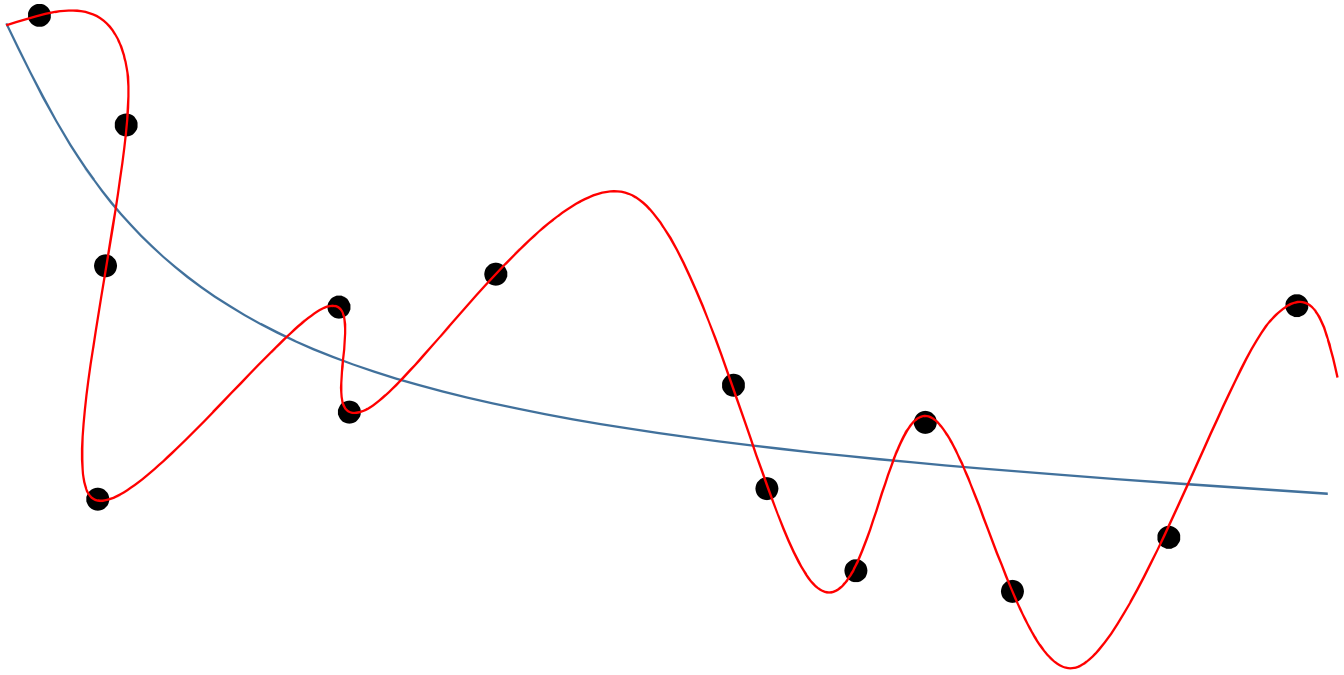
데이터셋

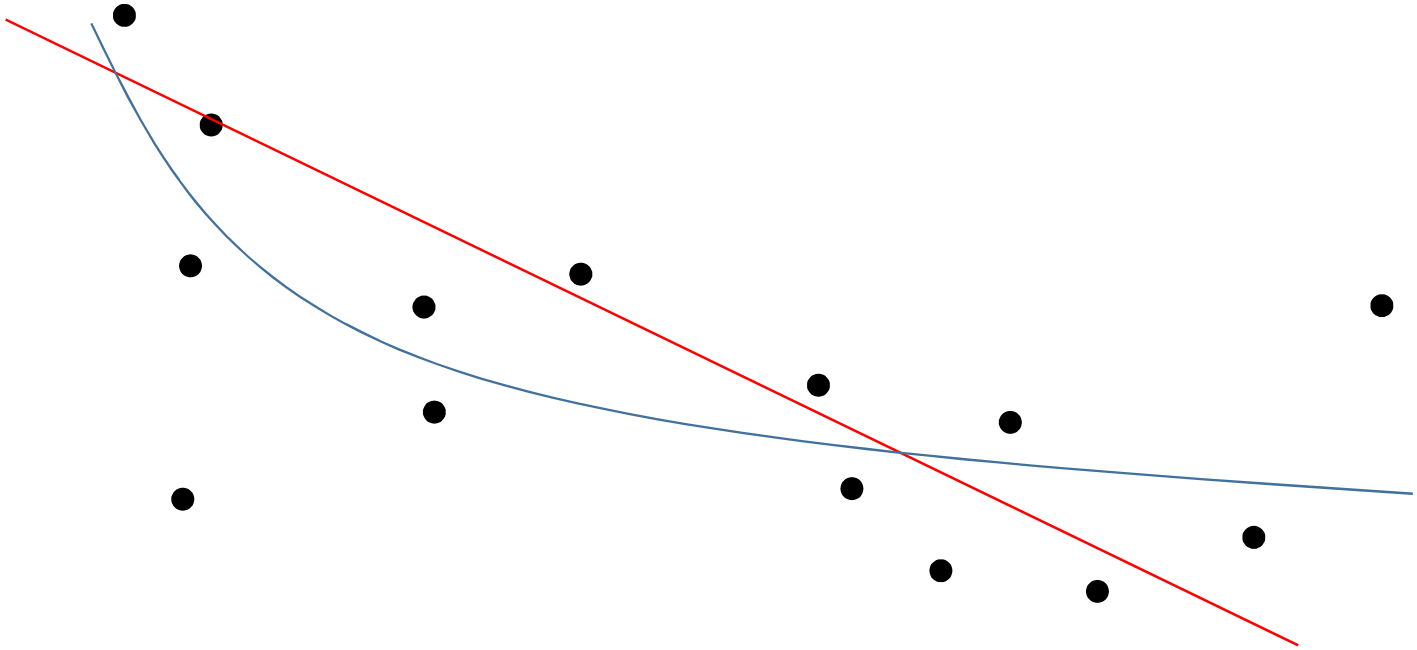
과대적합(Overfitting)

