

# 머신러닝(Machine Learning)

지도학습 알아보기(knn, linear regression)

# 목 차

01 머신러닝

02 k-최근접 이웃(k-Nearest Neighbors)

03 선형모델

04 하이퍼 파라미터

05 선형회귀

06 릿지 회귀와 라쏘 회귀

07 ElasticNet(엘라스틱넷)

# 01 머신러닝(Machine Learning)

- ▶ 머신러닝(Machine Learning)은 지도학습과 비지도학습으로 나뉘어진다.
- ▶ 지도학습은 예측하려는 값이 존재하는 것이고, 비지도학습은 존재하지 않는다.
- ▶ 지도학습은 다시 회귀(regression)과 분류(classification)으로 나뉘어진다.
- ▶ 머신러닝 시스템의 종류의 다른 범주로 준지도학습, 강화 학습 등이 있습니다.

# 01 머신러닝(Machine Learning)

- ▶ 지도학습(supervised learning)
  - ▶ 머신러닝 방법 중의 하나.
  - ▶ 지도학습은 입력과 출력 샘플 데이터가 있다.
  - ▶ 비지도학습은 입력이 있고 출력 샘플 데이터가 없다.

# 01 머신러닝(Machine Learning)

## ▶ 분류(classification)

- ▶ 분류는 미리 정의된, 가능성 있는 여러 클래스 레이블(class label)중 하나를 예측
- ▶ 두 개의 클래스로 분류하는 이진 분류(binary classification)
  - 예/아니오 또는 생존/사망 등으로 분류
  - 이진 분류에서 한 클래스를 양성(positive) 클래스, 다른 하나를 음성(negative) 클래스라고 한다.
- ▶ 셋 이상의 클래스로 분류하는 다중 분류(multiclass classification)

# 01 머신러닝(Machine Learning)

## ▶ 과대 적합, 과소적합, 일반화

▶ 모델이 처음보는 데이터에 대해 정확하게 예측할 수 있으면 이를 훈련 세트에서 테스트 세트로 **일반화(generalization)**되었다고 함.

