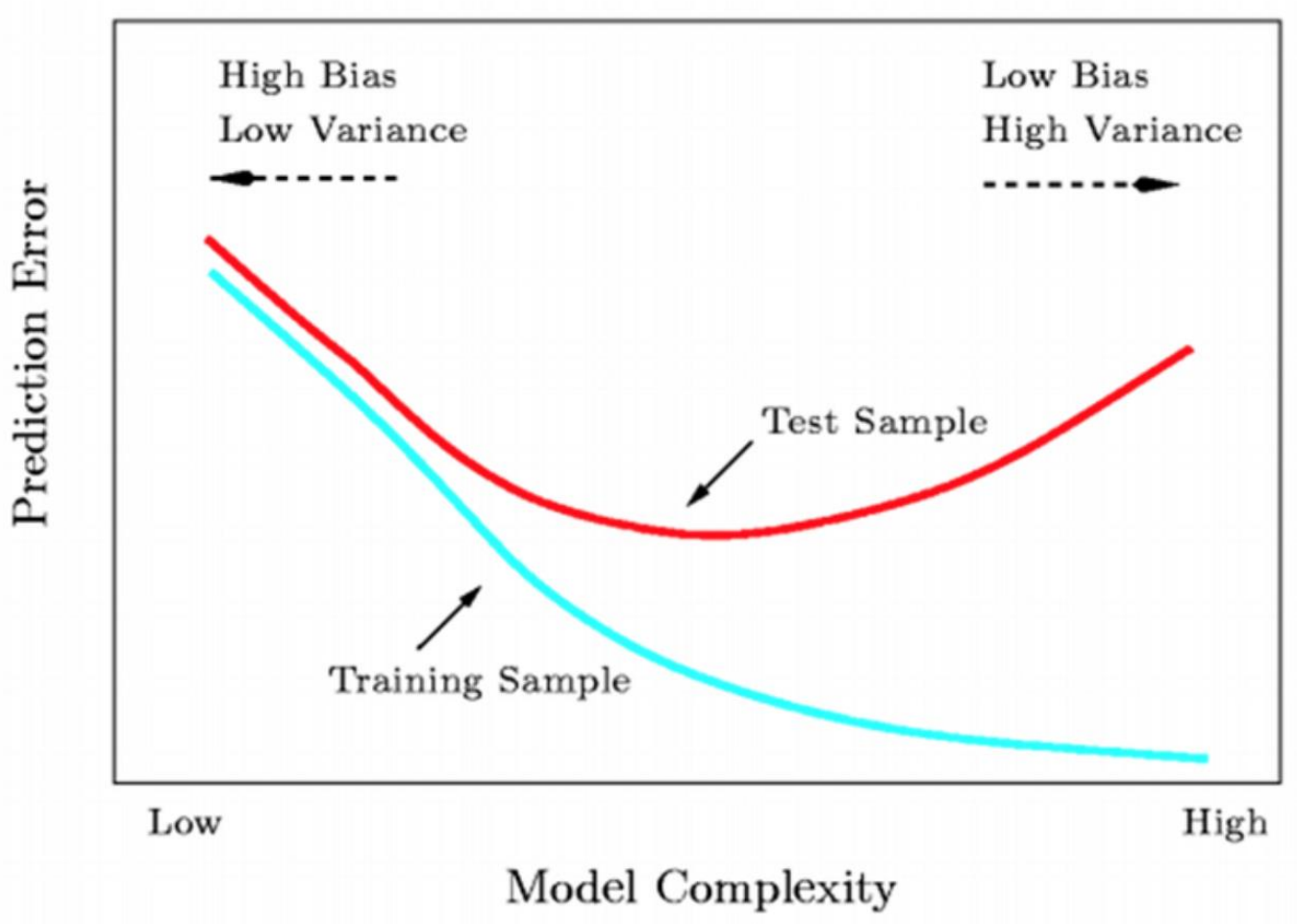
Overfitting과 Underfitting 정의 및 해결 방법



주어진 점들의 함수(곡선)를 추정한다고 할 때, 즉 optimize한다면
• 왼쪽은 지나친 단순화로 인해 에러가 많이 발생, **underfitting**
• 오른쪽은 너무 정확하게 표현한 나머지 training data에 대한 정확도는 좋지만 실제 test에서는 에러가 날 수 있는 상황, **overfitting**

