

머신러닝(Machine Learning)

지도학습 알아보기(knn, linear regression)

01 History

변경 이력

No	version	날짜	내용
1	ver 1.11	2020-05-14	의사결정 트리 부분 변경
2	ver 1.12	2020-09-10	내용 업데이트
3	ver 1.13	2020-10-23	결정계수 추가
4	ver 1.13	2020-12-02	릿지, 랏소 회귀 내용 추가
5	ver 1.14	2021-07-13	Font(나눔바른고딕)
6			
7			

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)

08 결정 트리(decision tree)

09 분류용 선형 모델

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)

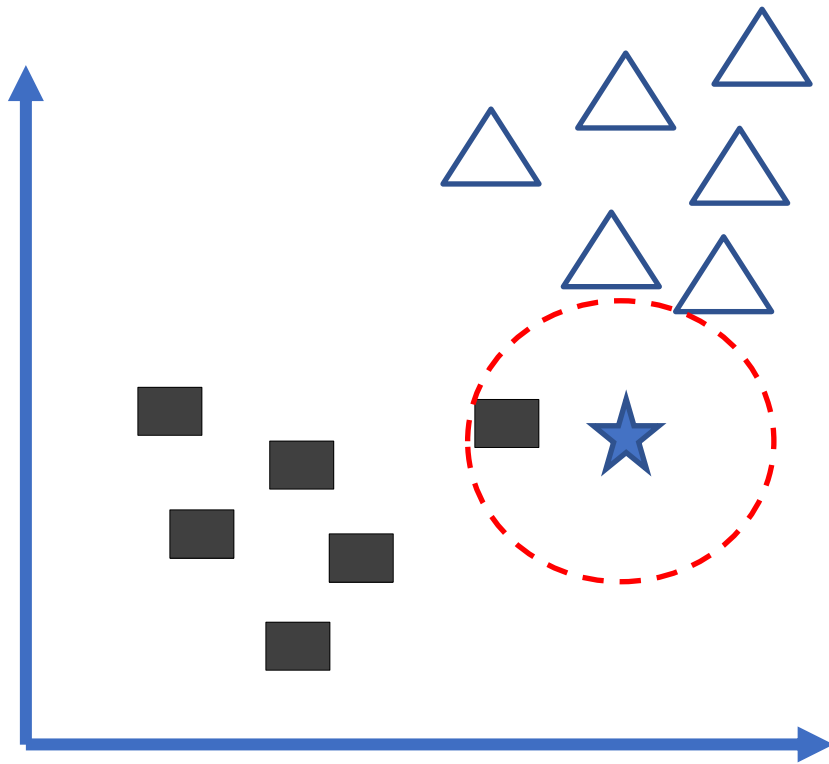
01 머신러닝(Machine Learning)

과대 적합, 과소적합, 일반화

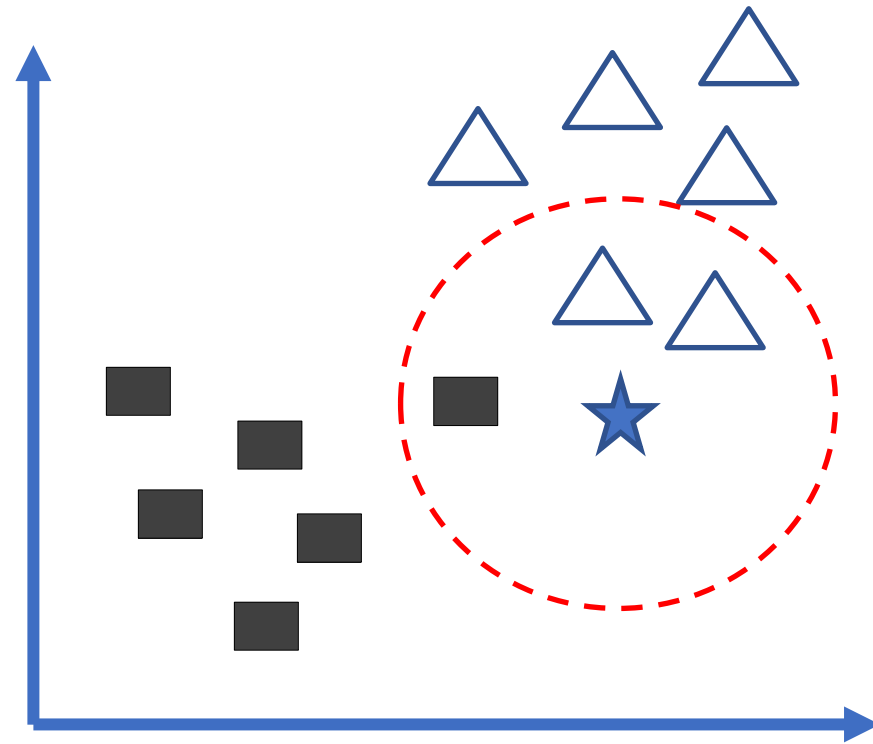
- ▶ 모델이 처음보는 데이터에 대해 정확하게 예측할 수 있으면 이를 훈련 세트에서 테스트 세트로 **일반화(generalization)** 되었다고 함.
- ▶ 가진 정보를 너무 사용해서 너무 복잡한 모델을 만드는 것을 **과대적합(overfitting)**이라 한다. 모델이 훈련 세트의 각 샘플에 가깝게 맞춰져서 새로운 데이터에 일반화 되기 어려울 때 발생.
- ▶ 반대로 너무 간단한 모델이 선택되는 것을 **과소 적합(underfitting)**라고 한다.

