

13장 과적합(overfitting) 피하기

: 초음파 광물 예측 실험을 예시로

17 김찬영, 17 최태규, 18 장윤정, 20 김준수

Contents

20 김준수

초음파 광물 예측 실습: 데이터 확인과 실행

17 김찬영

과적합, 학습셋과 테스트셋이란?

17 최태규

학습셋과 테스트셋 구분, 모델 저장과 재사용

18 장윤정

K겹 교차 검증

0

초음파 광물 예측 실습

1

광물 데이터 파일 : sonar.csv



출처 : 모두의 딥러닝 개정 2판

01

colab 실행하기

COLAB

0

과적합 이해하기

2

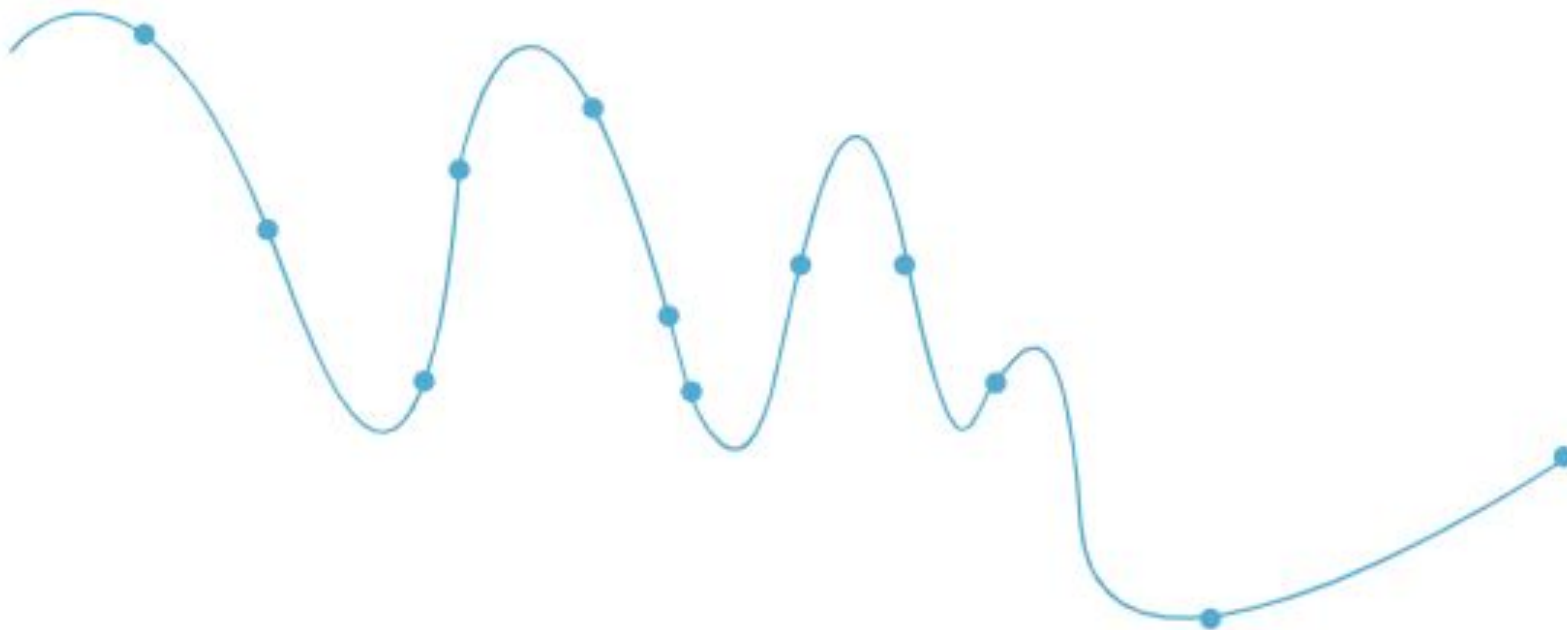
모델이 학습 데이터셋 안에서는 일정 수준 이상의 예측 정확도를 보이지만,
새로운 데이터에 적용하면 잘 맞지 않는 것.

