# Training (1)

## Data Splitting

### Training

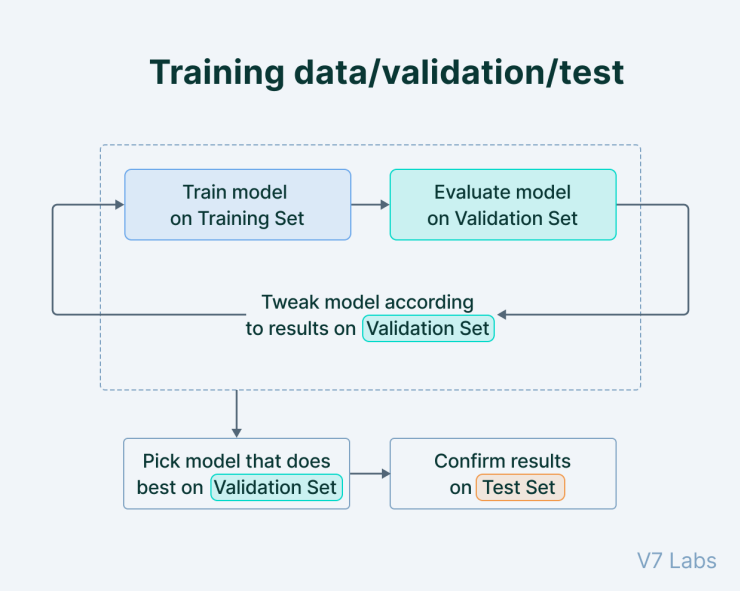
: 모델을 학습시킬 때 쓰이는 데이터

### Test

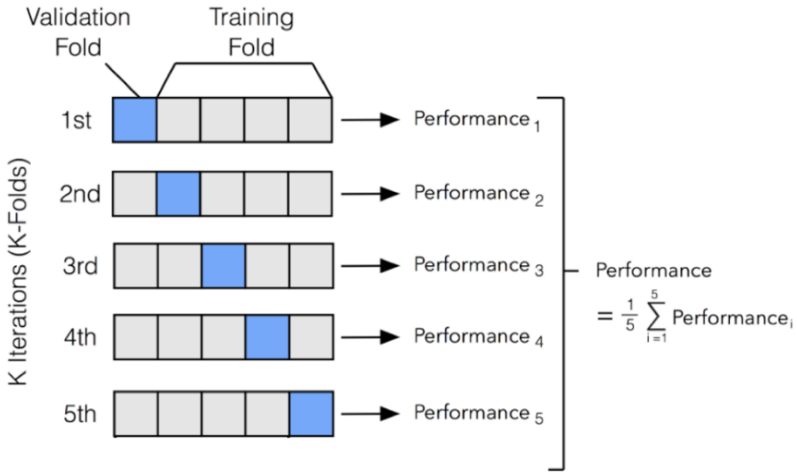
: 학습된 모델의 성능 측정하기 위해 사용되는 데이터

### Validation

: 학습 과정에서 모델에 대한 성능 평가를 미리 하기 위해 사용하는 데이터.




