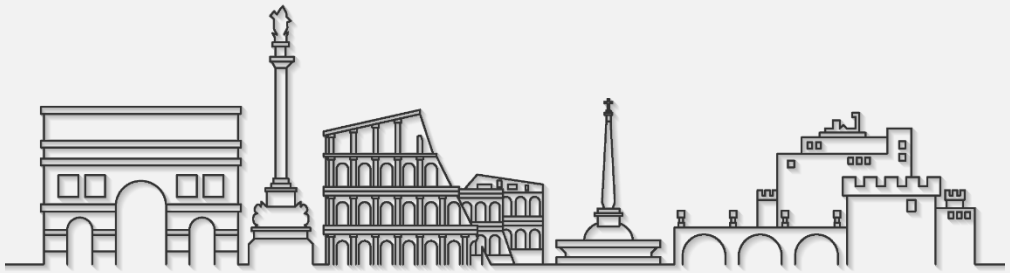


Model Evaluation

김성산. 이승연. 이승한

Contents

01.모형 평가란?

분류 분석모형의 평가 (Classification Model)

"예측 .분류를 위해 구축된 모형"이 "임의의 모형"보다

더 **우수한 분류 성과**를 보이는지?

그렇다면… 모형 평가를 해야 하는 이유는?

(1) test accuracy를 가늠해볼 수 있다!
Machine learning의 목적은 결국 unseen data에 대해 좋은 성능을 내는 것!
-> 모델을 만든 후, unseen data에 대해 잘 작동할지 반드시 확인해야!
(하지만 training data를 사용해 성능을 평가하면 안되기 때문에 따로 validation set을 만들어 정확도를 측정하는 것)

(2) 모델을 튜닝하여 모델의 성능을 높일 수 있다!
ex) 과적합을 막을 수 있음!
training accuracy는 높은데 validation accuracy는 낮다면 데이터가 training set 과적합이 일어났을 가능성 O!
-> training accuracy를 희생하더라도, validation & training accuracy를 비슷하게 맞춰줄 필요가 있음!

모형 성능을 평가하는 이유!

(Accuracy : 옳은 예측 비율)

모형의 성능을 단순히 정확도 (Accuracy)만으로 평가해도 괜찮은가?

NO!

다양한 지표들 배우기 전에….

02.모형을 평가하는 다양한 지표

confusion matrix		predicted class	
		Positive	Negative
actual class	Positive	TP	FN
	Negative	FP	TN

쉽게 외우는 법!

(1) ____ (2) _____

(1) T : 예측 성공!
F : 예측 실패!

(2) P : P라고 “예측”
N : N라고 “예측”

Confusion Matrix

다양한 지표들	계산식	의미
Precision	TP / (TP+FP)	Y 예측 중 실제 Y 비율
Accuracy	TP+TN / (TP+FP+TN+FN)	옳은 예측 비율
Recall (Sensitivity)	TP / (TP+FN)	실제 Y 중 Y 예측 비율
Specificity	TN / (FP+TN)	실제 N 중 N 예측 비율
FP Rate	FP / (FP+TN)	Y가 아닌데 Y 예측 비율 = (1-Specificity)
F Measure	Precision과 Recall의 조화 평균 (= 2*P*R / (P+R))	
Kappa	두 평가자의 평가가 얼마나 일치하는지 평가하는 값 (0~1 값)	

Ex) 문서를 “중요”와 “비중요”문서로 구분해야!
90개의 비중요 & 10개의 중요 문서가 있음.
나의 model은 90%의 accuracy(정확도)를 가짐

Case 1. 90%의 정확도

	중요 (10개)	비중요 (90개)
‘중요’로 분류	10	10
‘비중요’로 분류	0	80

불필요하기 ‘비중요’를 ‘중요’로 분류하긴 했지만,
그래도 ‘중요’ 10개 빠뜨린 건 없음! 문제 X

Case 2. 90%의 정확도

	중요 (10개)	비중요 (90개)
‘중요’로 분류	0	0
‘비중요’로 분류	10	90

모든걸 ‘비중요’로 분류! 문제 O

Accuracy 말고 어떤 지표가 좋을까요?