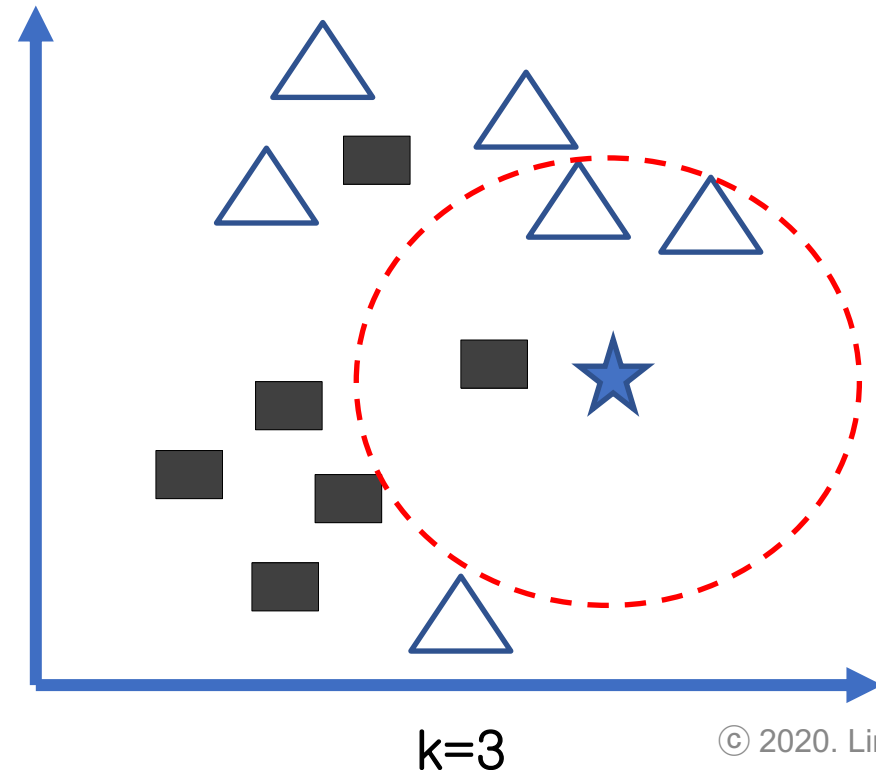
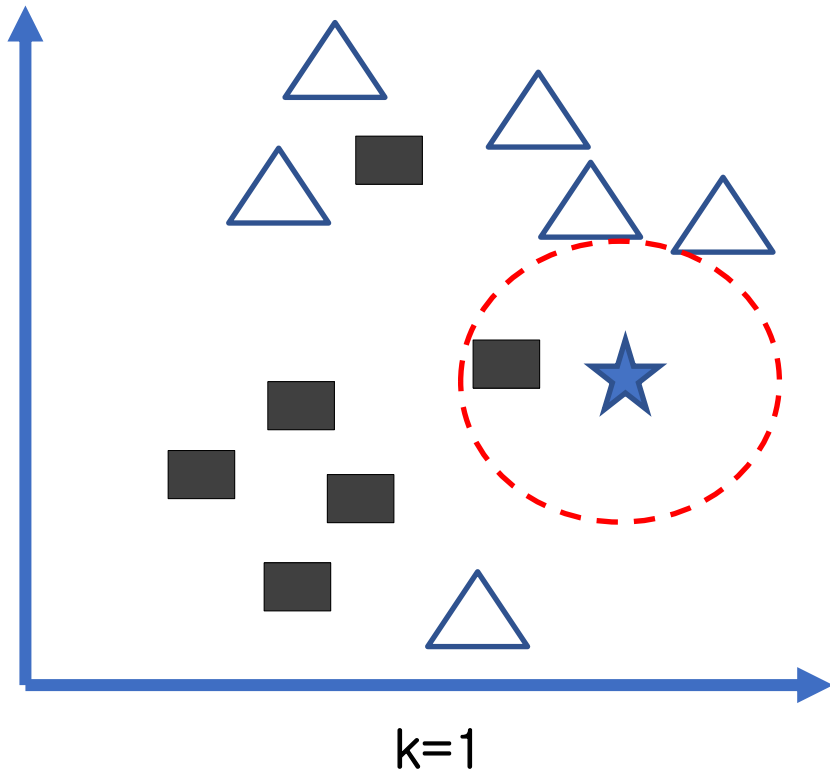
▶ 가진 정보를 너무 사용해서(과도 학습) 복잡한 모델을 만든 것을 **과대적합(overfitting)**이라 한다. 모델이 훈련 세트의 너무 최적화 되어, 새로운 데이터에 적합하지 않음. 이를 일반화 되지 않았다고 이야기하기도 함.

▶ 반대로 학습이 안된 너무 간단한 모델이 선택되는 것을 **과소 적합(underfitting)**라고 한다.