-> 학습 데이터 세트에 속한 각각의 개별 데이터들을 완벽하게 설명하기 위한 모델을 생성한다.



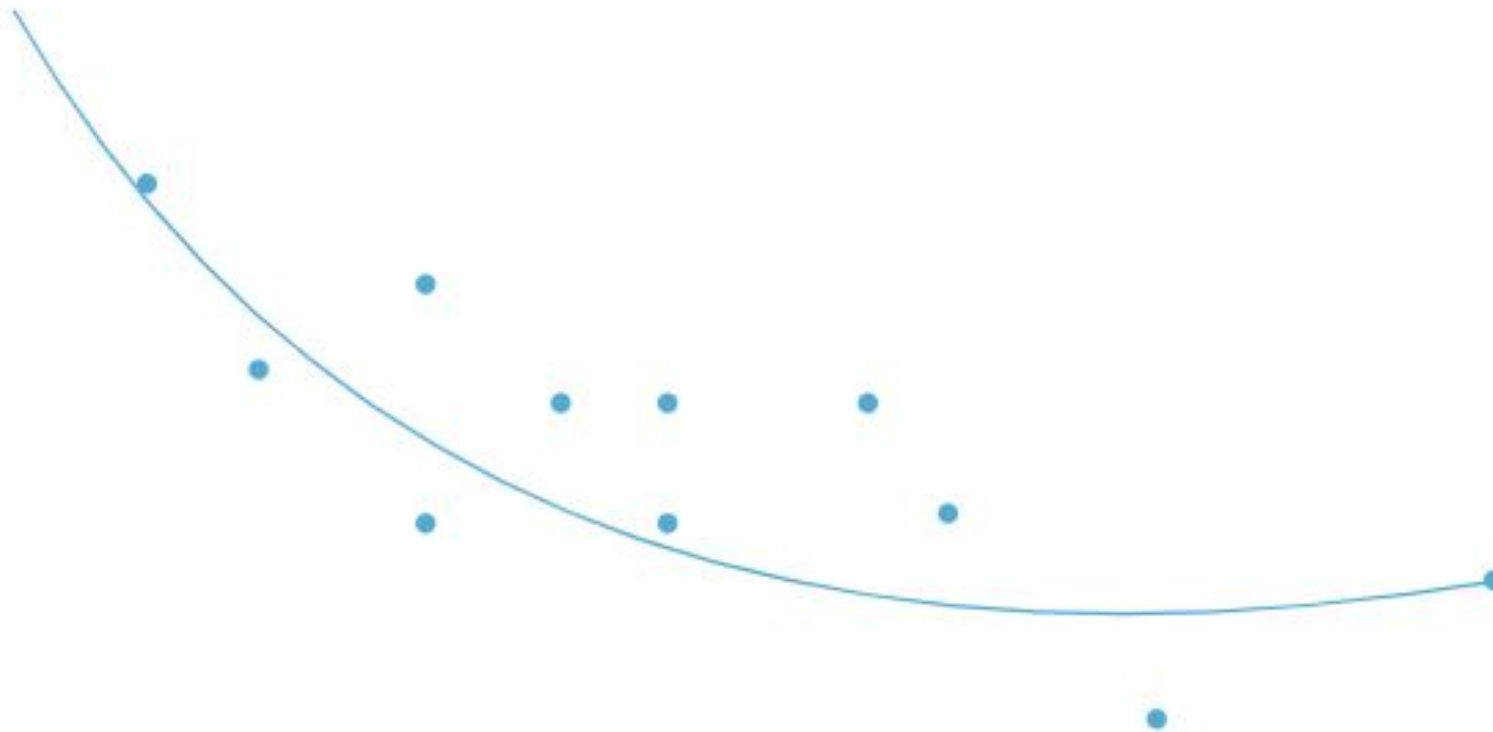
01

과적합 (Overfitting) 이란?



