

모형의 성능 지표

🕒 생성일	@2022년 8월 15일 오전 12:44
▼ 유형	머신러닝/딥러닝
👤 작성자	

reference :

‘언제 MSE, MAE, RMSE’ 를 사용하는가?

모형의 성능지표(MSE, MAPE, 정확도, 정밀도, 재현율, 특이도, F1 measure, ROC curve).

MSE, MAE, RMSE

Epoch	Prediction	Target
1	[0, 4, 9]	[3, 5, 7]
2	[2, 4, 2]	[3, 5, 7]
3	[3, 5, 6]	[3, 5, 7]

$$MSE = \frac{1}{N} \sum_i^N (pred_i - target_i)^2$$

$$RMSE = \sqrt{\frac{1}{N} \sum_i^N (pred_i - target_i)^2}$$

$$MAE = \frac{1}{N} \sum_i^N |(pred_i - target_i)|$$

$MSE = \frac{1}{N} \sum_i (pred_i - target_i)^2$	$RMSE = \sqrt{\frac{1}{N} \sum_i (pred_i - target_i)^2}$	$MAE = \frac{1}{N} \sum_i (pred_i - target_i) $
$E1 : \frac{1}{3}(9 + 1 + 4) = \frac{14}{3}$	$E1 : \frac{\sqrt{42}}{3}$	$E1 : 2$
$E2 : \frac{1}{3}(27) = 9$	$E2 : 3$	$E2 : \frac{7}{3}$
$E3 : \frac{1}{3}$	$E3 : \frac{\sqrt{3}}{3}$	$E3 : \frac{1}{3}$

파이썬 코드에서의 구현은 다음과 같다.

```
import numpy as np

def MSE(pred, target, epochs=len(pred)):
    losses=[]
    for i in range(epochs):
        losses.append(np.sum((pred[i]-target[i])**2)/len(pred))
    return losses

def RMSE(pred, target, epochs=len(pred)):
    losses=[]
    for i in range(epochs):
        losses.append(np.sqrt(np.sum((pred[i]-target[i])**2)/len(pred)))
    return losses

def MAE(pred, target, epochs=len(pred)):
    losses=[]
    for i in range(epochs):
        losses.append(np.sum(np.abs(pred[i]-target[i]))/len(pred))
    return losses
```

MSE

- Mean Square Error 는 예측값과 정답의 차이를 제공하기 때문에 이상치에 대해 민감하다.
- 오차값에 제곱을 취하기 때문에 오차가 0-1 사이인 경우에는 본래보다 더 작게 반영되고, 1보다 클 때는 본래보다 더 크게 반영된다.

- 모든 함수값이 미분 가능하다.

RMSE

- MSE 에서 루트를 취하기 때문에 오차 값의 반영이 극적이라는, MSE 의 단점을 완화한다.
- RMSE 는 MSE 에서 루트를 취하기에 미분 불가능한 지점을 갖는다.
- RMSE 는 이상치에 대한 민감도가 MSE와 MAE 사이에 있기 때문에 이상치를 적절히 잘 다룬다고 간주되는 경향이 있다.

MAE

- Mean Absolute Error 는 이상치에 둔감 혹은 강건 robust 하다. 오차값이 outlier 의 영향을 상대적으로 크게 받지 않는다.
- 함수에 미분 불가능한 지점이 있다.
- 모든 오차에 동일한 가중치를 부여한다.

언제 어떤 loss를 사용하는가

- 약간의 이상치가 있는 경우, 그 이상치의 영향을 적게 받으면서 모델을 만들고자 할 때, MAE를 쓰는 것이 적절하다. 이상치에 대한 MAE 의 강건함은 이상치를 포함한 훈련 데이터에 적합하게 학습되는 오버 피팅을 방지하는 데 도움이 될 수 있다.
- 이상치까지 고려하여 모델 일반화가 이루어진다면, RMSE를 채택하는 것이 적절하다.

MAPE

- Mean Absolute Percentage error 는 퍼센트 값을 가지며 0에 가까울 수록 회귀 모형의 성능이 좋다고 해석할 수 있다.
- 0~100 % 사이의 값을 가져 이해하기 쉬우므로 성능 비교 해석이 가능하다.

분류분석에서 사용되는 성능지표는 다음과 같다.

클래스 ={정상, 불량}		예측한 클래스	
		정상	불량
실제 제품	정상	TN	FP
	불량	FN	TP

정확도

- 전체 데이터 중에서 모형으로 판단한 값이 실제 값과 부합하는 비율
- $(\text{옳게 분류된 데이터의 수}) / (\text{전체 데이터의 수})$

정밀도

- 분류 모형이 불량을 진단하기 위해 얼마나 잘 작동했는지 보여주는 지표
- $TP / (FP + TP)$

재현율

- 감도와 같은 지표로써 불량 데이터 중 실제로 불량이라고 진단한 제품의 비율
- $TP / (FN + TP)$

실제 데이터의 특성상 정확도보다는 G-mean 지표나 불량에 관여하는 지표인 정밀도 및 재현율만 고려하는 F1 점수가 더 고려해볼 수 있는

지표이다.

$$G - mean = \sqrt{specificity \cdot recall} = \sqrt{(1 - \alpha) \cdot (1 - \beta)}$$

$$F_1 \text{ measure} = \frac{2}{\frac{1}{precision} + \frac{1}{recall}} = 2 \cdot \frac{precision \cdot recall}{precision + recall}$$

F1 점수는 정밀도와 재현율의 조화평균이다.

ROC Curve, AUC

ROC curve, AUC

- 가로축을 1-특이도(specifivity) 세로축을 재현율(recall)로 하여 시각화한 그래프를 ROC (Receiver Operating Characteristics) Curve라고함.
- 이때 ROC curve의 면적을 AUC라고함.
- AUC가 1에 가까울 수록 좋은 지표다. (0 , 1)일때 가장 좋음.