실제로 Y인걸 (‘중요’ 문서)인 것을 Y(‘중요’ 문서)라고 예측하는 정도가 중요!

Case 1.

Accuracy : 90%

Recall : 50%

Case 2.

Accuracy : 90%

Recall : 0%

이럴 때 필요한 것이 “RECALL”

Ex) 30명의 피의자가 있음. 이중 20명은 유죄, 10명은 무죄!
각각 70%, 80% 의 정확도로 유/무죄를 판별할 수 있는 model 2개

Case 1. 70%의 정확도

	유죄 (20명)	무죄 (10명)
'유죄'로 판결	13	2
'무죄'로 판결	7	8

무고한 10명 중 2명을 "유죄"로 오판함!

Case 2. 80%의 정확도

	유죄 (20명)	무죄 (10명)
'유죄'로 판결	20	6
'무죄'로 판결	0	4

무고한 10명 중 6명을 "유죄"로 오판함!

Case 2에 적용된 모델이 더 좋다고 할 수 있나…?

실제로 N인걸 (‘무죄’)인 사람을 N(‘무죄’)라고 판결하는 정도가 중요!

Case 1.

Accuracy : 70%

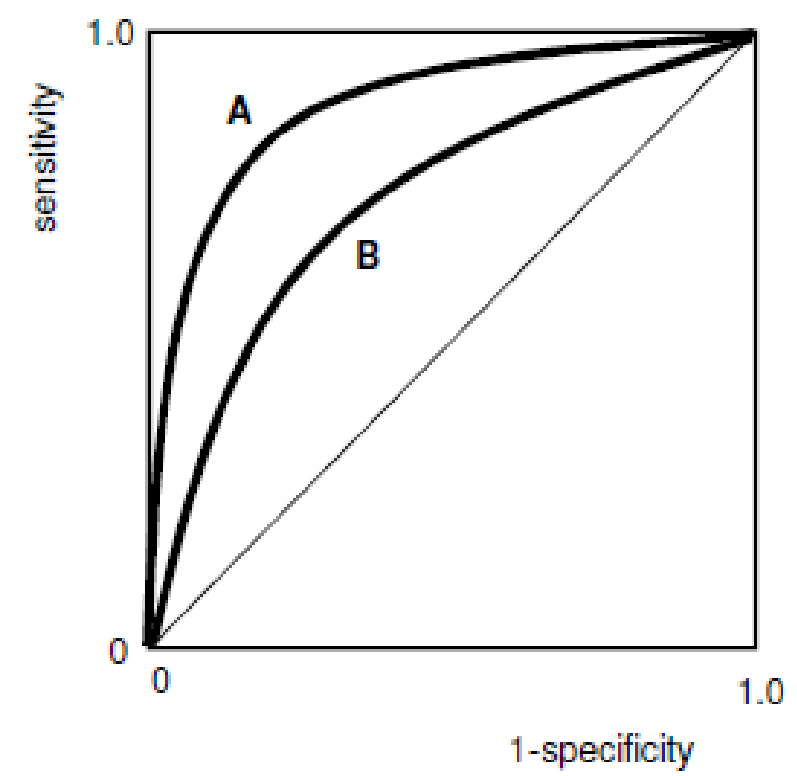
Specificity : 80%

Case 2.

Accuracy : 80%

Recall : 40%

이럴 때 필요한 것이 “Specificity”



for 두 개의 모델을 비교 분석!

