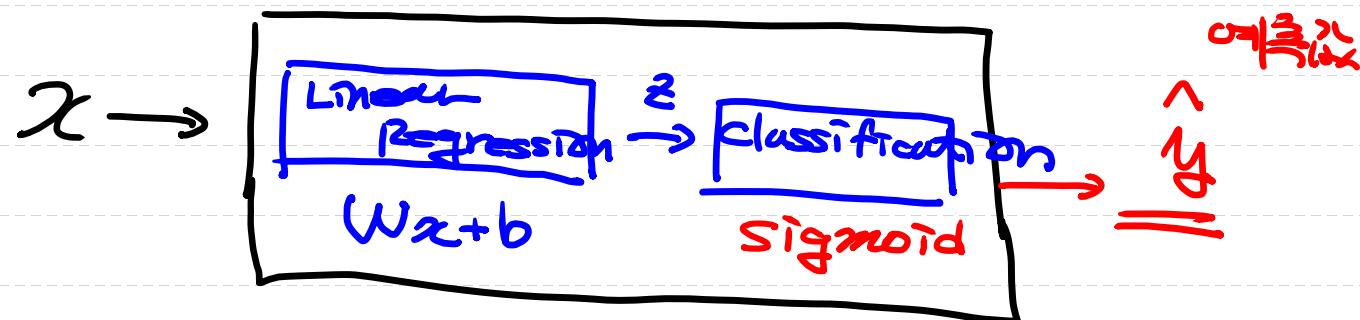


- 04/04 Logistic Regression
 - [binary classification (이중분류)]
 - Multinomial (다중분류)



Admission.csv を 이용해가 Logistic 구현
Regression] 투표 !!

Model 을 구현한 후 Evaluation (성능평가) 를 진행해야 해요 ~~

→ 성능평가도 많은 기준이 있어요 → Metrics

일단 먼저 성능평가는 어떻게 진행되는 건지부터 알아보아요 ~~



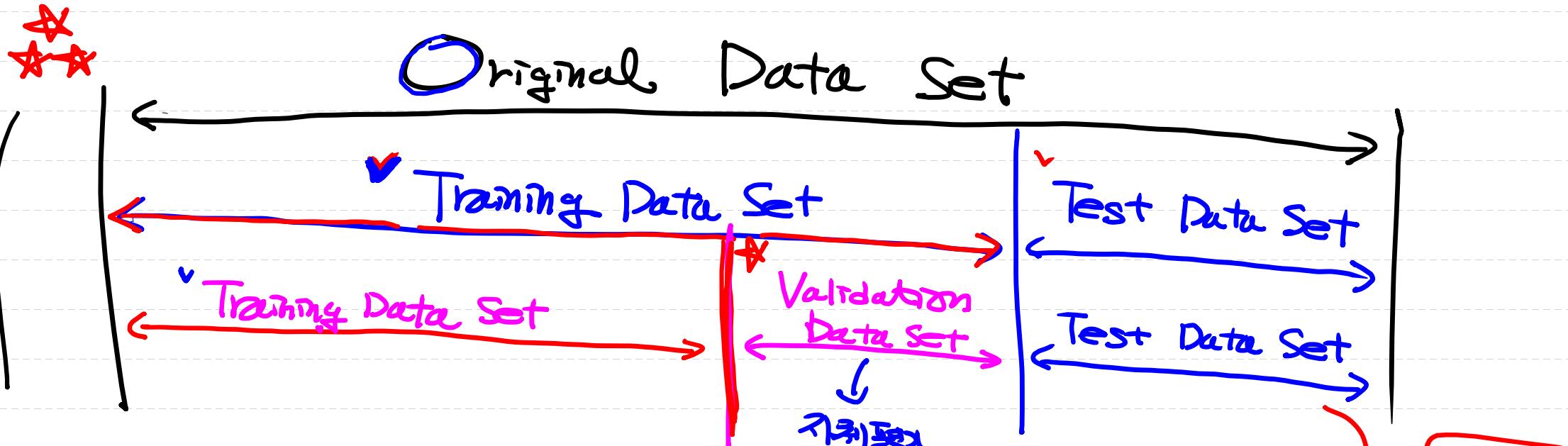
* Evaluation (성능평가)를 진행할 때

[Training Data Set 을 이용해서 학습한 후]

Training Data Set 을 이용해서 evaluation하면 안되요 !!