02 k-최근접 이웃(k-Nearest Neighbors)

- ▶ 가장 가까운 훈련 데이터 포인트를 찾아 이를 예측에 활용한다. (분류의 경우)
- ▶ 새로운 데이터는 해당 거리안의 데이터가 가장 많이 있는 클래스로 분류하게 된다. (분류)



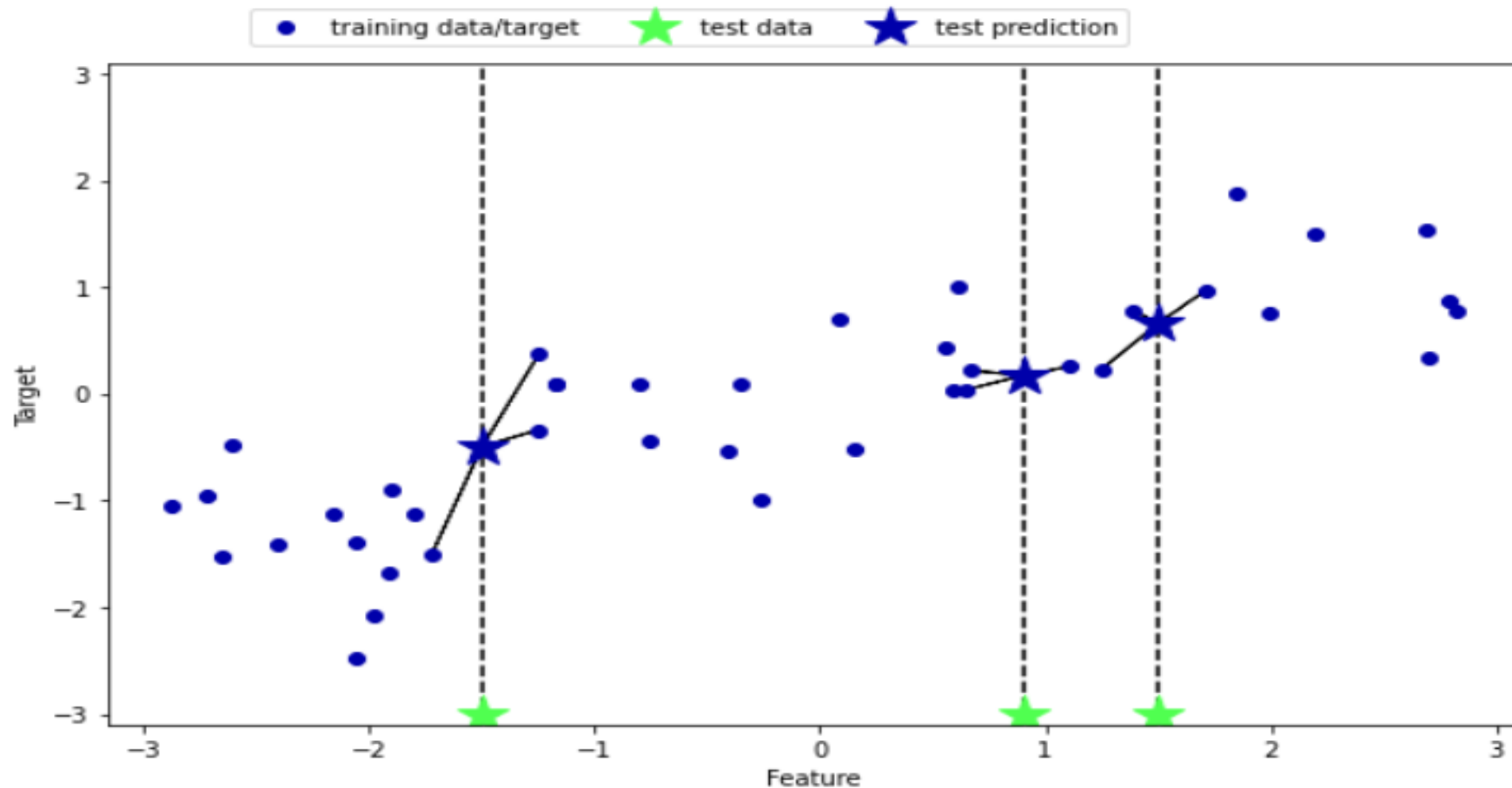
k=1



k=3

02 k-최근접 이웃(k-Nearest Neighbors)

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



k=3의 경우

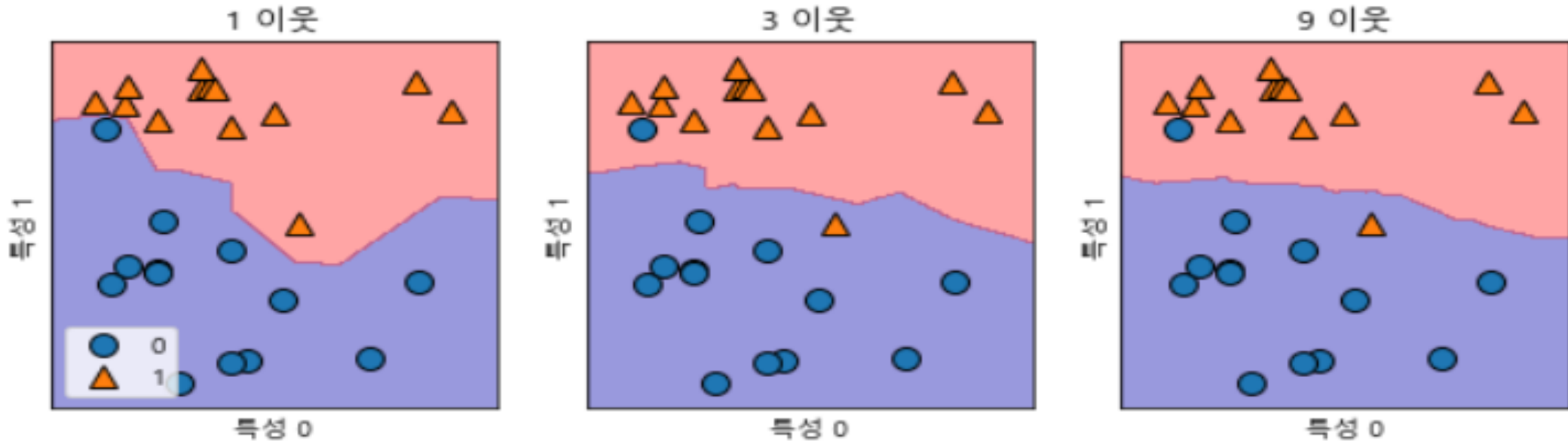
02 k-최근접 이웃(k-Nearest Neighbors)

- ▶ 가장 가까운 훈련 데이터 포인트를 찾아 이를 예측에 활용한다. (회귀의 경우)
- ▶ 새로운 데이터는 해당 거리안의 k개 데이터가 예측하는 **타겟(목표변수)의 값의 평균**으로 예측하게 된다. (회귀의 경우)
- ▶ 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 정규화**하는 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 값 대부분이 0인 데이터 셋은 잘 동작하지 않음.

03 선형모델(linear model)

- ▶ 선형 모델은 100여 년 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 사용한다.
- (2) 결정계수는 회귀 모델에서 예측의 적합도를 측정한 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 일반화하기

- ▶ 과적합을 해소하기 위해 우리는 정규화 항을 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)

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

06 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. Lasso도 계수를 얼마나 강하게 0으로 규제할지 α 매개변수를 이용.