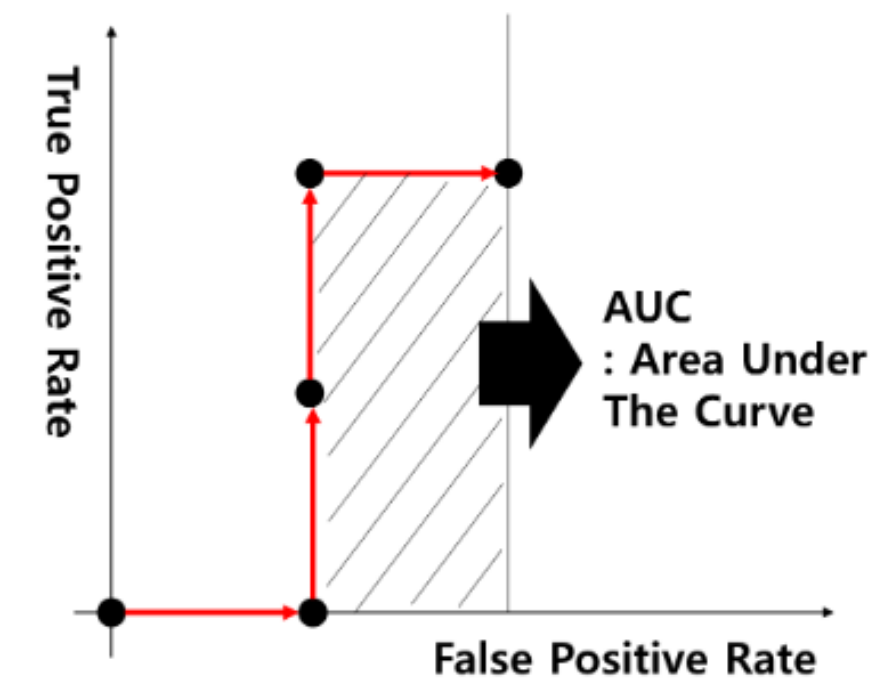
(x축) **FP Ratio** (1-Specificity)

(y축) **Recall(Sensitivity)**

ROC 그래프의 **밑부분 면적이 넓을수록 GOOD**

(x값은 작을수록, y값은 클수록 good이니까!)

ROC 그래프



0.5 ~ 1 사이의 값

< 등급 >
A (0.9~1.0)
B (0.8~0.9)
C (0.7~0.8)
D (0.6~0.7)
E (0.5~0.6)

AUC (Area Under the (ROC) Curve)

03. 다양한 모델 평가 방법

대표적 3가지 모델 평가 방법

1. Holdout 방법

2. Cross-Validation
(교차 검증법)

3. Bootstrap

- data를 두 분류로 (train + test)
- ex) 70% : 30%
- test data는 분류분석 모형에는 “영향을 미치지 않고”,
  “성과 측정”을 위해서만 사용

1. Holdout 방법

중요해서 뒤에서 자세히…

- 반복적으로 성과 측정 (그렇게 해서 나온 여러 결과들을 “평균”냄)

- ex) k-fold 교차 검증

(k개의 집합으로 나누고, k번째의 집합은 테스트/검증용, 나머지는 훈련용)

- 일반적으로 10-fold 교차 검증 사용

2. Cross-Validation (교차 검증법)

- 반복한다는 점에서 'Cross-Validation'과 유사

- 차이점 : "훈련용 자료"를 반복 재선정 (복원 추출법)

3. Bootstrap

04.Cross Validation (교차검증)

Cross validation 설명 전에…

Validation Dataset?

Train, test 2 종류의 데이터 말고도....