## 02 k-최근접 이웃(k-Nearest Neighbors)

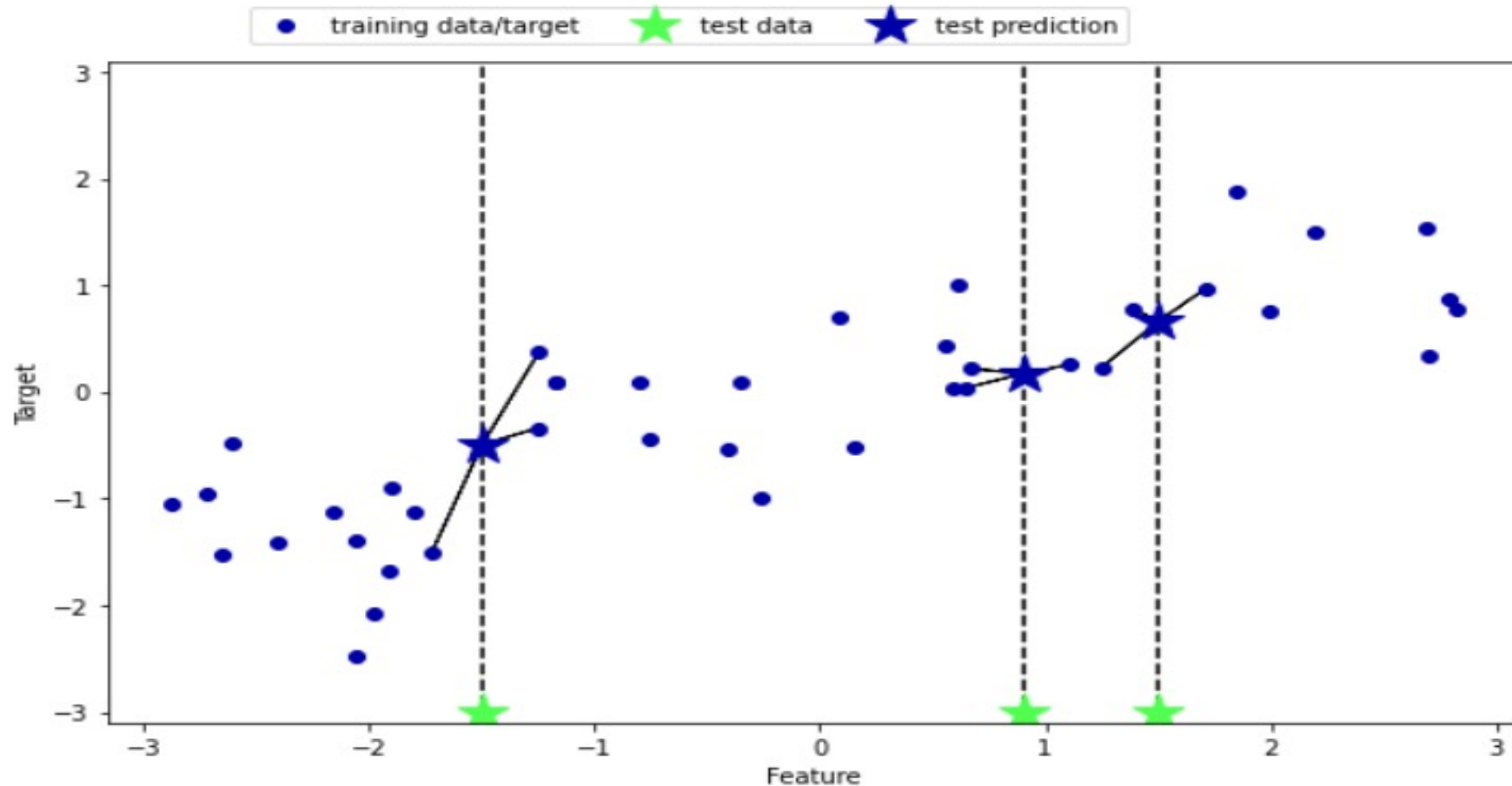
### ▶ 지도학습 - 분류(Classification)

- 새로운 데이터는 해당 거리안의 데이터가 가장 많이 있는 클래스로 분류하게 된다.
- 가장 가까운 학습 데이터 포인트를 찾아 이를 예측에 활용한다.



## 02 k-최근접 이웃(k-Nearest Neighbors)

- ▶ 가장 가까운 훈련 데이터 포인트를 찾아 이를 예측에 활용한다. (회귀의 경우)



k=3의 경우, 가장 가까운 데이터 3개를 찾는다.