- 모델은 과대적합(Overfitting)과 과소적합(Underfitting)이 발생하지 않도록 설계하는 것이 중요

Overfitting



Overfitting은 학습 데이터(Training Set)에 대해 **과하게** 학습된 상황이다.
따라서 학습 데이터 이외의 데이터에 대해선 모델이 잘 동작하지 못하는 경우로

학습 데이터가 부족하거나, 데이터의 특성에 비해 모델이 너무 복잡한 경우 발생한다.

Training Set에 대한 loss는 계속 떨어지는데, Test Set에 대한 loss는 감소하다가 다시 증가하는 경우임

Overfitting

원인: Model Capacity의 문제
Model Capacity는 모델이 복잡한 형상을 나타낼 수 있는 정도를 의미
Model Capacity를 늘리려면 layer를 더 deep하게 쌓거나 layer당 hidden unit 개수를 늘리면 된다.
하지만 model capacit를 무한정 늘리면 overfitting이 발생한다.

overfitting을 방지하려면 **학습 데이터에 적합한 Model Capacity**를 가지도록 모델을 설계할 필요가 있다.
예를 들어 dataset이 Train set만 있다면, 이 데이터로 Model Capacity가 매우 높은 모델을 학습하면, **모델이 Train set을 외워버리는 것**으로 학습하게 됨.
모델이 Train set을 외웠기 때문에, test시 다른 데이터가 들어오면 올바른 결과를 주지 못하게 된다.

이런 문제를 방지하기 위해 train에 사용하지 않는 test set을 두고, test set에 대해서도 모델이 잘 동작하는지 확인한다. Train sample의 loss는 계속 감소하는데, Test sample의 loss만 계속 증가한다면 overfitting이 발생한다고 볼 수 있다.

Overfitting

해결책
-Model Capacity 낮추기: 모델이 학습 데이터에 비해 과하게 복잡하지 않도록, hidden layer 크기를 줄이거나 layer 개수를 줄이는 등 모델을 간단하게 만든다.
-Dropout: 학습을 할 때 일부 뉴런을 끄고 학습
-L1/L2 정규화(L1/L2 regularization)
-학습 데이터 늘리기(data augmentation)

Test Set Accuracy가 증가하다가 감소하면 학습 데이터가 부족한 경우로 볼 수 있다.
학습 데이터를 늘릴 필요가 있는데 이미지 같은 경우 이미지의 비율을 바꾸거나, 일부분을 가리거나, 회전하는 것으로 데이터를 늘릴 수 있다. 이것을 data augmention이라고 함