- 모델 파라미터들을 학습 데이터에 너무 가깝게 맞췄을 경우 발생하는 문제
- 학습 데이터에 과하게 학습되어 실제 데이터에서는 오차가 증가하는 현상



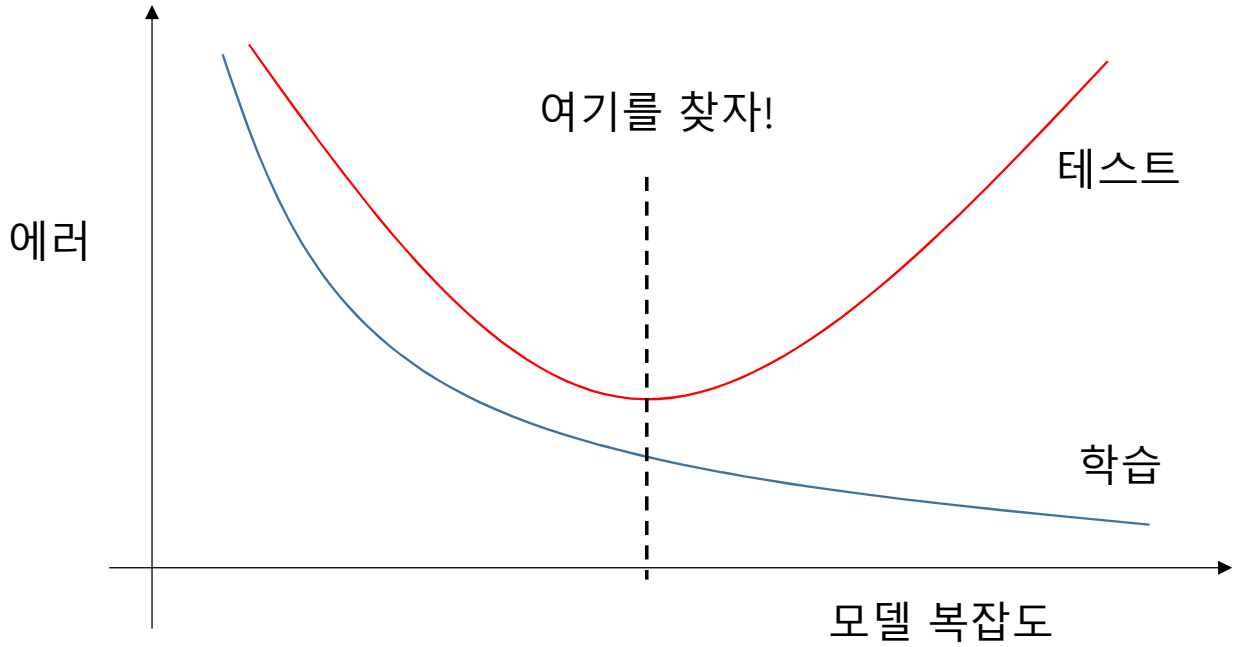
과소적합(Underfitting)

- 모델이 너무 단순하여, 학습 데이터의 특징 및 구조를 충분히 반영하지 못한 경우 발생
- 학습 데이터가 충분하지 않을 경우, 테스트 데이터가 학습 데이터 특징에 대한 유추가 힘듦



모델 최적화

- 모델이 너무 복잡하거나, 학습 데이터에 너무 최적화하여 과대적합일 경우 현실에서 사용하기 힘듦
- 모델이 너무 단순하거나, 학습이 충분하지 않아 과소적합일 경우에도 에러가 높아 사용할 수 없음



