

classification_report

classification_report

- 해당 함수는 주요 분류 메트릭의 결과를 리턴해주는 함수

```
from sklearn.metrics import classification_report
y_true = [0, 1, 2, 2, 0]
y_pred = [0, 0, 2, 1, 0]
target_names = ['class 0', 'class 1', 'class 2']
print(classification_report(y_true, y_pred, target_names=target_names))
```

- 결과 해석

	precision	recall	f1-score	support
class 0	0.67	1.00	0.80	2
class 1	0.00	0.00	0.00	1
class 2	1.00	0.50	0.67	2
accuracy			0.60	5
macro avg	0.56	0.50	0.49	5
weighted avg	0.67	0.60	0.59	5

- macro avg와 weighted avg가 서로 다른 값을 갖는 이유는 각 클래스의 데이터 수가 다르기 때문임
- weighted avg는 클래스의 데이터 수를 고려하여 평균을 구하기 때문에, 클래스 1의 데이터 수가 클래스 0보다 많으므로 weighted avg에서는 클래스 1의 평균값이 가장 큰 비중을 차지함
 - 클래스의 개수에 비례하여 가중치를 적용하여 평균을 계산
- macro avg는 각 클래스별로 동일한 비중을 둔 평균을 구하기 때문에, 클래스별 데이터 수의 영향을 받지 않음
 - 일반적인 평균이라고 생각하면 됨
- **macro avg**
 - 레이블별 가중치를 포함하지 않고 값을 평균함
 - precision 예) $(0.67 + 0 + 1.0) / 3 = 0.56$
- **weighted avg**

- 레이블별 가중치 값을 활용하여 평균을 계산
 - precision 예) $((0.67) \times 2 + (0.0) \times 1 + (1.0) \times 2) / 5 = 0.67$
- 레이블의 중요도에 따라 어떤 지표를 선택할지 달라짐
- 모든 레이블이 유사한 중요도를 가진다면 macro average 값을 참고
- 샘플이 많은 레이블에 중요도를 집중해서 보고 싶다면 weighted average 값을 참고하면 됨
- 클래스별 수가 다른 경우에는 weighted avg가 더 의미있는 평가 지표가 될 수 있음