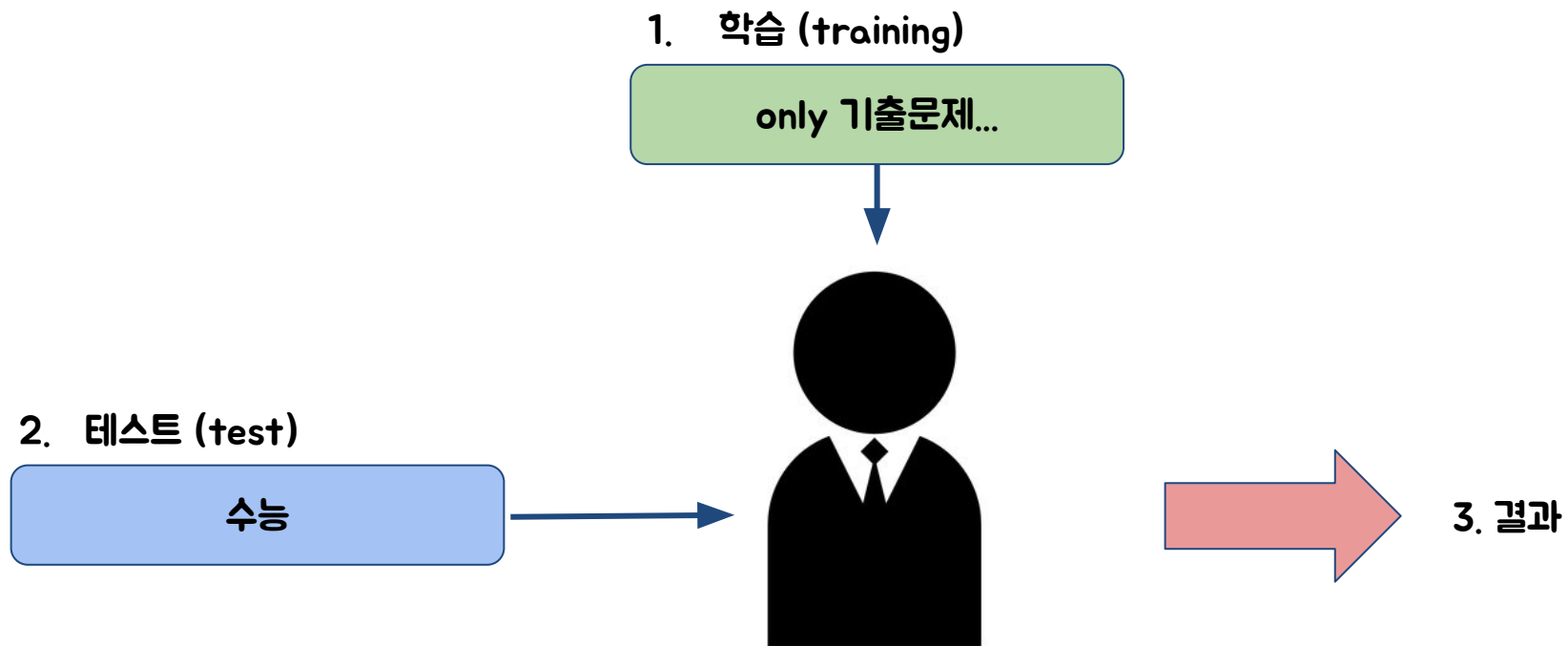
01

과적합 (Overfitting) 이란?