### K-fold Cross Validation

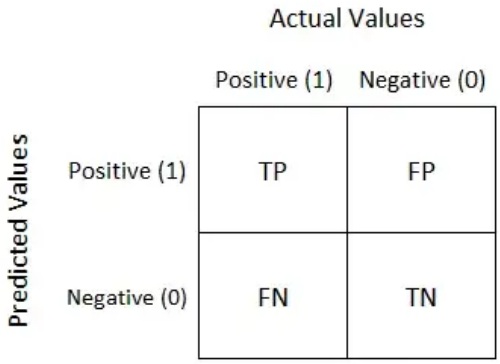


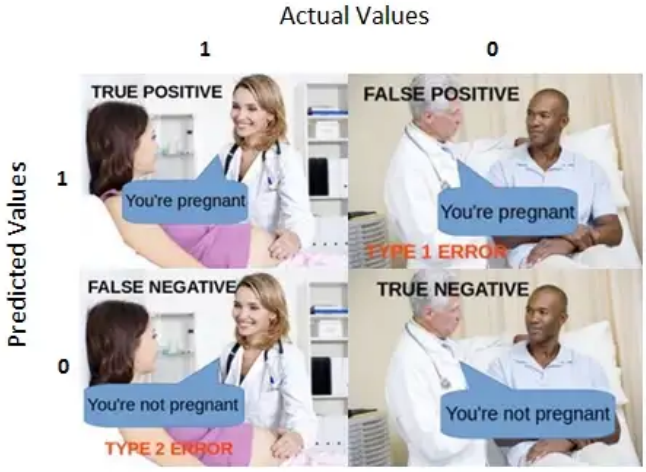

**Cross validation (교차 검증)**: 데이터를 학습용/평가용 데이터 세트로 여러 번 나눈 것의 평균적인 성능을 계산하여 일반화된 성능을 얻는 방법

**K-fold cross validation**: 데이터를 k개로 분할한 뒤, k-1개를 학습용 데이터 세트로, 1개를 평가용 데이터 세트로 활용. k번 반복하여 k개의 성능 지표를 얻는다

## Confusion Matrix

: 단일 및 다중 분류 모델의 성능을 측정하는 지표




**TP (True Positive)**: positive로 예측, 실제로 negative

**TN (True Negative)**: negative로 예측, 실제로 negative

**FP (False Positive)** ⇒ **Type 1 Error**: positive로 예측, 실제로 negative

**FN (False Negative)** ⇒ **Type 2 Error**: negative로 예측, 실제로 positive

→ 즉 TP와 TN이 높을 수록, FP와 FN이 낮을수록 좋은 것

### Accuracy (정확도)

: 전체 중 모델이 바르게 분류한 비율.

$$Accuracy = \frac{TP + TN}{TP + FN + FP + TN}$$

- class imbalance가 있다면 정확도는 좋은 성능지표가 아님. 예를들어 웹사이트 방문자 중 99%는 구매를 하지 않고 1%만 구매를 한다고 했을 때 분류모델이 “방문자 중 100%가 구매를 한다” 라고 예측하면 1%만 잘못 맞춘거니까 정확도는 99%가 된다
- imbalanced data에 대한 성능지표 개선: F1 score

### Precision (정밀도)

: 모델이 positive라 분류한 것 중 실제값이 positive인 비율.

$$Precision = \frac{TP}{TP + FP}$$

### Recall (재현도)

: 실제값이 positive인 것 중 모델이 positive라고 올바르게 분류한 비율.

$$Recall = \frac{TP}{TP + FN}$$

한 가게에서 실수로 팔아버린 불량품을 모두 회수해야 할 때, 불량품이 아닌 제품이 껴있어도 상관 없음.

### Precision-Recall Trade-Off

Precision과 Recall을 모두 갖추면 좋겠지만, 불가능하기 때문에 precision을 낮추는 대신 recall을 높이거나 그 반대의 경우로 모델 조정하는 것

### F1 score

: Precision과 Recall의 조화평균. Precision이 낮고 Recall이 높거나 그 반대인 경우의 모델들을 비교하기 어렵기 때문에 이용하는 지표. 불균형한 데이터에 적합함

- $a$와 $b$의 조화평균 = $\frac{2ab}{a+b}$. 평균적인 변화율을 구할 때 사용됨

$$\frac{2 * Precision * Recall}{Precision + Recall}$$

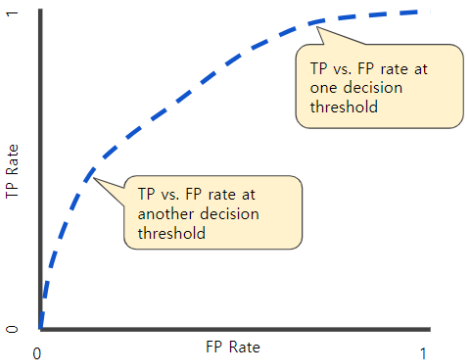
- 조화평균을 썼기 때문에 Precision과 Recall에 같은 가중치가 가해짐.
- 즉 f1 score이
  - 높으면 precision과 recall 둘 다 높다는 뜻
  - 낮으면 precision과 recall 둘 다 낮다는 뜻
  - 중간이면 precision이 높고 recall이 낮고 vice versa
- Grid search같은 자동화 작업시 유용함

### ROC (Receiver Operating Characteristic) Curve

: True Positive Rate (=recall)와 False Positive Rate (전체 negative 중 positive로 잘못 분류한 비율) 를 그려 분류모델의 성능을 보여주는 그래프

