
02

머신러닝 필수 개념

2.1 지도학습과 비지도학습

2.1.1 지도학습

- 정답을 알려주면서 진행되는 학습.
- 학습시 데이터와 함께 레이블(정답)이 항상 제공돼야 함.

2.1.2 비지도학습

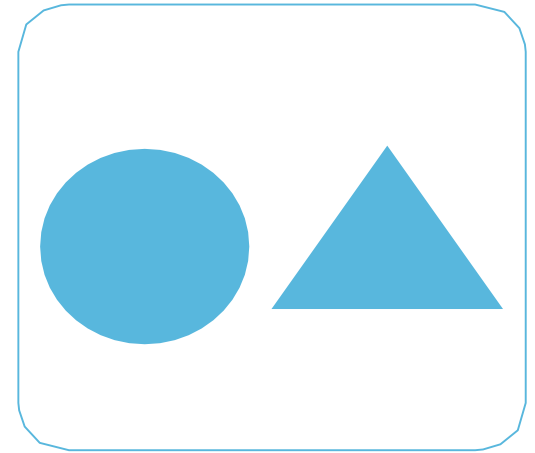
- 레이블(정답)이 없이 진행되는 학습.
- 학습할 때 레이블 없이 데이터만 필요.

2.2 분류와 회귀

2.2.1 분류

- 데이터가 입력됐을 때 지도학습을 통해 미리 학습된 레이블 중 하나 또는 여러 개의 레이블로 예측.

Ex) 이진 분류, 다중 분류



다중 레이블 분류 테스트
데이터 예제

2.2.2 회귀

- 입력된 데이터에 대해 연속된 값으로 예측.

Ex) 날씨 예측

2.3 과대적합과 과소적합

2.3.1 과소적합

- 모델 학습 시 충분한 데이터의 특징을 활용하지 못할 경우 발생.
- 테스트 데이터뿐만 아니라 학습 데이터에 대해서도 정확도가 낮게 나올 경우 과소적합된 모델일 가능성이 높음.

과소적합 학습 데이터의 예

사물	분류값	생김새
야구공	공	동그라미
농구공	공	동그라미
테니스공	공	동그라미
딸기	과일	세모
포도알	과일	동그라미

2.3 과대적합과 과소적합

2.3.2 과대적합

- 학습 데이터에 대한 정확도는 상당히 높지만 테스트 데이터 또는 학습 데이터 외의 데이터에는 정확도가 낮게 나오는 모델.

과대적합 학습 데이터의 예

사물	분류값	생김새	크기	줄무늬
야구공	공	원형	중간	있음
농구공	공	원형	큼	있음
테니스공	공	원형	중간	있음
딸기	과일	세모	중간	없음
포도알	과일	원형	작음	없음

명제: “생김새가 원형이고 크기가 작지 않으며, 줄무늬가 있으면 공이다”

2.3 과대적합과 과소적합

2.3.2 과대적합

명제: “생김새가 원형이고 크기가 작지 않으며, 줄무늬가 있으면 공이다”

테스트 데이터

사물	분류값	생김새	크기	줄무늬
골프공	공	원형	작음	없음
수박	과일	원형	큼	있음
당구공	공	원형	중간	없음
럭비공	공	타원형	큼	있음
볼링공	공	원형	큼	없음

- 과대적합은 특징이 필요 이상으로 많을 경우(분산이 높을 경우) 발생.
 - 더 많은 데이터를 확보해서 부족한 학습 데이터를 충분히 채움.
 - 특징들의 수치값을 정규화함으로써 특정 특징에 의한 편향을 줄임.
 - 딥러닝인 경우 조기 종료(early stopping) 및 드롭아웃(drop out)을 사용.

2.4 혼동행렬

- 모델의 성능을 평가할 때 사용되는 지표.

혼동행렬의 예

		예측값			
		A	B	C	D
실제값	A	9	1	0	0
	B	1	15	3	1
	C	5	0	24	1
	D	0	4	1	15

- 혼동한다는 정보를 알아낼 수 있고 그에 따른 모델 개선을 생각해볼 수 있음.
- 대략적인 모델의 성능도 눈으로 확인할 수 있음.