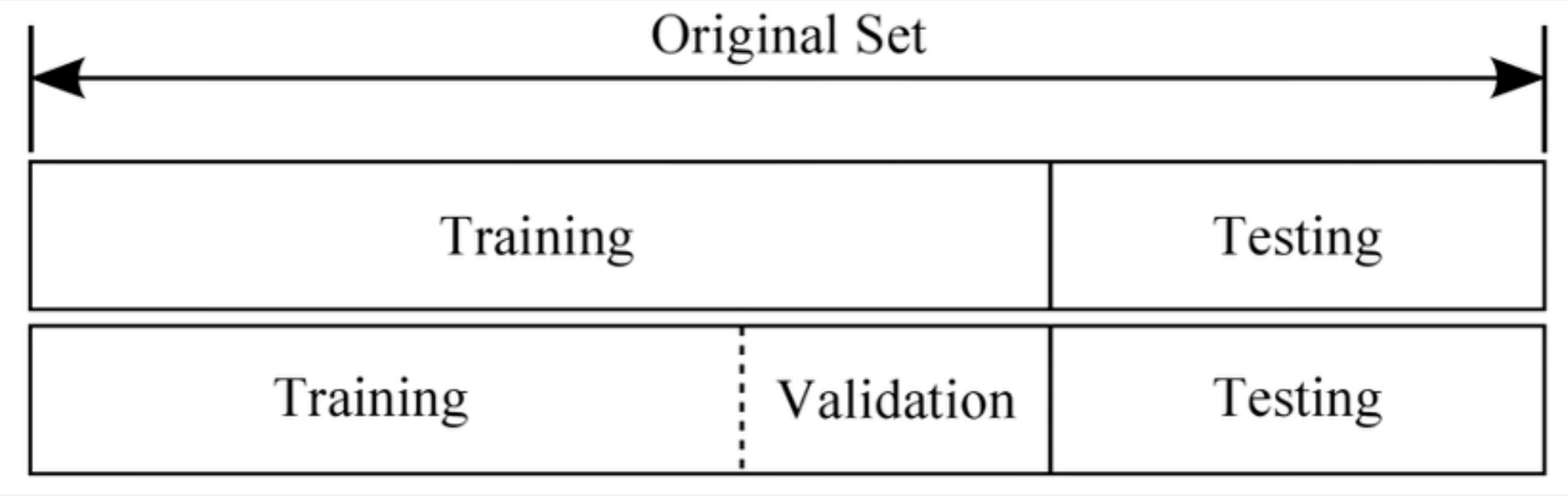
Validation Dataset (검증 데이터)

- “일반화”의 성능을 측정하기 위해!

- Train 데이터로 생성한 모델을,
  Test 데이터로 평가하기 전에,
  Validation 데이터를 통해 먼저 다듬기!

- (전) Train + Test
  (후) Train + Validation + Test

training set으로 training을 하고 test만 하면 되지, 왜 validation set을 나누는 것일까?



“모델의 성능을 평가하기 위해!”

training을 한 후에 만들어진 모형이 잘 예측
하는지 그 성능을 평가하기 위해서 사용!

(대신 training data의 일부를 희생)