## 02 k-최근접 이웃(k-Nearest Neighbors)

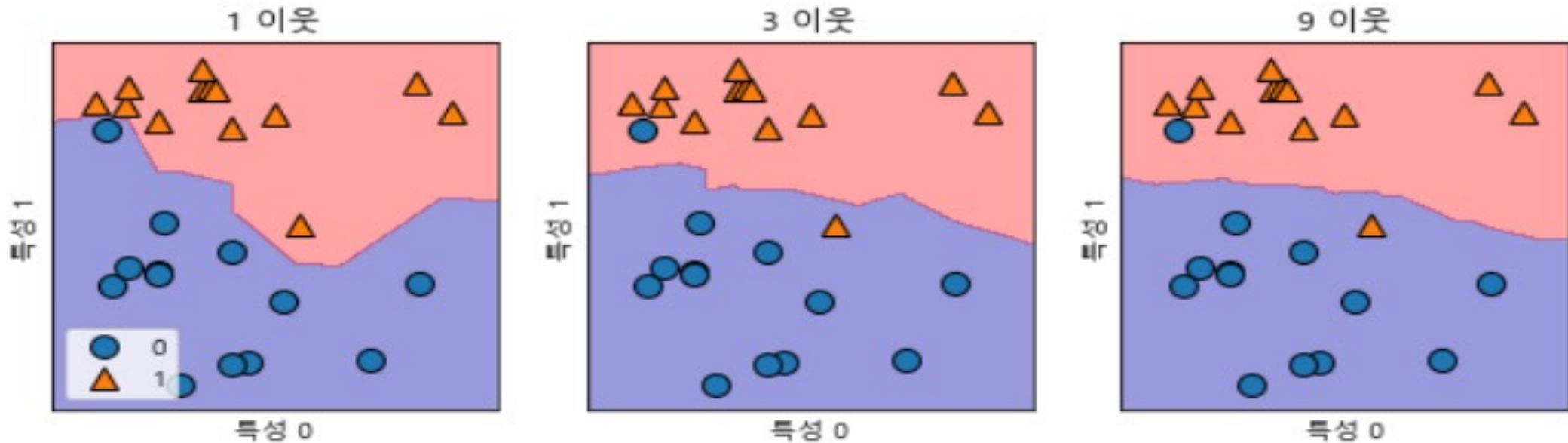
### ▶ 지도학습 - 회귀(Regression)

- A. 가장 가까운 훈련 데이터 포인트를 찾아 이를 예측에 활용한다.
- B. 새로운 데이터는 해당 거리안의 k개 데이터가 예측하는 타겟(목표변수)의 값의 평균으로 예측
- C. 이웃 간의 거리를 계산할 때 특성마다 값의 범위가 다를 경우, 범위가 작은 특성에 영향을 받는다.  
따라서 k-NN 알고리즘을 사용할 때는 특성들이 같은 스케일을 갖도록 정규화하는 것이 일반적이다.

## 02 k-최근접 이웃(k-Nearest Neighbors)

### ▶ k의 값에 따른 결정 경계(decision boundary)

새로운 데이터가 어느 값으로 분류될지 결정되는 경계를 말한다.



이웃의 수를 늘릴 수록 결정경계는 더 부드러워집니다.

## 02 k-최근접 이웃(k-Nearest Neighbors)

### ▶ knn 모델의 장점

- A. 매우 쉬운 모델이다.
- B. 많이 조정하지 않아도 자주 좋은 성능 발휘

### ▶ knn 모델의 단점

- A. 수백 개 이상의 많은 특성을 가진 데이터 셋을 가진 데이터 셋에는 잘 동작하지 않음.
- B. 특성(feature) 값 대부분이 0인 데이터 셋은 잘 동작하지 않음.

## 03 선형모델(linear model)

- ▶ 선형 모델은 약 200년 전에 개발되었고, 가장 간단하고 오래된 회귀용 선형 알고리즘.
- ▶ 선형모델은 입력 특성(feature)에 대한 선형 함수를 만들어 예측 수행

$$\hat{y}(\text{예측값}) = w_1 * x_1 + w_2 * x_2 + \dots + w_p * x_p + b$$

$\hat{y}$  : 모델이 만들어낸 예측값

$x_i$  : 특성

$w_i$  : 각 특성에 대한 기울기(가중치)

- ▶ 선형모델은  $w$ (가중치 or 계수)와  $b$ (편향(offset) 또는 절편)를 학습하여 정한다.
- ▶ 선형모델은 특성이 하나일 때는 직선, 두 개일 때는 평면, 더 높은 차원에서는 초평면이 된다.

# 04 하이퍼 파라미터(hyperparameter)

## ▶ 모델 파라미터 or 계수

머신러닝에서 알고리즘이 주어진 데이터로 부터 학습하는 파라미터

## ▶ 하이퍼파라미터(hyperparameter) or 매개변수

모델이 학습할 수 없는 파라미터로서 사람이 직접 설정해 주어야 하는 파라미터

(예) knn의 k 값 등.