2.5 머신러닝 모델의 성능 평가

2.5.1 TP(true positive) – 맞는 것을 올바르게 예측한 것

- 데이터를 입력했을 때 데이터의 실제값을 올바르게 예측한 케이스

혼동행렬에서 TP 찾기

		예측값			
		A	B	C	D
실제값	A	9	1	0	0
	B	1	15	3	1
	C	5	0	24	1
	D	0	4	1	15

2.5 머신러닝 모델의 성능 평가

2.5.2 TN(true negative) – 틀린 것을 올바르게 예측한 것

- 데이터를 입력했을 때 틀린 것을 올바르게 예측한 것

혼동행렬에서 A 클래스의 TN 찾기

		예측값			
		A	B	C	D
실제값	A	9	1	0	0
	B	1	15	3	1
	C	5	0	24	1
	D	0	4	1	15

2.5 머신러닝 모델의 성능 평가

2.5.2 TN(true negative) – 틀린 것을 올바르게 예측한 것

- 데이터를 입력했을 때 틀린 것을 올바르게 예측한 것

혼동행렬에서 D 클래스의 TN 찾기

		예측값			
		A	B	C	D
실제값	A	9	1	0	0
	B	1	15	3	1
	C	5	0	24	1
	D	0	4	1	15

2.5 머신러닝 모델의 성능 평가

2.5.3 FP(false positive) – 틀린 것을 맞다고 잘못 예측한 것

- 데이터를 입력했을 때 틀린 것을 맞다고 잘못 예측한 것

혼동행렬에서 A 클래스의 FP 찾기

		예측값			
		A	B	C	D
실제값	A	9	1	0	0
	B	1	15	3	1
	C	5	0	24	1
	D	0	4	1	15

2.5 머신러닝 모델의 성능 평가

2.5.3 FP(false positive) – 틀린 것을 맞다고 잘못 예측한 것

- 데이터를 입력했을 때 틀린 것을 맞다고 잘못 예측한 것

혼동행렬에서 B 클래스의 FP 찾기

		예측값			
		A	B	C	D
실제값	A	9	1	0	0
	B	1	15	3	1
	C	5	0	24	1
	D	0	4	1	15

2.5 머신러닝 모델의 성능 평가

2.5.4 FN(false negative) – 맞는 것을 틀렸다고 잘못 예측한 것

- 데이터를 입력했을 때 맞는 것을 틀렸다고 잘못 예측한 것

혼동행렬에서 A 클래스의 FN 찾기

		예측값			
		A	B	C	D
실제값	A	9	1	0	0
	B	1	15	3	1
	C	5	0	24	1
	D	0	4	1	15

2.5 머신러닝 모델의 성능 평가

2.5.5 정확도

- 입력된 데이터에 대해 얼마나 정확하게 예측하는지를 나타내는 지표.
- 혼동 행렬 상에서는 대각선(TP)을 전체 셀로나눈 값에 해당.

혼동행렬에서 정확도 구하기

		예측값			
		A	B	C	D
실제값	A	9	1	0	0
	B	1	15	3	1
	C	5	0	24	1
	D	0	4	1	15

$$\text{정확도} = 9 + 15 + 24 + 15 / 80 = 0.78$$

2.5 머신러닝 모델의 성능 평가

2.5.6 정밀도

- 모델의 예측값이 얼마나 정확하게 예측됐는가를 나타내는 지표.

$$\text{정밀도} = \text{TP} / (\text{TP} + \text{FP})$$

암 예측 모델 A의 혼동행렬

	암환자	일반환자
암환자	9	1
일반환자	30	60

모델 A의 암환자 정밀도
 $= 9/39 = 23\%$,

암 예측 모델 B의 혼동행렬

	암환자	일반환자
암환자	1	9
일반환자	20	70

모델 B의 암환자 정밀도
 $= 1/21 = 4.7\%$,

2.5 머신러닝 모델의 성능 평가

2.5.7 재현율

- 실제값 중에서 모델이 검출한 실제값의 비율을 나타내는 지표.