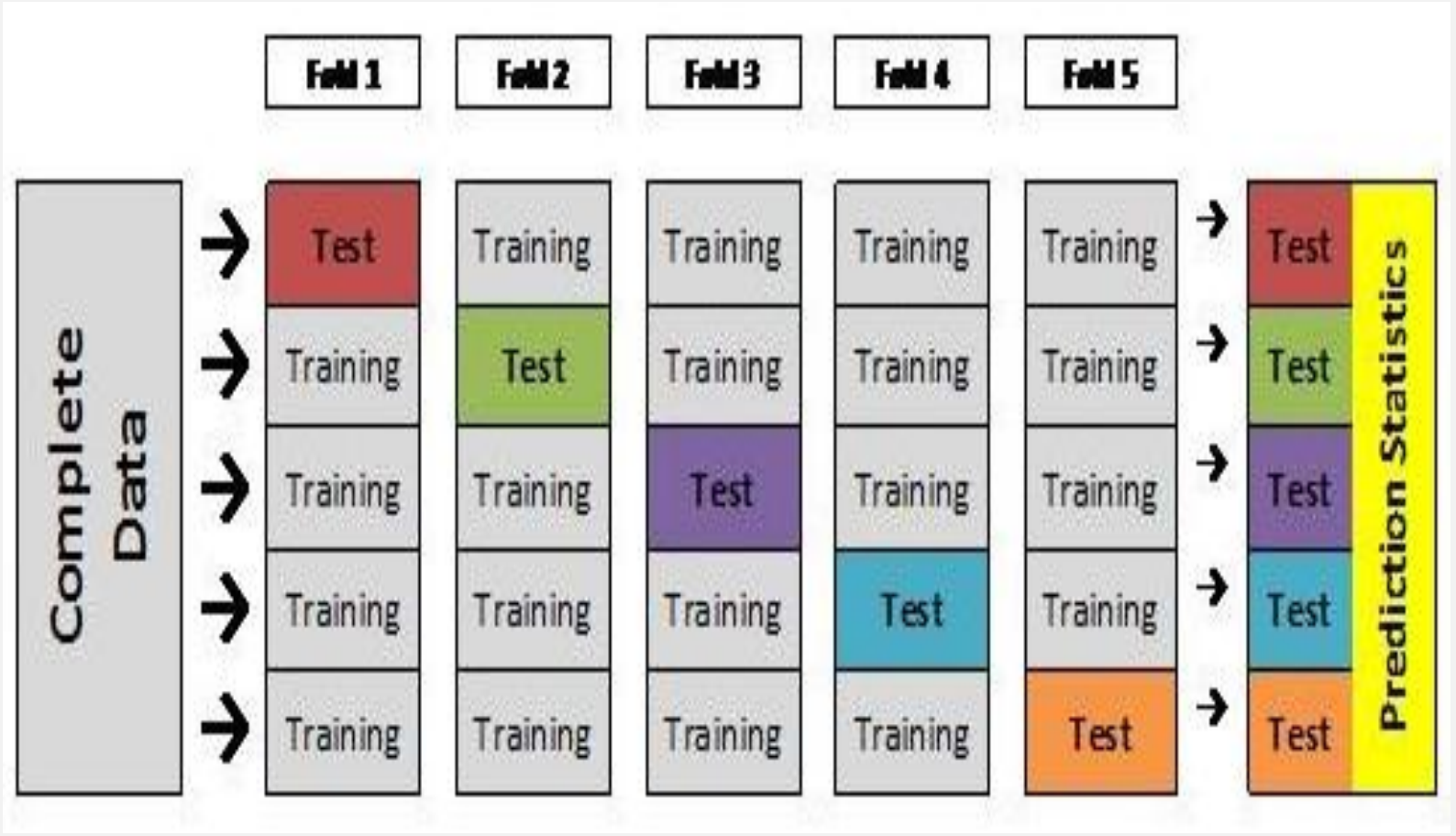
< 요약 >

일반적으로 검증용 데이터가 별도로 존재하는 경우가 많지 않음! (데이터 많지 않아 ㅠㅠ)