"왜 그려요 ?? " → 성능해보면 훈련한 모델예요~~

그다면 어떻게 해야 하나요 ?



Training Data Set : 학습 데이터 셋

→ 'model의 성능을 위해'

Test Data Set : 최종 성능을 평가(evaluation)하기 위해 사용.

모델 테스트를
하기 않아요

• Validation Data Set

→ 우리가 만든 모델이 잘 만들어졌는지 그려지기 위해서

자체 평가를 하기 위한 Data Set

→ Training Data Set의 일부를 셀ектив하게 빼기

→ 이런 방식을 "Hold-out validation"

단점이 있어서요

① Training Data Set이 작아져요!!

② 데이터의 편향이 생겨나 있어요!

↳ 우리가 해결해야 하는 문제!!

다른 Validation 방식

⇒ "k-Fold Cross Validation"

직도학습 (Supervised Learning) * Regression

Metrics (평가지표)

연속적인 숫자 예측

불균형 분류 → Classification

Regression의 Metrics

① MAE (Mean Absolute Error)

→ 예측값과 진정의 차이를 절대값으로 평균!

직관적이다, 단위가 같아요. MAE \rightarrow 10 이면 진정과 10까지는 4로!!

단, Scale에 따라 의존적

\downarrow
100일때

\Rightarrow MAE를 쓰울화려면 \rightarrow sklearn이 기능을 제공!!

진정

예측

$$\textcircled{2} \quad \text{MSE} (\text{Mean Squared Error}) = \frac{1}{n} \sum (t - \hat{t})^2$$

\downarrow

MAE 보다 error가 가중치(제곱)를 주는 Metrics. 이점에 더 민감.

$\underbrace{\hspace{10em}}$

$\underbrace{\hspace{10em}}$ 제곱근

$$\textcircled{3} \quad \text{RMSE} (\text{Root Mean Squared Error}) = \sqrt{\frac{1}{n} \sum (t - \hat{t})^2}$$

④ MAPE (Mean Absolute Percentage Error)

MAE의 100분율 표현

$$\text{MAPE} = \frac{100\%}{n} \sum \left(\left| \frac{x - \hat{x}}{x} \right| \right)$$

\rightarrow x % \hat{x} % 실际의 값을 가진 흐름 model

⑤ R squared → 분산을 기반으로 한 평가지표

$$R^{\text{squared}} = \frac{\text{예측값의 Variance}}{\text{정답의 Variance}}$$

⇒ “과거 가까운 흐름 총을 모를”

* Classification의 Metrics!! → 자세히 알아둘 필요가 있나요!

* Confusion Matrix (binary classification)