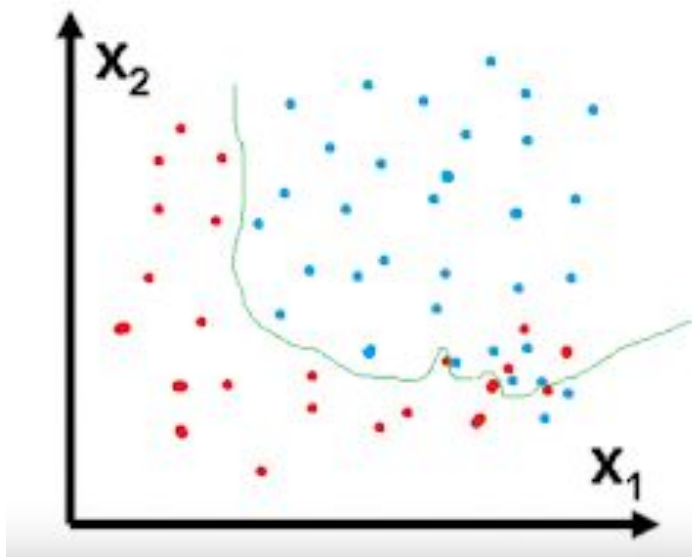


01

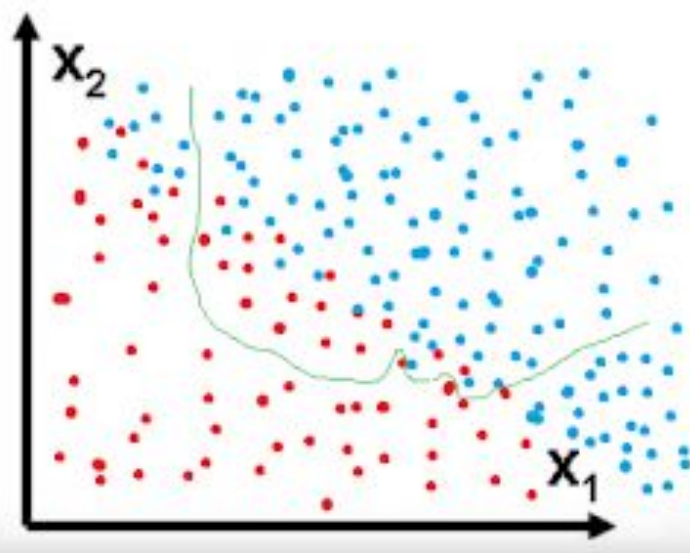
과적합 (Overfitting) 이란?



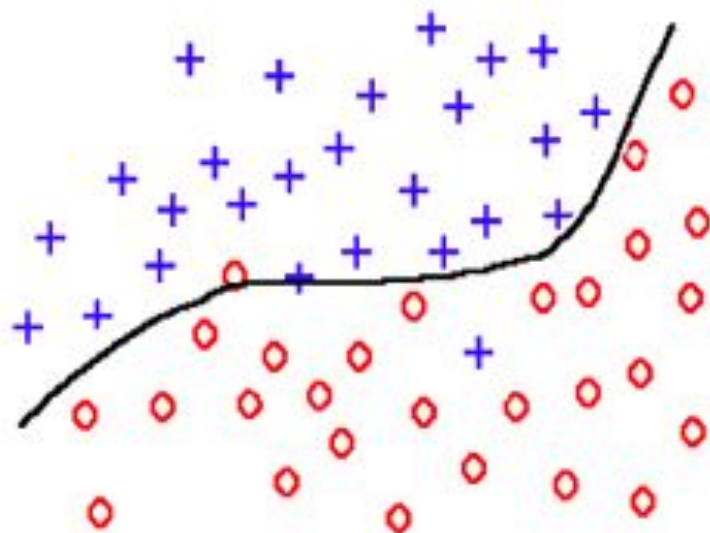
-> 너무 세밀하게 학습 데이터 하나하나를 다 설명하려고 하다보니 정작 **중요한 패턴**을 설명할 수 없게 되는 현상