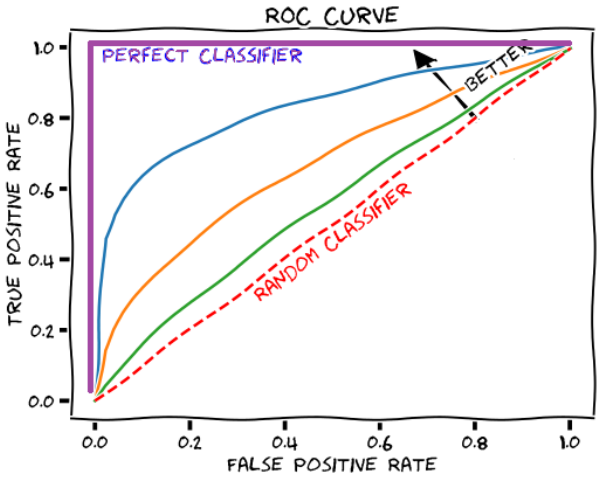
$$TPR = \frac{TP}{TP + FN}$$

$$FPR = \frac{FP}{FP + TN}$$



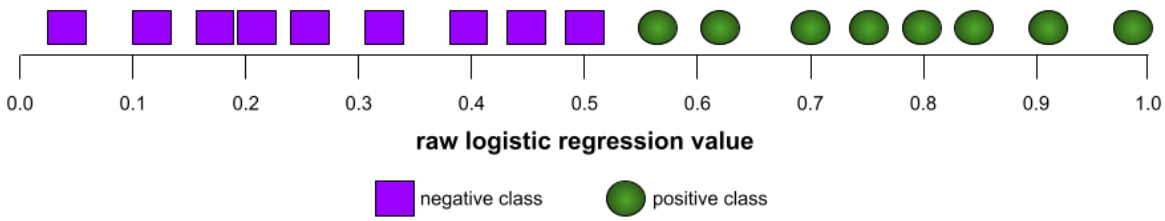
- 다양한 임계값에서 분류기의 성능을 보여줌

### AUC (Area Under the ROC Curve)

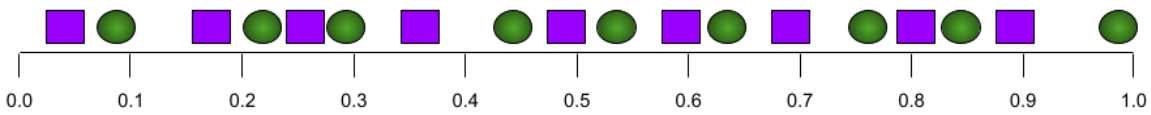



- **모든 임계값**에서의 총 성능을 측정함. 곡선 아래의 면적이 1에 가까울수록 (곡선이 왼쪽으로 치우칠수록) 모델이 정확히 분류한다는 뜻
- imbalanced 데이터에서 정확도보다 훨씬 좋은 지표
- 모델이 negative 샘플보다 positive 샘플을 우선순위로 매길 확률이라고도 해석할 수 있음. 즉 아래 그림에서 임의의 녹색 샘플이 임의의 보라색 샘플의 오른쪽에 위치할 확률

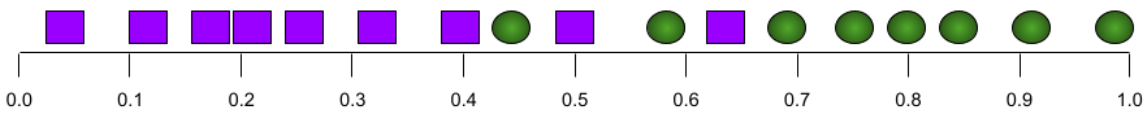
1) “perfect model”




2) AUC가 0.5인 모델




3) AUC가 0.5~1인 모델




참고

https://www.v7labs.com/blog/train-validation-test-set

https://scikit-learn.org/stable/modules/cross_validation.html

https://towardsdatascience.com/understanding-confusion-matrix-a9ad42dcfd62

https://neosla.tistory.com/18

https://towardsdatascience.com/the-f1-score-bec2bbc38aa6

https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc?hl=en

https://developers.google.com/machine-learning/glossary?hl=en#AUC