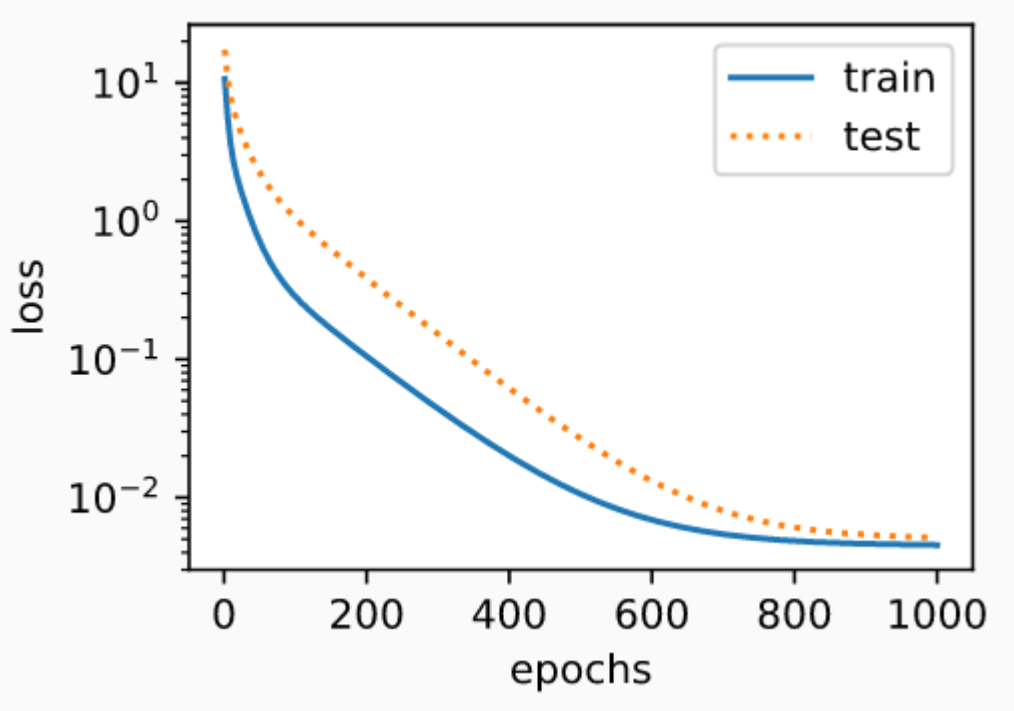
Training Set Accuracy가 100%에 가깝지만 Test Set Accuracy가 상당히 낮은 경우는
학습 데이터가 편향되어 있지 않은지 확인할 필요가 있다. 특수한 경우의 데이터를 가지고 일반적 문제를 해결하는 경우가 해당될 수 있다.

Underfitting

Underfitting(과소적합)은 이미 있는 Train set도 학습을 하지 못한 상태를 의미
Overfitting과 반대되는 상태를 의미합니다.

Overfitting이 발생하는 이유:
- 학습 반복 횟수가 너무 적음
- 데이터의 특성에 비해 모델이 너무 간단함
- 데이터 양이 너무 적음

선형 함수의 경우를 보면
초기 에포크(epoch)를 수행하면서 학습 오류가 감소한 후로 더 이상 모델 학습의 오류가 감소하지 않는 경우로 마지막 epoch까지 마친 후에도 학습 오류는 여전히 높다.