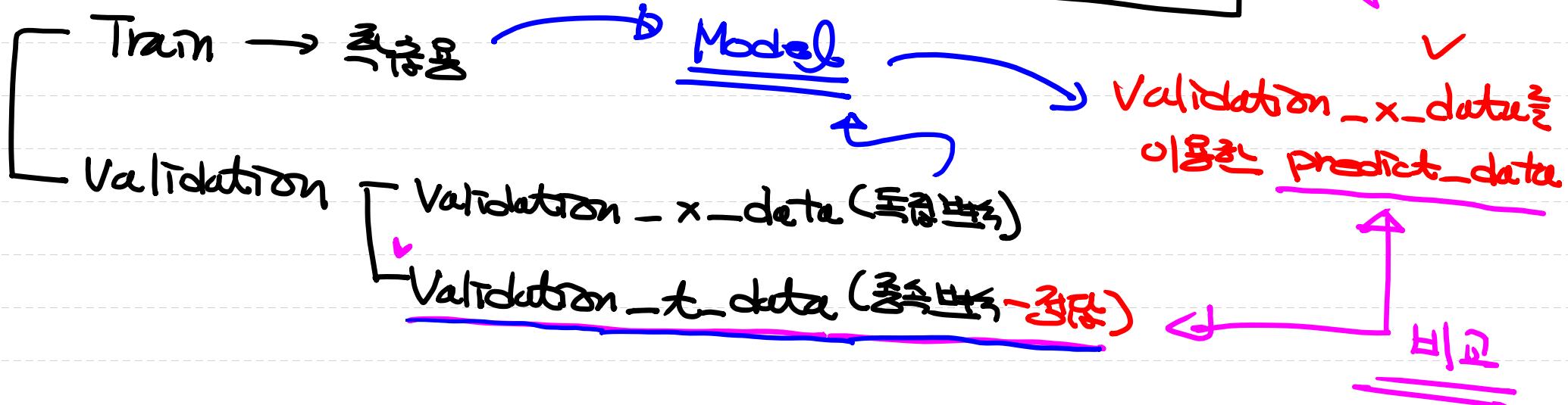
Confusion Matrix

		실제정답	
		TRUE	FALSE
분류결과 (예측값)	TRUE	TRUE positive TP	FALSE positive FP
	FALSE	FALSE Negative FN	TRUE Negative TN

→ Positive → negative

FP → Type I Error
FN → Type II Error

* Data



* O! Confusion Matrix를 이용해서 Metric을 정의!!!

A red star and a red circle containing diagonal lines.

Accuracy (정확도)

$$* \text{accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FN} + \text{FP} + \text{TN}}$$

고려해야 할 사항:

bias 브이어스의 표현
(치우침)을 반드시

도기이터의 폭행이 심嗽 예.)

고급

↪ CT 사진으로 음 흡연 드릴!!

이런 데이터를 사용하는 Model은 성능을 어떻게
하고자 좋을까요?

12

Recall (재현율)

$$* \text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \rightarrow \text{총답이 } 1 \text{의 경우}$$

Sensitivity
hit rate

v 정답

四

$$\begin{array}{r} 0 \\ - 0 \\ \hline 0 \\ 0 \\ - 0 \\ \hline 0 \end{array}$$

$O \rightarrow TN^1$
 $O \rightarrow FN$
 $I \rightarrow FP$
 $O \rightarrow TN^2$
 $O \rightarrow FN$
 $I \rightarrow TP$
 $O \rightarrow TN^3$

$$\frac{4}{7}$$

accuracy

99.9%



Precision (정밀도)

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

일반적으로

- (Recall ↑, Precision ↓)
- (Recall ↓, Precision ↑)

반례 고지의 경우가 많아요

A 키워드 → 기존 정보를 찾았어요!!
 ↳ 사진에 고양이가 있나요?
 B 키워드 → “ ” .

* 문제: 오점들이 많다!

오점률 13% 10건
 ✗ 우리 카페의 Cat-finder 기준은
 * 검출율이 95%입니다
 → 우리 카페의 Cat-searcher 기준은
 * 검출율이 60%입니다
 오점률이 거의 ✗

(4) Precision - Recall graph

*



F1-score

↳ Precision과 recall의 "조화평균"

*

$$\text{F1-score} = \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

(5) Fall-out $\Rightarrow \frac{FP}{TN+FP}$

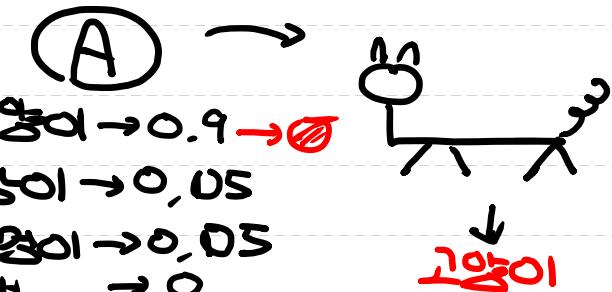
* 실제값이 False인 데 Model이 True로 잘못 예측한 비율

(6) ROC curve \rightarrow recall 의 값을 사용해 Fall-out

"Log-loss"

(다중분류에서 일반적으로 사용)

A



B

- ① 고양이 $\rightarrow 0.4 \rightarrow$
- ② 코뿔이 $\rightarrow 0.2$
- ③ 엄嚇이 $\rightarrow 0.3$
- ④ 새 $\rightarrow 0.1$

④ 이제부터는 code 작성시 model 평가도 같이 구현 (accuracy 측정이용)