## 05 선형 회귀(linear regression)

- ▶ 선형회귀는 평균제곱오차(MSE)를 최소화하는 파라미터(w,b)를 찾는다.

$$MSE = \frac{1}{n(\text{샘플개수})} \sum_{i=1}^n (y_i(\text{실제값}) - \hat{y}_i(\text{예측값}))^2$$

- ▶ 선형 회귀는 매개변수(직접 지정하는 변수)가 없는 것이 장점
  - 따라서 모델의 복잡도를 제어할 방법이 없음.

# 05 선형 회귀(linear regression)- 결정계수

## ▶ 결정계수( $R^2$ )

- (1) scikit-learn의 score메소드에서 결정계수 값을 확인할 수 있다.
- (2) 결정계수는 회귀 모델에서 **예측의 적합도를 측정**한 것이다.

## ▶ 결정계수( $R^2$ )

$$R^2 = 1 - \frac{\sum (y - \hat{y})^2}{\sum (y - \bar{y})^2}$$

$$\left[ \begin{array}{l} y : \text{타깃값} \\ \hat{y} : \text{예측값} \\ \bar{y} : \text{평균} \end{array} \right]$$

선형회귀에 대한 과대적합 해결 방법은 있을까?



## 06 정규화항을 통한 일반화

- ▶ 과적합을 해소하기 위해 우리는 **정규화 항**을 사용한다.

MSE + regular-term(정규화 항)

- ▶ 선형회귀 모델

$$\hat{y}(\text{예측값}) = w_1 * x_1 + w_2 * x_2 + \dots + w_p * x_p + b$$

- ▶ 오차함수(cost function)

$$\sum_{i=1}^M (y_i - \hat{y}_i)^2 = \sum_{i=1}^M (y_i - \sum_{j=0}^p w_j * x_{ij})^2$$

# 07 라쏘 회귀(Lasso)와 릿지회귀(Ridge)

## ▶ 라쏘 회귀(Ridge) - L1규제

A. 선형 모델,

B. 가중치( $w$ ..)의 절대값을  $w$ 의 모든 원소가 0에 가깝게, 어떤 것은 정말 0이 된다.

- 일부 계수를 0으로 만들면 모델을 이해하기 쉽고 모델의 가장 중요한 특성이 드러남

- 자동으로 특성 선택(feature selection)이 이루어짐.

C. 모든 특성(feature)이 특성에 영향을 주는 영향을 최소한으로 함.

이런 제약을 우리는 규제(regularization)이라고 한다. 라쏘 회귀에 사용하는 규제를 L1규제라 한다.

D. Lasso는 계수를 얼마나 강하게 0으로 규제할지  $\alpha$  매개변수를 이용.

# 07 라쏘 회귀(Lasso)와 릿지회귀(Ridge)

## ▶ 릿지 회귀(Ridge) - L2규제

- A. 선형 모델,
- B. 가중치( $w$ ..)의 절대값을 가능한 한 작게 만든다.  $w$ 의 모든 원소가 0에 가깝게
- C. 모든 특성(feature)가 특성에 영향을 주는 영향을 최소한으로 만든다.  
이런 제약을 우리는 규제(regularization)이라고 한다.  
릿지 회귀에 사용하는 규제를 **L2규제**라 한다.
- D. 데이터를 충분히 주면 릿지 회귀에서의 규제항(alpha)은 덜 중요해 진다.

# 07 라쏘 회귀(Lasso)와 릿지회귀(Ridge)

## ▶ 라쏘 회귀 - L1규제

$$\sum_{i=1}^M (y_i - \hat{y}_i)^2 = \sum_{i=1}^M (y_i - \sum_{j=0}^p w_j x_{ij})^2 + \lambda \sum_{j=0}^p |w_j|$$

## ▶ 릿지 회귀 - L2규제

$$\sum_{i=1}^M (y_i - \hat{y}_i)^2 = \sum_{i=1}^M (y_i - \sum_{j=0}^p w_j x_{ij})^2 + \lambda \sum_{j=0}^p w_j^2$$

# 08 ElasticNet(엘라스틱넷)

## ▶ 엘라스틱 넷

- A. Lasso와 Ridge을 결합한 모델.
- B. 가장 좋은 성능을 내지만 L1과 L2규제를 위한 매개변수 두개를 조정해야 함.