$$\text{재현율} = \text{TP} / (\text{TP} + \text{FN})$$

암 예측 모델 A의 혼동행렬

	암환자	일반환자
암환자	9	1
일반환자	30	60

모델 A의 암환자 재현율
= $9/10 = 90\%$,

암 예측 모델 B의 혼동행렬

	암환자	일반환자
암환자	1	9
일반환자	20	70

모델 B의 암환자 재현율
= $1/10 = 10\%$,

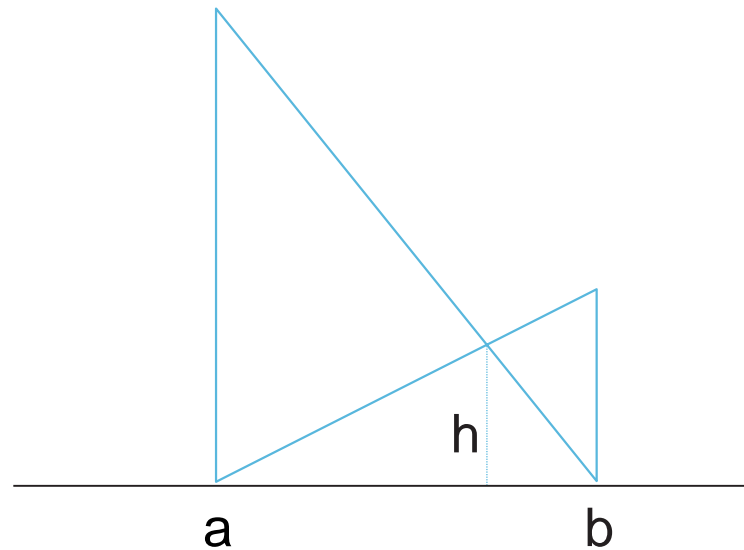
2.5 머신러닝 모델의 성능 평가

2.5.7 F1점수

- 정밀도와 재현율 두 값을 조화평균 내서 하나의 수치로 나타낸 지표.

$$\text{조화평균} = 2 * a * b / (a + b)$$

$$\text{F1 점수} = 2 * \text{재현율} * \text{정밀도} / (\text{재현율} + \text{정밀도})$$



a 와 b 의 조화 평균 시각화($h = \text{조화평균}/2$)

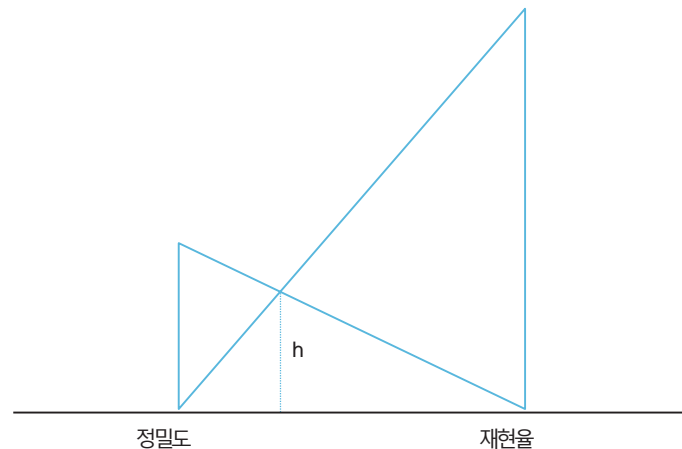
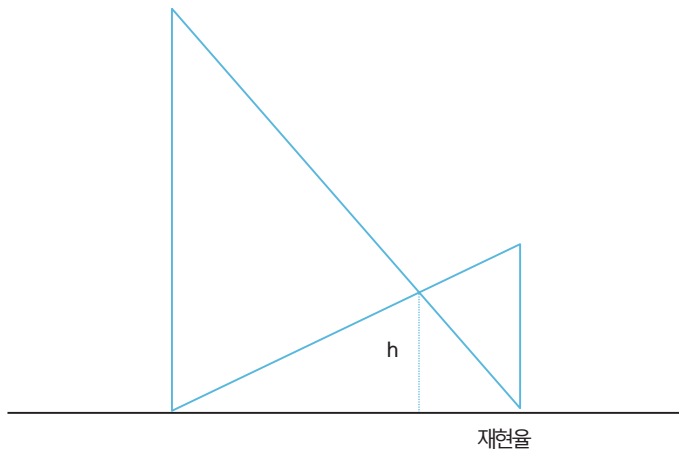
2.5 머신러닝 모델의 성능 평가

