🤪

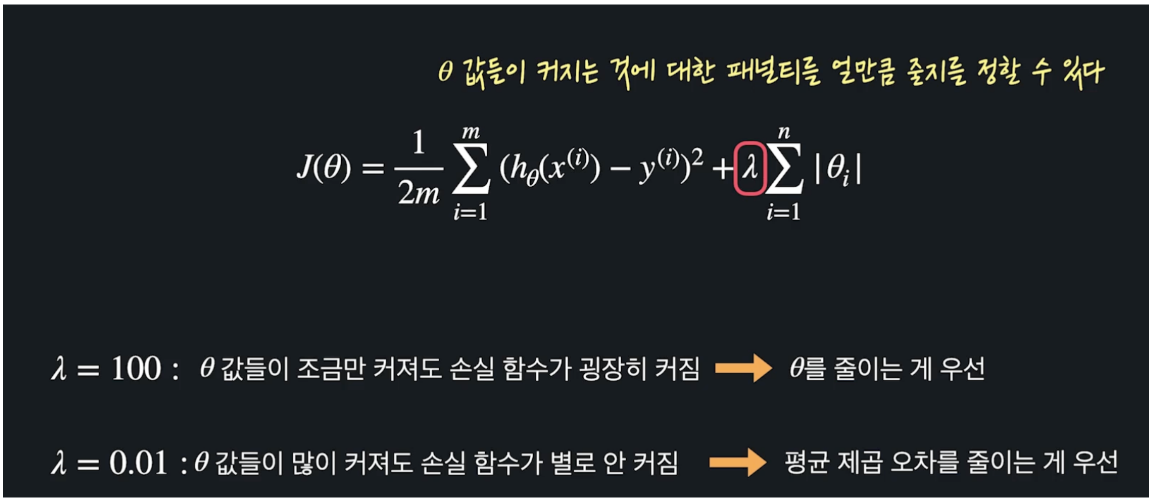
편향(Bias)과 분산(Variance)

- 직선 모델은 너무 간단해서 복잡한 곡선 관계를 학습할 수 없다
    - → 편향이 높다 (training data에 과소적합)
- 모델이 데이터들 사이의 관계를 완벽하게 학습했다
    - → 편향이 낮다 (training data에 과적합)
- 항상 편향이 낮은 것이 좋은 걸까? : X
- 분산
    - 데이터 셋 별로 모델이 얼마나 일관된 성능을 보여주는지
- 정리
    - 편향이 높은 모델은 너무 간단해서 주어진 데이터의 관계를 잘 학습하지 못함
    - 편향이 낮은 모델은 주어진 데이터의 관계를 아주 잘 학습함
    - 분산은 다양한 테스트 데이터가 주어졌을 때 모델의 성능이 얼마나 일관적인지를 나타낸다.
- 일반적으로 편향과 분산은 하나가 줄어들수록 하나는 늘어나는 관계

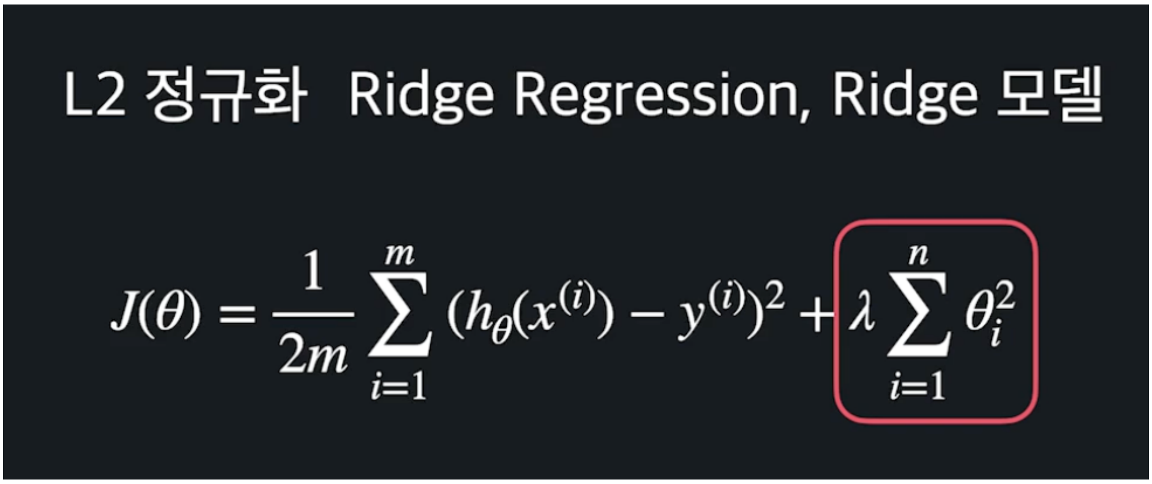
  → 편향-분산 트레이드오프
- 편향과 분산, 다르게는 과소적합과 과적합의 밸런스를 찾아야 함.

정규화

- 가설 함수의 세타(0) 값들이 너무 커지는 걸 방지해서 과적합을 예방하는 방법
- 새로운 기준
    - Traing data에 대한 오차도 작고 세타 값들도 작아야지 좋은 가설 함수다.
- L1 정규화 - Lasso 모델



- L2 정규화 - Ridge 모델




```
LogisticRegression(penalty='none')  # 정규화 사용 안함
LogisticRegression(penalty='l1')  # L1 정규화 사용
LogisticRegression(penalty='l2')  # L2 정규화 사용
LogisticRegression()  # 위와 똑같음: L2 정규화 사용
```

- L1, L2 정규화의 차이점
  - L1 정규화는 여러 세타값들을 0으로 만들어줌. 모델에 중요하지 않다고 생각되는 속성들을 아예 없애줌
  - L2 정규화는 세타값들을 0으로 만들기보다는 조금씩 줄여줌. 모델에 사용되는 속성들을 L1 처럼 없애지는 않음
  - 즉, L1 정규화는 어떤 모델에 쓰이는 속성 또는 변수의 개수를 줄이고 싶을 때 사용
- https://huidea.tistory.com/154