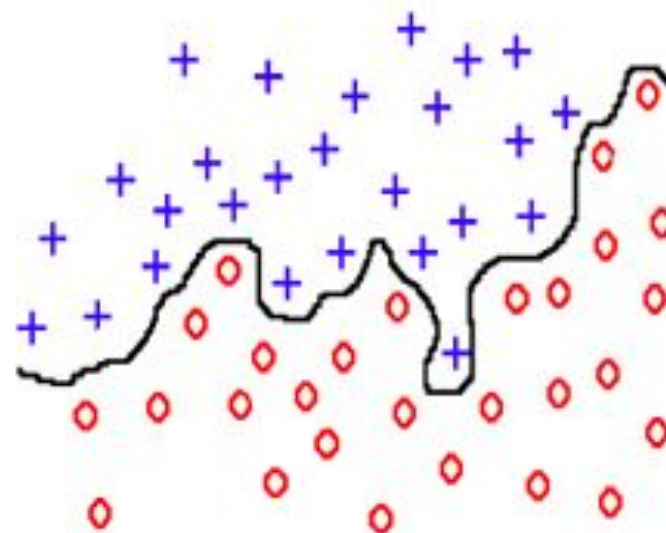
<Train Data>



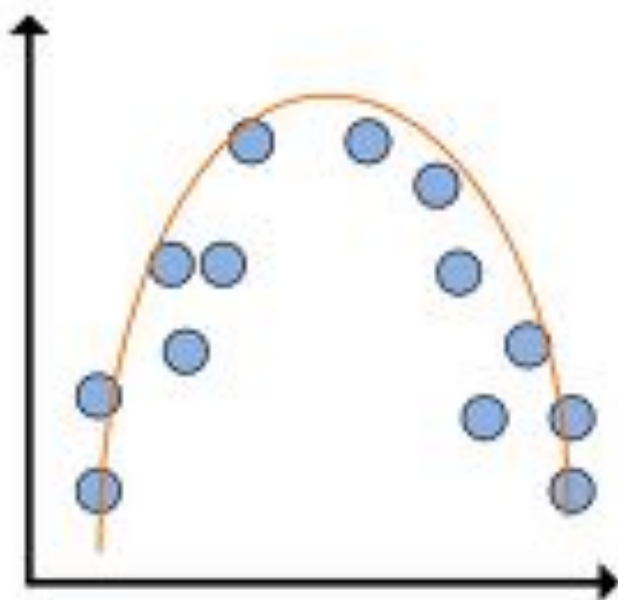
<Test Data>



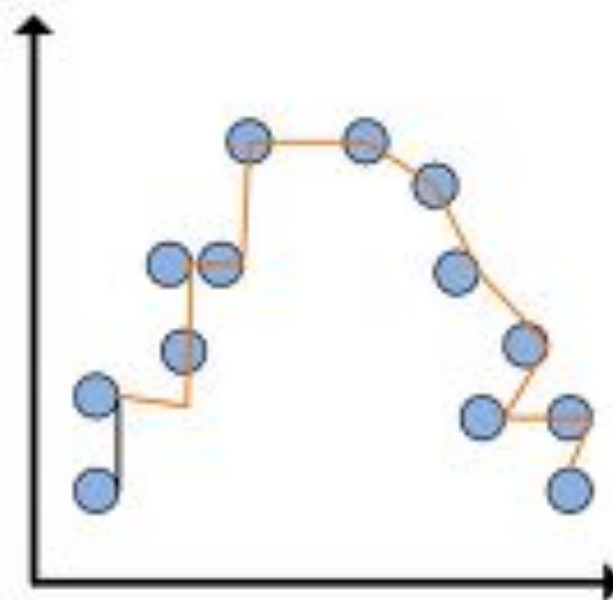