06 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.

06 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

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

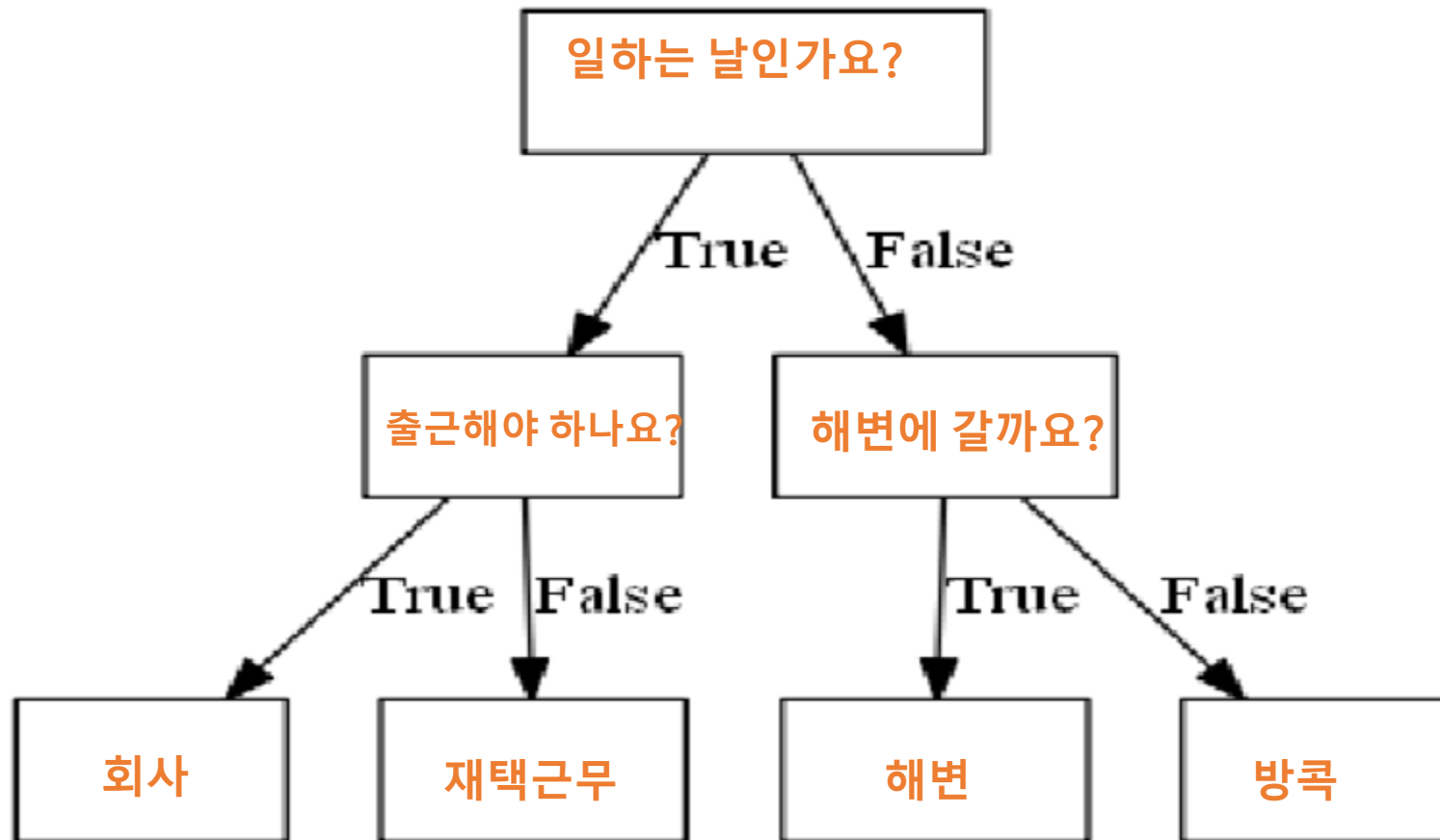
07 ElasticNet(엘라스틱넷)

▶ 엘라스틱 넷

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

08 결정트리 (decision tree)