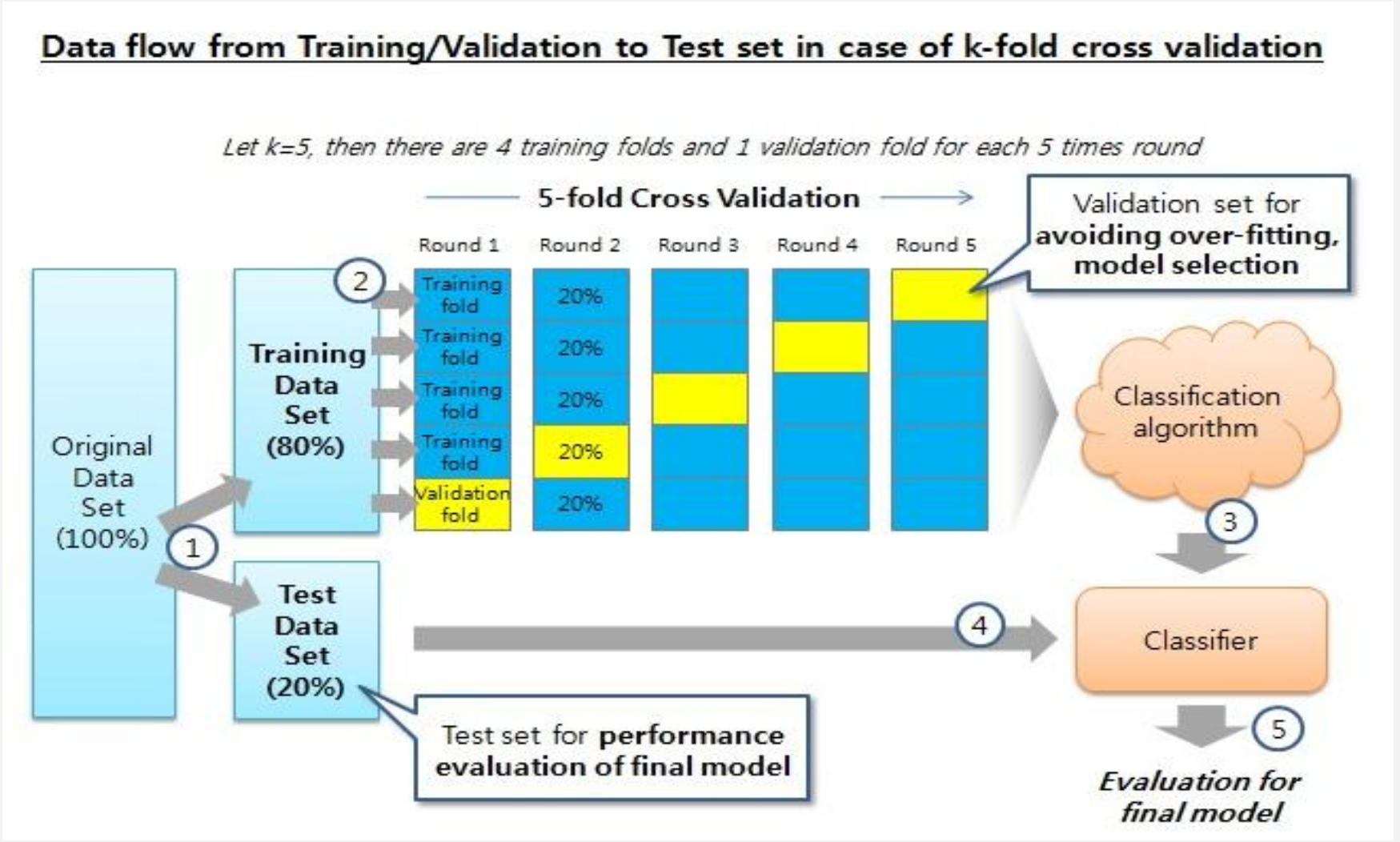
그래서 대부분 **학습용(train)**으로 확보한 데이터를 **학습용(train)과 검증용(validation)**으로 분리하여 학습용 데이터로 학습한 후 검증용 데이터로 검증!

이 때, 데이터를 어떻게 분리하느냐에 따라 검증 성능이 조금씩 달라질 수 있으므로,

여러 가지 방식으로 데이터를 분리하여 검증을 실시하고 **평균 성능**을 구함!



Train & Test로 나눈 Cross Validation



Train & Validation & Test로 나눈 Cross Validation

04.Cross Validation (교차검증)

주로 언제 사용?

“ 데이터의 양이 충분하지 않을 때! “

-> 적은 data지만, “반복”을 통해 여러 번 사용가능하니까!

“ 분류기 성능 측정의 통계적 신뢰도를 높이기 위해! “

-> 하나의 결과가 아닌, 여러 개의 결과를 평균내기 때문에!

제대로 이해했는지 마지막 check!

https://youtu.be/TIgfjmp-4BA

감 사 합 니 다