모델의 복잡도

$$\hat{y} = \sum_{i=0}^{d} x^i w_i$$

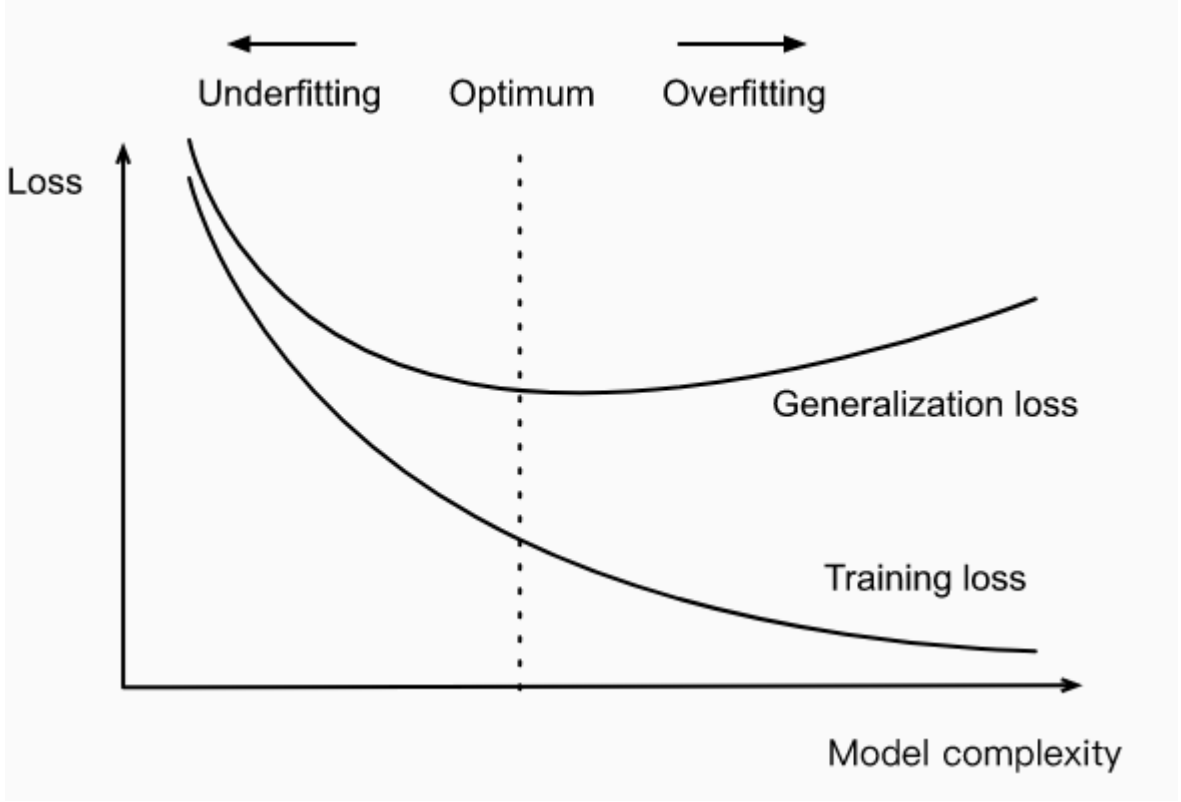
위 다항식의 경우
스칼라 데이터 특성(feature) x 와
이에 대한 스칼라 레이블(label) y 로
구성된 학습 데이터가 주어진 경우,

Y 를 추정하는 d 차원 다항식을 찾는다

w_i ; 모델의 가중치 파라미터
편향(bias)은 $x^0 = 1$ 이기 때문에 wo 가 된다.