good fit



overfitting



C

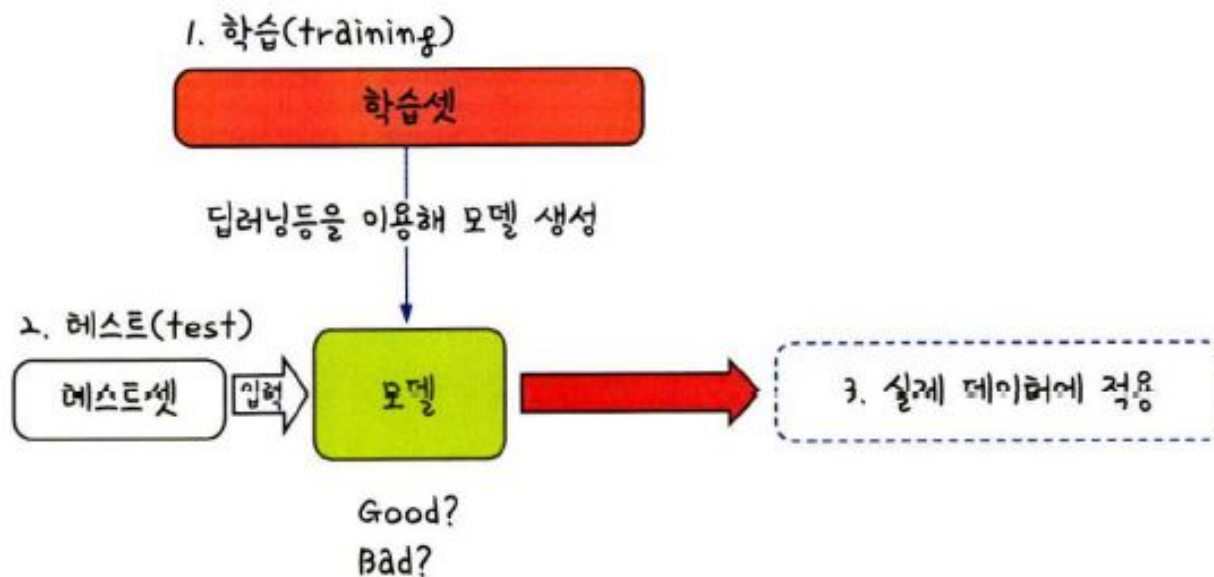
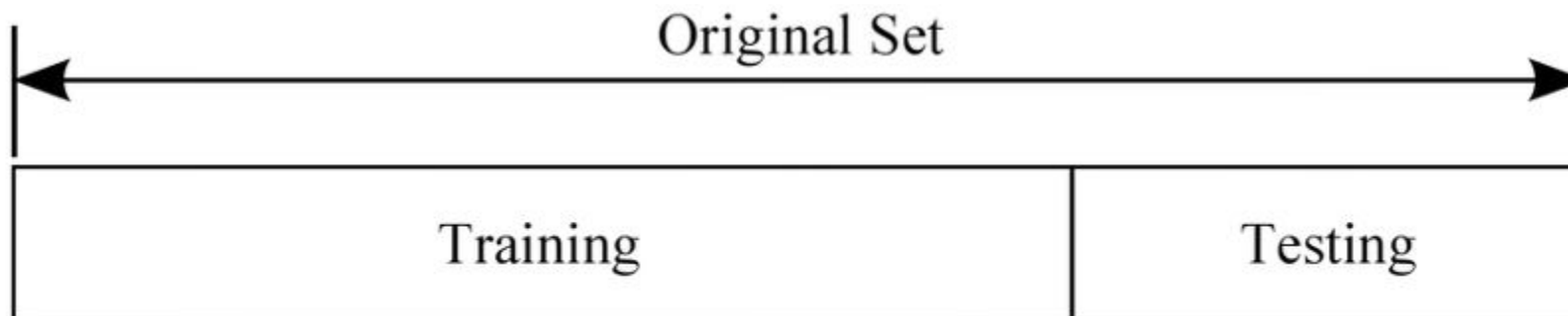


