

로지스틱 회귀모형(Logistic Regression)



Fall, 2021

Syllabus

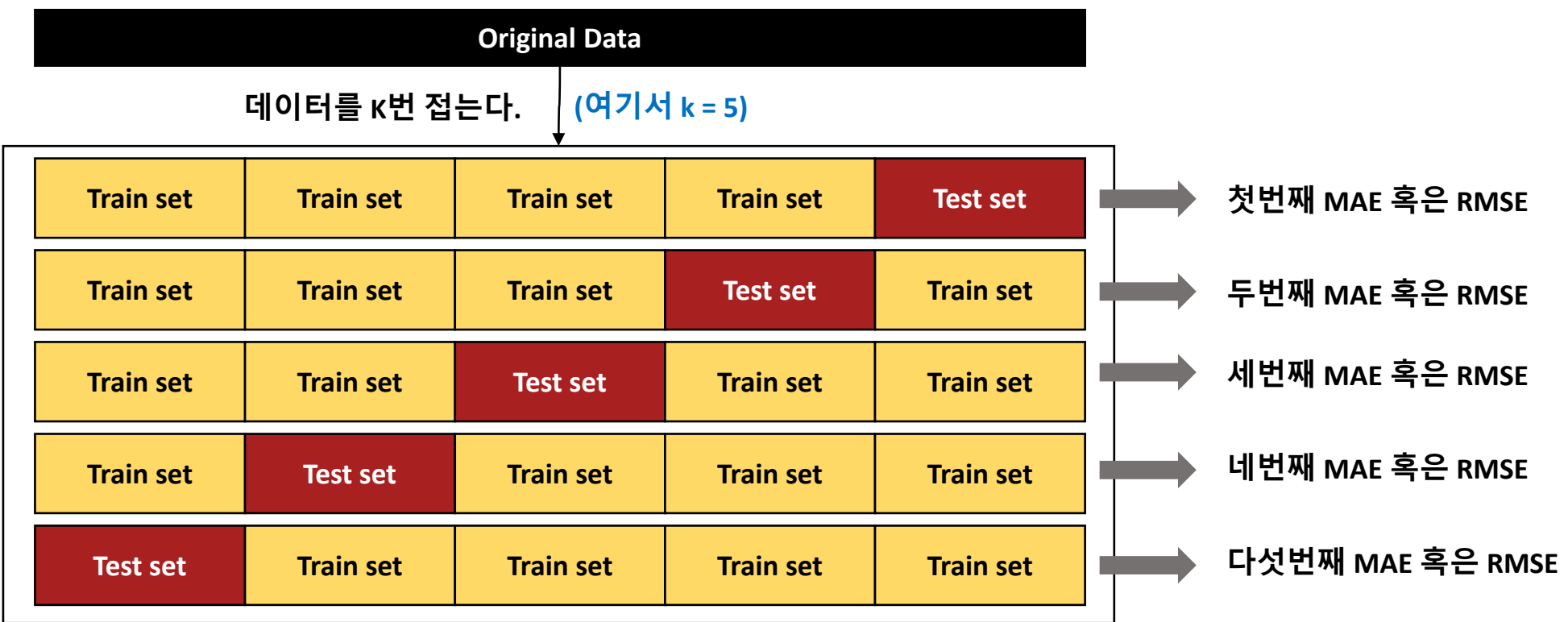
Week	Date	Topic	Note
1	9/6(월)	R Basic - R 기초 문법 학습	
2	9/13(월)	R Basic – Data Manipulation I	과제#1
3	9/20(월) (추석)	<추석> (보충영상) R Basic - Data Manipulation II	
4	9/27(월)	Descriptive Analytics I - 데이터 요약하기/상관관계/차이검증	과제#2
5	10/4(월) (대체공휴일)	<대체공휴일> (보충영상) Descriptive Analytics II - 데이터 시각화	과제#2
6	10/11(월) (대체공휴일)	<대체공휴일> (보충영상) Supplementary Topic I - 외부 데이터 수집 (정적 콘텐츠 수집)	과제#4 과제#3
7	10/18(월)	Predictive Analytics I – Linear regression	
8	10/25(월)	Predictive Analytics II – Logistic Regression	시험 대체 수업
9	11/1(월)	Predictive Analytics III – Clustering & Latent Class Analysis	과제#4
10	11/8(월)	Predictive Analytics IV – Tree-based Model and Bagging (Random Forest)	
11	11/15(월)	Predictive Analytics V – Association Rules	
12	11/22(월)	Prescriptive Analytics I – Linear Programming	과제#5
13	11/29(월)	Prescriptive Analytics II – Data Envelopment Analysis (DEA)	
14	12/6(월)	Prescriptive Analytics III – Integer Programming	과제#6
15	12/13(월)	Prescriptive Analytics IV – Simulation	Quiz
16	12/20(월)	Final Presentation	

Lecture 8-1

교차 검증법
(복습)

모형개선#1 - K-겹 교차 검증법(Cross Validation)

모형 개선을 위해 우리는 수 차례 모형을 개선시키는 데, 이 과정에서 고정된 Train set과 Test set을 활용한다면 결국 과적합(Overfitting) 문제가 발생할 수 있음.