d=1 인 경우로 선형 회귀(linear regression) 모델을 뜻함

모델의 복잡도



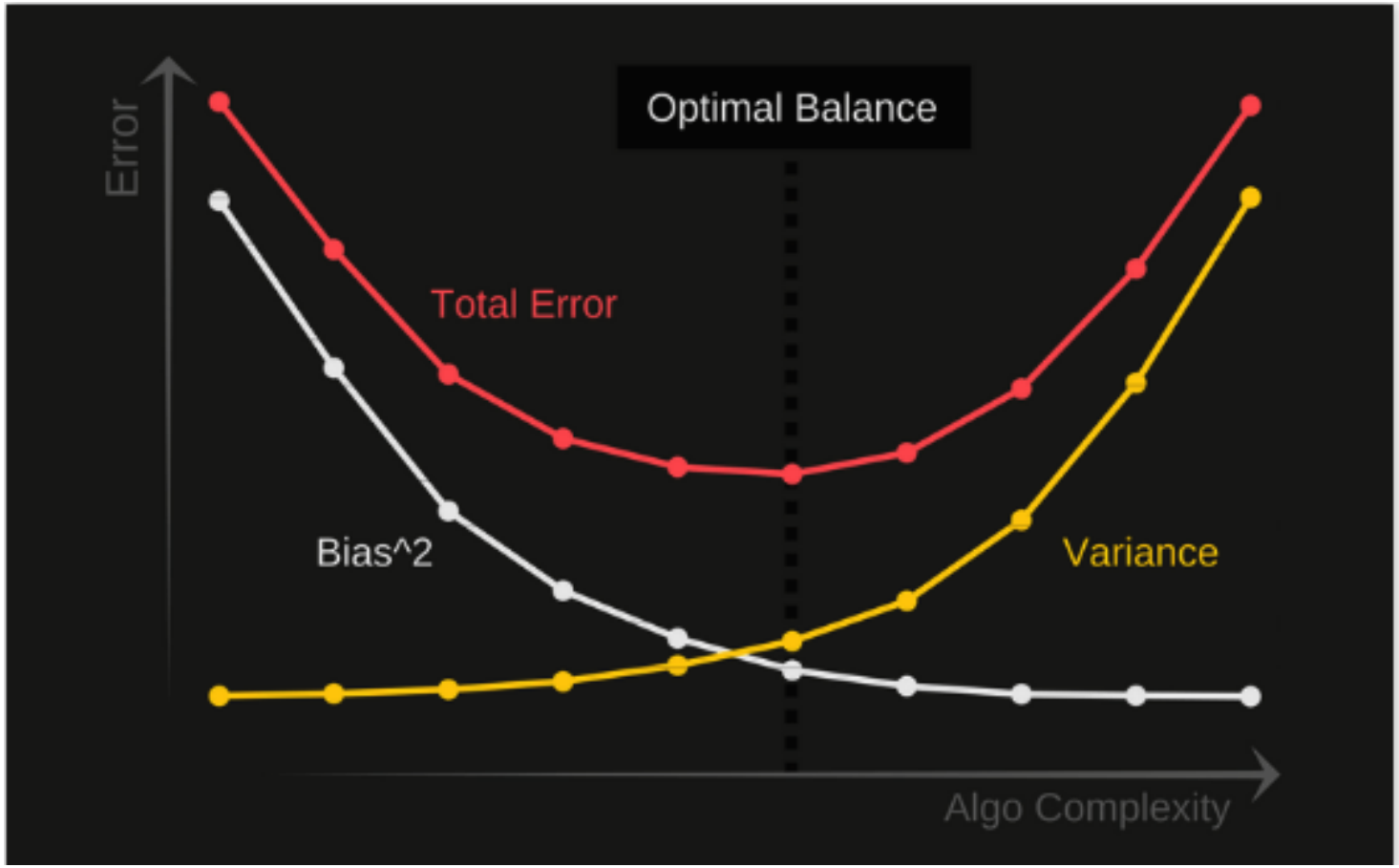
데이터에 비해서 모델이 너무 간단하면,
언더피팅(underfitting)이 발생하고,

모델을 너무 복잡하게 선택하면
오버피팅(overfitting)이 발생한다.

데이터에 대한 모델을 적절한 복잡성을
선택하는 것이 오버피팅(overfitting)과
언더피팅(underfitting) 문제를 피하는 방법
중에 하나이다.

Trade-off

분산이 낮을수록 편향이 높아지고(underfitting), 분산이 높을수록 편향이 낮아진다(overfitting)
빨간색 선이 Total Error, 즉 얼만큼 에러가 발생했냐를 나타내는 것인데 모델이 복잡도(x축)가 높아질수록 Total Error는 증가하는 경향을 보인다(overfitting). 따라서 에러를 가장 낮게 하는(편향과 분산이 적절하게 낮은 지점이 교차되는) 곳까지 학습을 시켜야 하는데 이 지점이 편향과 분산의 Trade-off이다.



$$\mathrm{Bias}\left[\hat{f}\left(x\right)\right] = \mathrm{E}\left[\hat{f}\left(x\right) - f(x)\right]$$

$$\mathrm{Var}\left[\hat{f}\left(x\right)\right] = \mathrm{E}\left[(\hat{f}\left(x\right) - \mathrm{E}[\hat{f}\left(x\right)])^2\right]$$
$$\mathrm{E}[\hat{f}\left(x\right)^2] - \mathrm{E}[\hat{f}\left(x\right)]^2$$

분산은 알고리즘을 학습하는데 있어서
학습의 일관성을 의미.
분산에 의한 에러가 크다는 것은 각각의
알고리즘 학습이 일관성 없이
중구난방으로 이뤄졌음을 뜻한다.
즉, prediction funtion이 학습용
데이터(training data)에 포함된
노이즈까지 학습한 것이라 볼 수 있다.