D

0

학습셋과 테스트셋

3



모델 학습에 관여 X

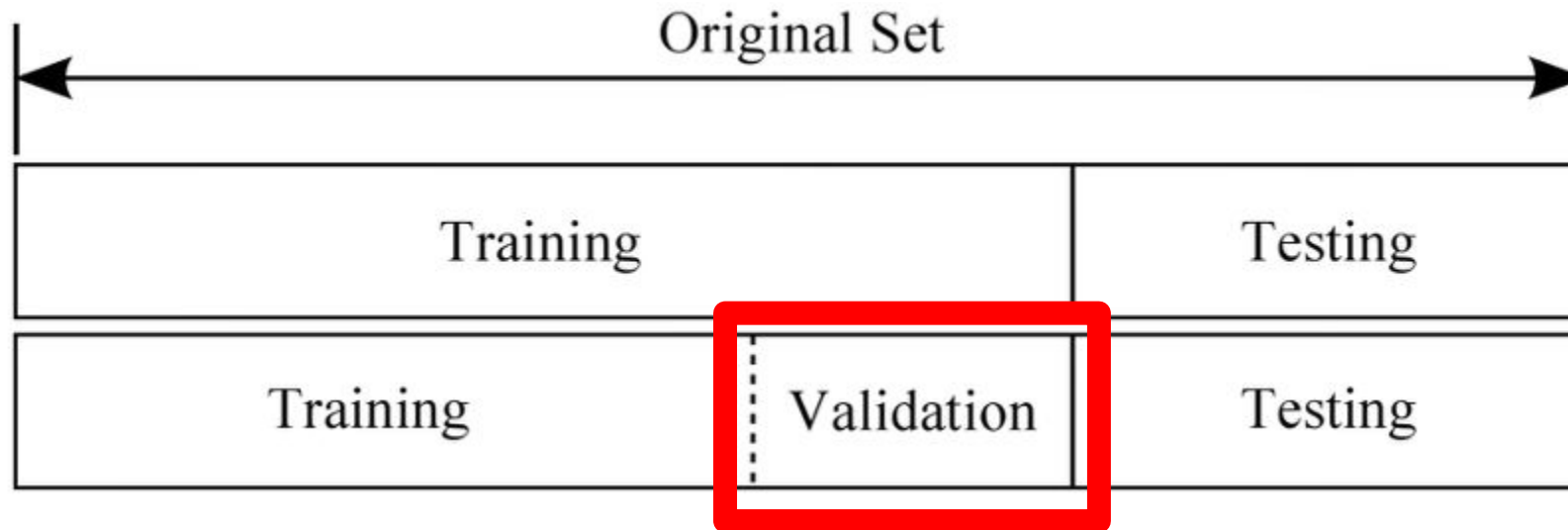
```
from sklearn.model_selection import train_test_split
```

```
# 학습셋과 테스트셋의 구분
```

```
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_  
size=0.3, random_state=seed)
```