- ▶ 결정 트리 (decision tree)는 분류와 회귀 문제에 널리 사용하는 모델

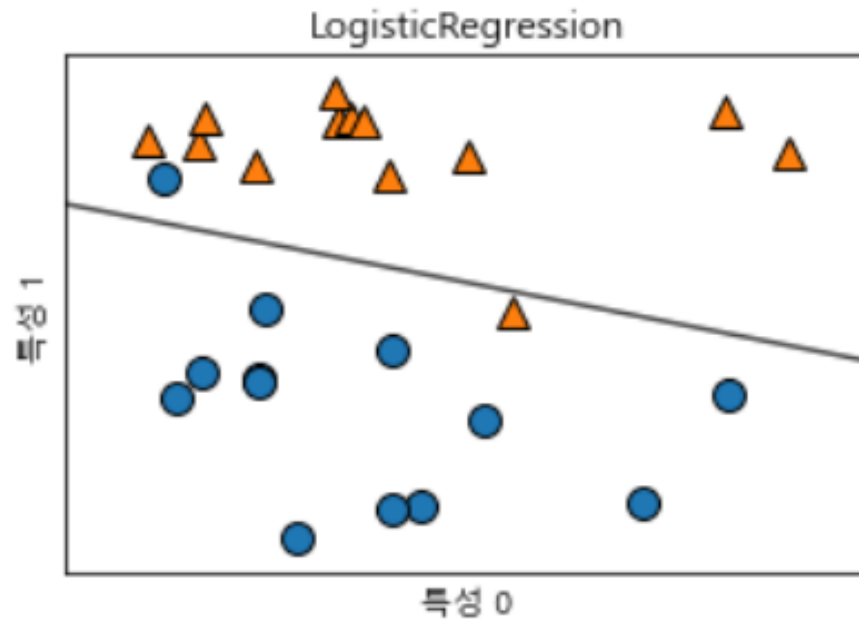
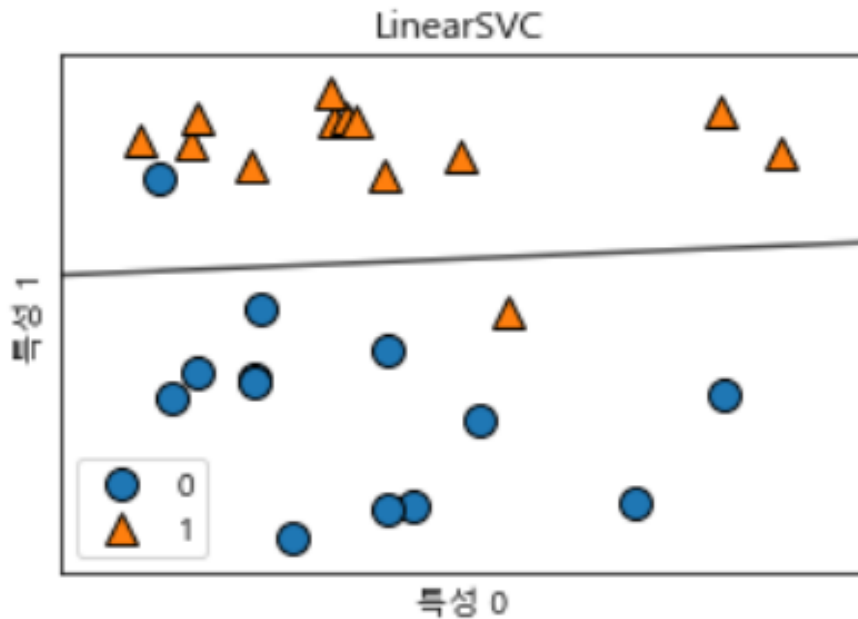


09 분류용 선형 모델

- ▶ 선형 분류기는 선, 평면, 초평면을 사용해서 두 개의 클래스를 분류하는 분류기
- ▶ 두 개의 선형 분류 알고리즘
 - (1) 로지스틱 회귀(Logistic Regression)
 - (2) 서포트 벡터 머신(support vector machine)