2.5.7 F1점수

- 정밀도와 재현율 두 값을 조화평균 내서 하나의 수치로 나타낸 지표.

$$\text{조화평균} = 2 * a * b / (a + b)$$

$$\text{F1 점수} = 2 * \text{재현율} * \text{정밀도} / (\text{재현율} + \text{정밀도})$$



F1 점수의 작동 원리

2.5 머신러닝 모델의 성능 평가

2.5.7 F1점수

- 하지만, 데이터의 레이블이 불균일하게 분포돼 있을 경우 정확도는 왜곡된 성능 평가로 이어질 수 있음.

혼동행렬에서 정확도 구하기

		예측값			
		A	B	C	D
실제값	A	9	1	0	0
	B	1	15	3	1
	C	5	0	24	1
	D	0	4	1	15

$$\text{정확도} = (9 + 8 + 7 + 9) / 40 = 82.5\%$$

2.5 머신러닝 모델의 성능 평가

2.5.7 F1점수

- 아래의 모델 1은 B, C, D 레이블에 대해 하나도 예측을 제대로 하지 못했음에도 A 레이블을 가진 데이터가 너무 많아서 정확도가 96.6%로 나타나게 되어 성능 평가가 상당히 높아지는 문제를 보임.

모델1의 정확도 구하기

		예측값			
		A	B	C	D
실제값	A	955	5	0	0
	B	8	0	1	1
	C	10	0	0	1
	D	0	1	9	0

$$\text{정확도} = (995 + 0 + 0 + 0) / 1030 = 96.6\%$$

2.5 머신러닝 모델의 성능 평가

2.5.7 F1점수

- 아래의 모델 2는 A에 대한 예측율은 떨어지지만 보편적으로 예측을 상당히 잘하는 모델임에도 정확도는 모델1보다 낮게 평가됨.

모델2의 정확도 구하기

		예측값			
		A	B	C	D
실제값	A	700	100	100	100
	B	0	9	1	0
	C	0	0	9	1
	D	0	1	0	9

$$\text{정확도} = (700 + 9 + 9 + 9) / 1030 = 70.5\%$$

2.5 머신러닝 모델의 성능 평가

2.5.7 F1점수

- 레이블이 데이터 상에서 불균일하게 분포된 경우 F1 점수를 사용하면 정확도보다 나은 성능 평가 비교가 가능.

2.5 머신러닝 모델의 성능 평가

2.5.7 F1점수

모델1의 F1 점수 계산

	A	B	C	D	재현율
A	955	5	0	0	0.99
B	8	0	1	1	0
C	10	0	0	1	0
D	0	1	9	0	0
정밀도	0.98	0	0	0	

$$\text{평균 정밀도} = (0.98 + 0 + 0 + 0) / 4 = 0.245$$

$$\text{평균 재현율} = (0.99 + 0 + 0 + 0) / 4 = 0.2475$$

$$\text{F1 점수} = 2 * 0.245 * 0.2475 / (0.245 + 0.2475)$$

$$= 0.121275 / 0.4925$$

$$= 0.246$$

2.5 머신러닝 모델의 성능 평가

2.5.7 F1점수

모델2의 F1 점수 계산

	A	B	C	D	재현율
A	700	100	100	100	0.7
B	0	9	1	0	0.9
C	0	0	9	1	0.9
D	0	1	0	9	0.9
정밀도	1	0.08	0.08	0.08	