01

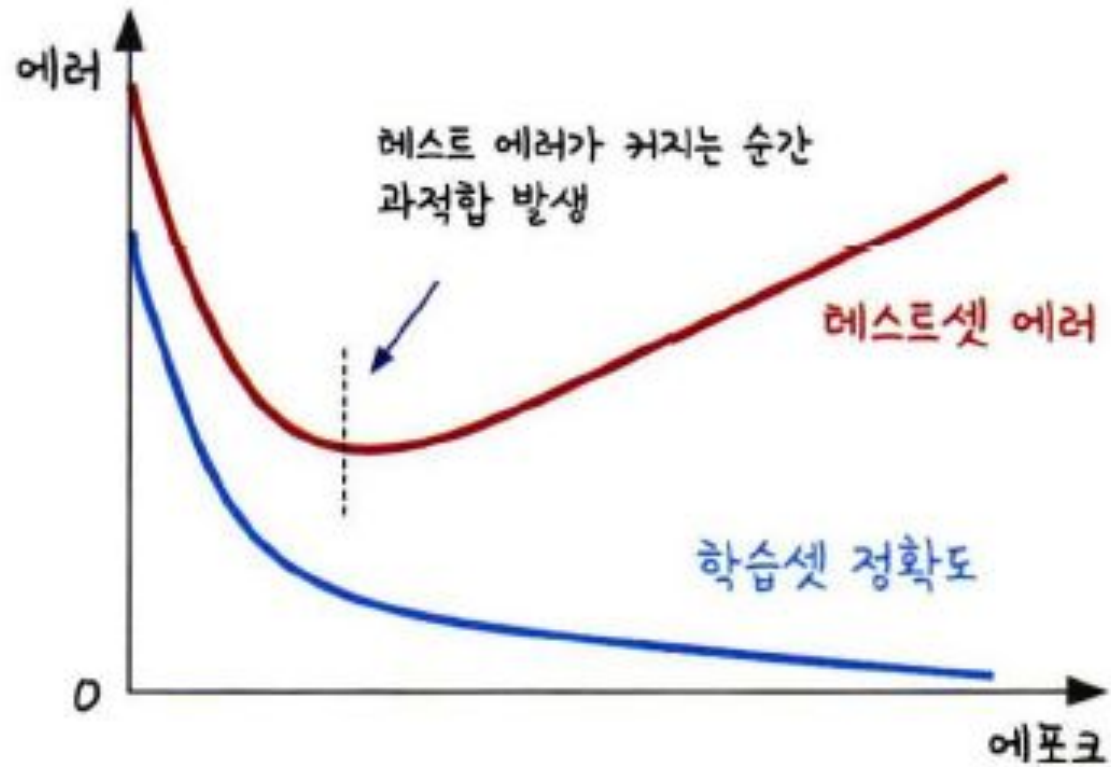
Validation Set



```
1 model.fit(x_train,y_train,epochs=200, batch_size=5, validation_split=0.2)
```

01

Validation Set



자세한 내용은 14장에서..

0

모델 저장과 재사용

4

모델의 저장과 재사용

- 필요성

: 자신이 원하는 구조나 학습량을 가진 모델을 재사용 할 수 있다.

모델의 저장과 재사용

라이브러리

```
from keras.models import load_model
```

모델 저장

```
model.save('my_model.h5')
```

모델 불러오기

```
model = load_model('my_model.h5')
```

함수 안의 모델 삭제

```
del model
```

0

K겹 교차 검증

5

K겹 교차 검증 (K-fold cross validation)

- 필요성

: 데이터 개수가 적을 때, Hold out 평가(70% 학습셋-30% 테스트셋) 방식의 낮은 신뢰성 문제 해결

- K겹 교차 검증이란?

- 가지고 있는 데이터를 동일한 크기의 K겹으로 나눔 (통상 K=10)
- 각 시행 단계에서 하나의 폴드만 테스트셋으로, 나머지는 학습셋으로
- 이 과정을 K번 반복 시행, 최종적으로는 각 폴드를 테스트셋으로 한 번씩 사용
- K번 시행 과정에서 나온 정확도의 평균을 구해 모델 성능 계산



- 장점

- 데이터를 전부 테스트/학습셋으로 사용
- 데이터 개수가 적은 경우 단점 보완
→의 변동성, 편향성 감소

과적합(오버피팅) 방지

교차 검증 (K-fold cross validation) 의 종류

- K겹 교차 검증(k-fold cross validation)

KFold

- 계층화 K겹 교차 검증 (stratified k-fold validation)

StratifiedKFold

```
from sklearn.model_selection import StratifiedKFold
```

```
n_fold = 10
```

```
skf = StratifiedKFold(n_splits=n_fold, shuffle=True, random_  
state=seed)
```

Etc. 셔플 분할 교차 검증, 계층화 셔플 분할 교차 검증, leave-one-out 교차검증, leave-p-out 교차검증

K겹 교차 검증 (K-fold cross validation) 실습 : 초음파 광물 예측하기

colab

Thank you (🌸 ' ◡ ' 🌸)

QnA