09 분류용 선형 모델

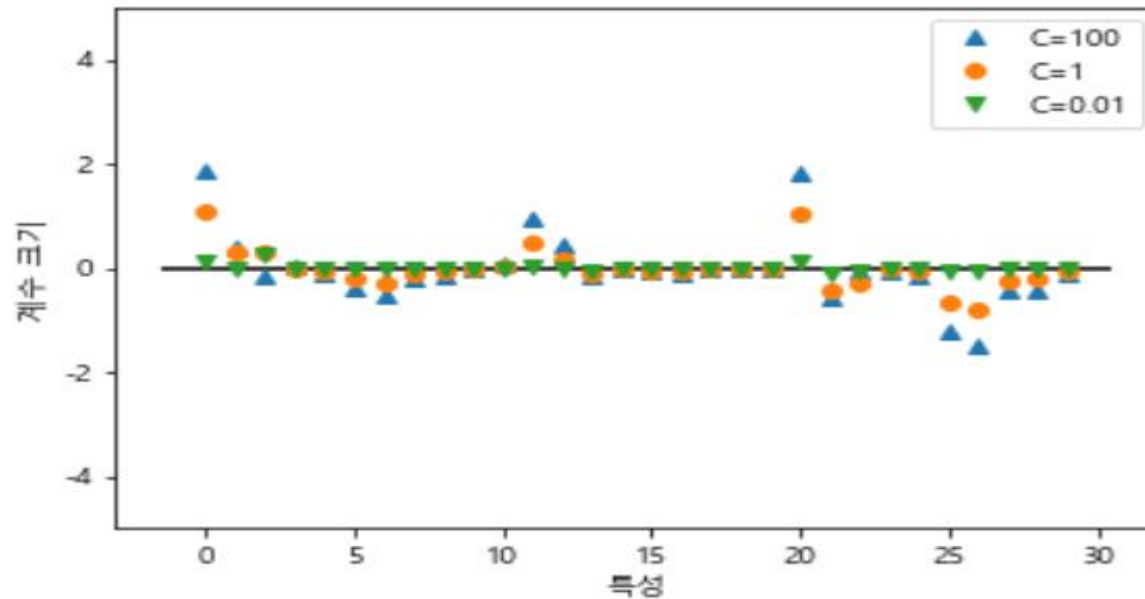
어떻게 분류되는가?



- LinearSVC와 LogisticRegression으로 만든 결정 경계가 직선으로 표현됨.
- 직선을 기준으로 위쪽은 클래스 1, 아래쪽은 클래스 0으로 나뉨.

09 분류용 선형 모델

규제를 결정하는 매개변수 C



- C의 값이 작을수록 규제가 크고, C값이 커지면 규제가 줄어든다. 반비례

09 분류용 선형 모델

다중 클래스 분류용 선형 모델

- A. 로지스틱 회귀만을 제외하고 많은 선형 분류 모델은 태생적으로 이진 분류만을 지원
- B. 이진 분류 알고리즘은 다중 클래스 분류 알고리즘으로 확장하는 보편적 기법은 **일대다 방법**이다.
- C. 일대다 방식은 각 클래스를 다른 모든 클래스와 구분하도록 이진 분류 모델을 학습
- D. **클래스의 수만큼 이진 분류 모델**이 만들어진다.
- E. 예측을 할 때, 이렇게 만들어진 모든 이진 분류기가 작동하여 가장 높은 점수를 내는 분류기의 클래스를 예측값으로 선택

09 분류용 선형 모델

장단점과 매개변수

- A. 선형 모델의 주요 매개변수는 회귀 모델은 α 이다.
- B. LinearSVC와 LogisticRegression에서는 **C가 매개변수**이다.
 - C와 α 는 로그 스케일 (0.01, 0.1, 1, 10)으로 최적치를 결정.
- C. L1규제를 사용할지, L2규제를 사용할지 결정해야함.
- D. LogisticRegression과 Ridge에 대용량 데이터 셋이라면 `solver='sag'` 옵션을 줄 수 있다. 또는 다른 대안으로 SGDClassifier와 SGDRegressor를 사용 가능.

09 분류용 선형 모델

▶ 장점

- A. 선형 모델은 학습 속도가 빠르고 예측도 빠르다.
- B. 매우 큰 데이터셋과 희소한 데이터셋에도 잘 작동한다.
- C. 샘플에 비해 특성이 많을 때 잘 동작.

▶ 단점

- A. 저 차원 데이터셋에서는 다른 모델들의 일반화 성능이 좋음.

History

날짜	내용	비고
2020/09/10	내용 업데이트	
2020/12/02	릿지, 랏소 내용 추가, 분류용 선형 모델 추가	v113
2021/07/13	내용 정리 및 폰트 및 기타 변경	v114