데이터셋 분할

- 학습(Train) 데이터 : 모델 학습하면서 파라미터를 찾기 위한 데이터
- 검증(Validation) 데이터 : 학습이 완료된 모델을 검증하기 위한 데이터
- 테스트(Test) 데이터 : 학습 과정과 무관한 모델 성능을 평가하기 위한 데이터

sklearn.model_selection.train_test_split

```
>>> import numpy as np
>>> from sklearn.model_selection import train_test_split
>>> X, y = np.arange(10).reshape((5, 2)), range(5)
>>> X
array([[0, 1],
       [2, 3],
       [4, 5],
       [6, 7],
       [8, 9]])
>>> list(y)
[0, 1, 2, 3, 4]
```

```
>>> X_train, X_test, y_train, y_test = train_test_split(
...     X, y, test_size=0.33, random_state=42)
...
>>> X_train
array([[4, 5],
       [0, 1],
       [6, 7]])
>>> y_train
[2, 0, 3]
>>> X_test
array([[2, 3],
       [8, 9]])
>>> y_test
[1, 4]
```

https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.train_test_split.html#sklearn.model_selection.train_test_split

Parameters

sklearn.model_selection.train_test_split(*arrays, test_size=None, train_size=None, random_state=None, shuffle=True, stratify=None)

arrays : *sequence of indexables with same length / shape[0]
Allowed inputs are lists, numpy arrays, scipy-sparse matrices or pandas dataframes.

test_size : *float or int, default=None*
If float, should be between 0.0 and 1.0 and represent the proportion of the dataset to include in the test split. If int, represents the absolute number of test samples. If None, the value is set to the complement of the train size. If `train_size` is also None, it will be set to 0.25.

train_size : *float or int, default=None*
If float, should be between 0.0 and 1.0 and represent the proportion of the dataset to include in the train split. If int, represents the absolute number of train samples. If None, the value is automatically set to the complement of the test size.

random_state : *int, RandomState instance or None, default=None*
Controls the shuffling applied to the data before applying the split. Pass an int for reproducible output across multiple function calls. See Glossary.

shuffle : *bool, default=True*
Whether or not to shuffle the data before splitting. If shuffle=False then stratify must be None.

stratify : *array-like, default=None*
If not None, data is split in a stratified fashion, using this as the class labels. Read more in the User Guide.

https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.train_test_split.html#sklearn.model_selection.train_test_split

학습데이터 및 테스트데이터 구분

```python
import pandas as pd
data3 = pd.read_csv("data/health.csv")
data = data3[["H", "W"]]
t = data3["T"]
```

```python
from sklearn.model_selection import train_test_split
train_data, test_data, train_target, test_target = train_test_split(
    data, t, test_size = 0.3, random_state = 42)
print("Total-Data\n", data)
print("Train-Data\n", train_data)
print("Test-Data\n", test_data)
print("Train-Target\n", train_target)
print("Test-Target\n", test_target)
```

```python
from sklearn.neighbors import KNeighborsClassifier
kn = KNeighborsClassifier(n_neighbors = 5, p = 2)
kn.fit(train_data, train_target)
print("Train-Eval:", kn.score(train_data, train_target))
print("Test-Eval :", kn.score(test_data, test_target))
```

```
H,W,T
130.1,30.7,1
120.5,29.2,1
127.3,25.4,1
122.9,23.0,1
126.0,25.8,1
152.8,49.9,0
155.9,46.2,0
158.5,60.0,0
156.6,62.2,0
150.1,49.4,0
```

Train-Data

	H	W	
0	130.1	30.7	1
7	158.5	60.0	0
2	127.3	25.4	1
9	150.1	49.4	0
4	126.0	25.8	1
3	122.9	23.0	1
6	155.9	46.2	0

Test-Data

	H	W	
8	156.6	62.2	0
1	120.5	29.2	1
5	152.8	49.9	0

샘플링 편향

7:3

H,W,T
130.1,30.7,1
120.5,29.2,1
127.3,25.4,1
122.9,23.0,1
126.0,25.8,1
152.8,49.9,0
155.9,46.2,0
158.5,60.0,0
156.6,62.2,0
150.1,49.4,0

Train-Data

	H	W	
0	130.1	30.7	1
7	158.5	60.0	0
2	127.3	25.4	1
9	150.1	49.4	0
4	126.0	25.8	1
3	122.9	23.0	1
6	155.9	46.2	0

Test-Data

	H	W	
8	156.6	62.2	0
1	120.5	29.2	1
5	152.8	49.9	0

Good

7:3

학습데이터

130.1,30.7,1
120.5,29.2,1
127.3,25.4,1
122.9,23.0,1
126.0,25.8,1
152.8,49.9,0
155.9,46.2,0