$$\text{평균 정밀도} = (1 + 0.08 + 0.08 + 0.08) / 4 = 0.31$$

$$\text{평균 재현율} = (0.7 + 0.9 + 0.9 + 0.9) / 4 = 0.85$$

$$\text{F1 점수} = 2 * 0.31 * 0.85 / (0.31 + 0.85)$$

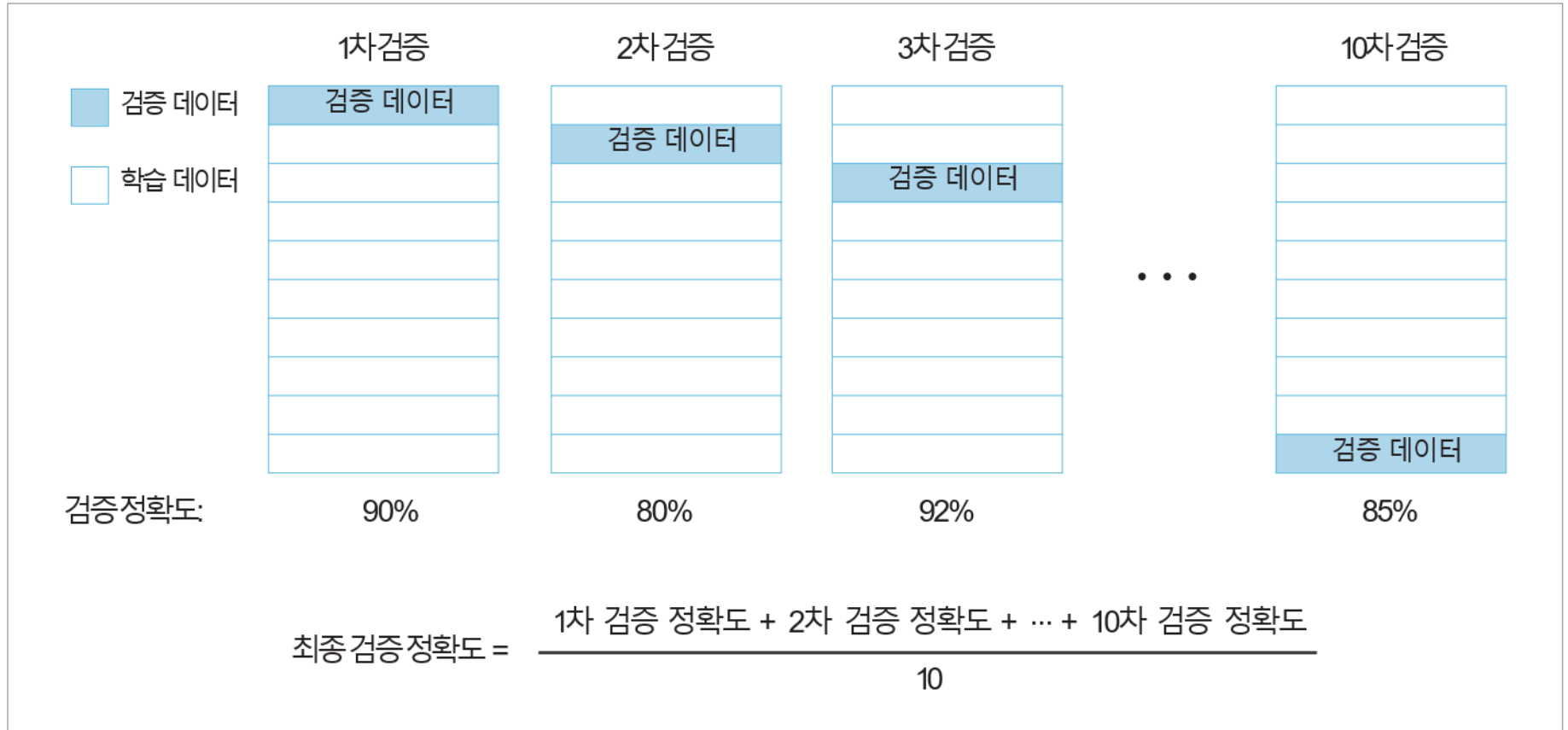
$$= 0.527 / 1.16$$

$$= 0.454$$

2.5 k-폴드 교차 검증

- 학습 데이터의 일정 부분을 검증데이터로 쓰되, n 번의 검증 과정을 통해 학습 데이터의 모든 데이터를 한 번씩 검증데이터로 사용해서 n 개의 검증 결과를 평균낸 값을 검증 성능 평가 지표로 사용하는 방식.
- 장점
 1. 검증 결과가 일정 데이터에 치우치지 않고 모든 데이터에 대한 결과이므로 신빙성이 높음.
 2. 별도로 검증 데이터를 분리하지 않아도 됨.

2.5 k-폴드 교차 검증



k-폴드 검증의 작동 원리