테스트데이터

158.5,60.0,0
156.6,62.2,0
150.1,49.4,0

5:5

학습데이터

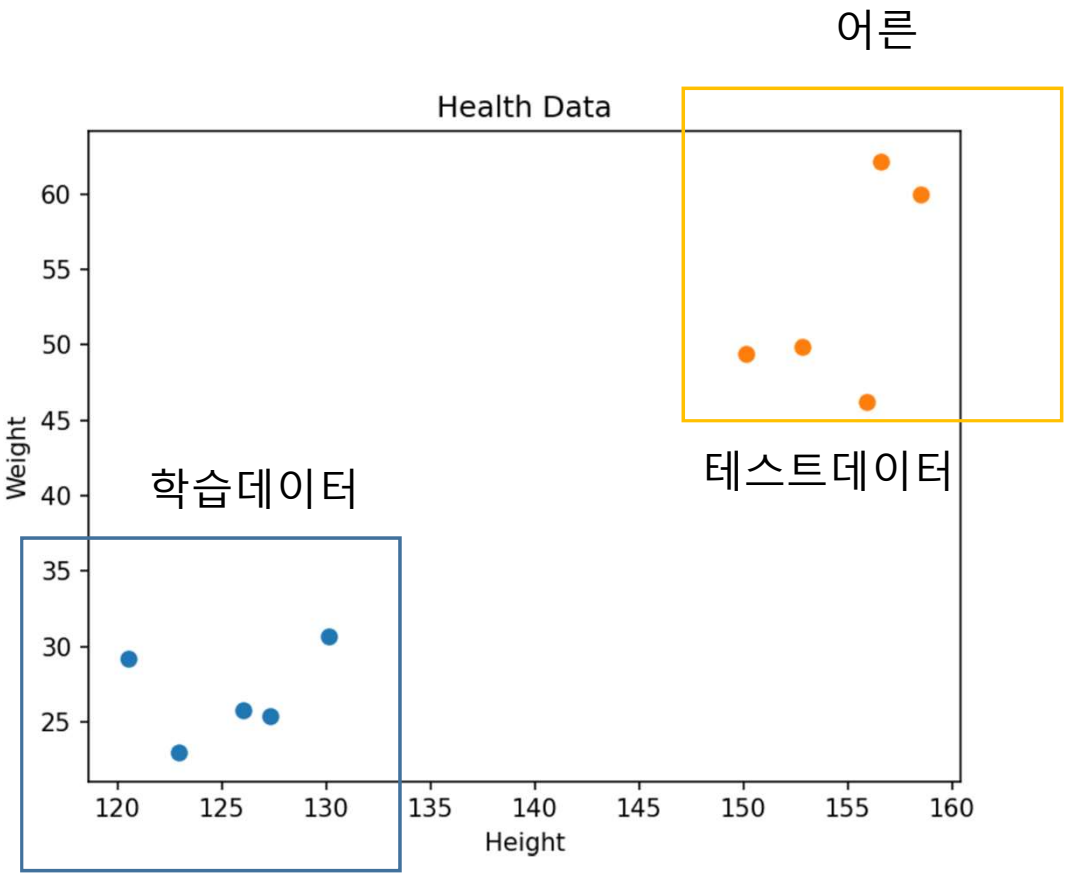
130.1,30.7,1
120.5,29.2,1
127.3,25.4,1
122.9,23.0,1
126.0,25.8,1

테스트데이터

152.8,49.9,0
155.9,46.2,0
158.5,60.0,0
156.6,62.2,0
150.1,49.4,0

계층적 샘플링

- 샘플링 편향을 방지하기 위해 원 데이터셋의 클래스 비율에 맞게 데이터셋 분할
- 무작위 샘플링은 하지만, 샘플링 데이터의 비율을 일정하게 유지
- 무직위 샘플링 시에도 비율은 유지될 가능성이 높지만, 100% 안전한 경우는 없음




```
from sklearn.model_selection import train_test_split
train_data, test_data, train_target, test_target = train_test_split(
    data, t, test_size = 0.3, random_state = 42, stratify = t)
```

참고자료

- 지능기전공학부 최유경 교수님 자료, https://github.com/sejongresearch/2021.MachineLearning
- 코랩(Colab), https://colab.research.google.com/
- 파이썬(Python), https://www.python.org/doc/
- 사이킷런(sckit-learn), https://scikit-learn.org/stable/index.html
- 판다스(pandas), https://pandas.pydata.org/
- 맷플롯립(matplotlib), https://matplotlib.org/
- 씨본(seaborn), https://seaborn.pydata.org/
- 캐글(Kaggle), https://www.kaggle.com/
- 넘파이(numpy), https://numpy.org/doc/stable/
- 스택오퍼플러우(stackoverflow), https://stackoverflow.com/