모형의 총 MAE = mean(첫번째 MAE, 두번째 MAE, ..., 다섯번째 MAE)

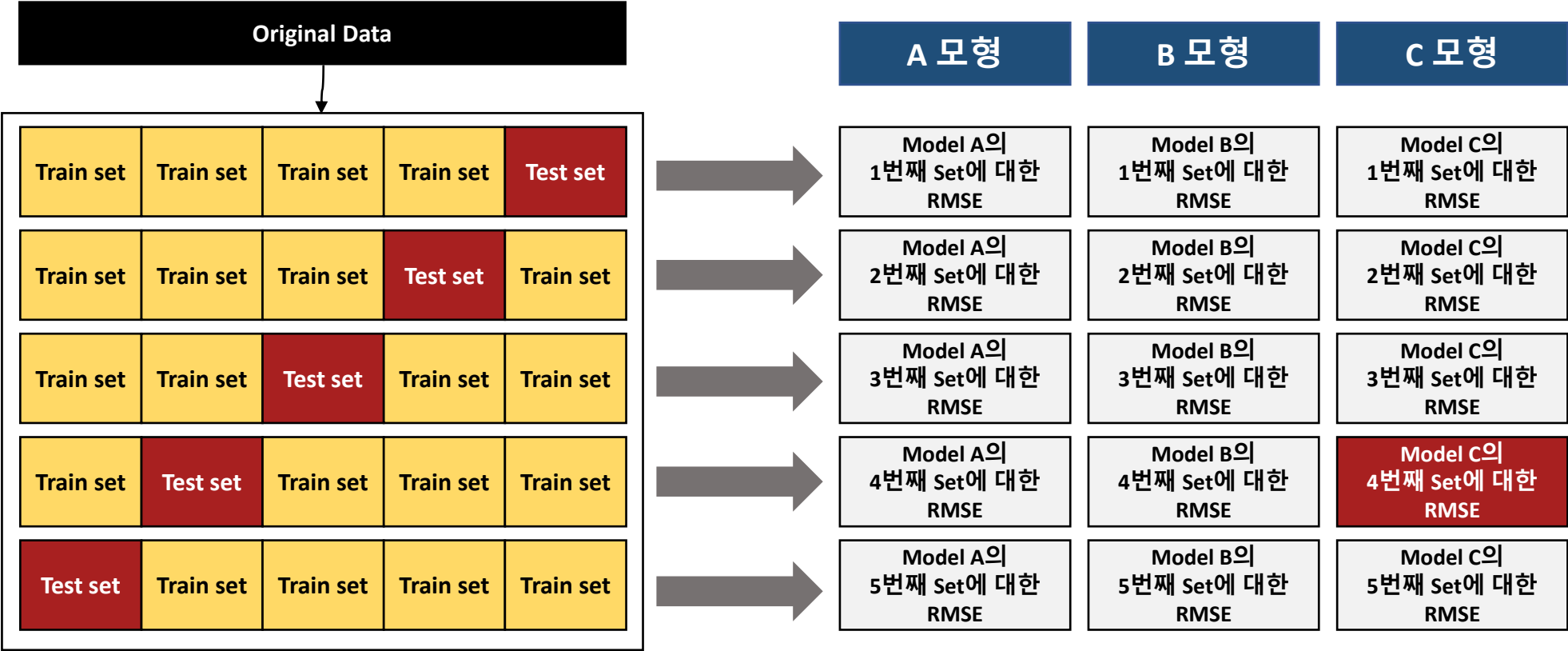
모형의 총 RMSE = mean(첫번째 RMSE, 두번째 RMSE, ..., 다섯번째 RMSE)

모형개선#1 – K-겹 교차 검증법(Cross Validation)

- 1) 데이터를 Random하게 k개의 같은 크기로 쪼갬. 그럼 k개의 Folded set이 나옴
- 2) 첫번째 folded set에서 K-1개의 데이터를 Training set으로 이용하여 모델을 학습시킴
- 3) 나머지 1개의 데이터를 Test set으로 하여 Y값을 예측(Prediction)
- 4) 2)~3)번 과정을 K번 반복해 모든 Y값에 대한 예측값(Predicted value)을 찾아냄
- 5) 1~4번까지 과정을 각각의 후보모델마다 실행함
- 6) 1-5까지 과정을 반복함
- 7) 각 단계마다 MAE 혹은 RMSE를 계산함.
- 8) MAE 및 RMSE를 모아서 가장 작은 값을 나타내는 모델을 선택함

모형개선#1 - K-겹 교차 검증법(Cross Validation)

예) 현재 고려 중인 모형은 A, B, C 모형 3개이고, 5-folded 교차검증을 하려고 한다.
평가 기준은 RMSE를 기준으로 최적 모형을 도출하고자 한다.



총 “15개의 모형”을 비교하는 것과 동일함!

모형개선#2 – 피처엔지니어링 (Feature Engineering)

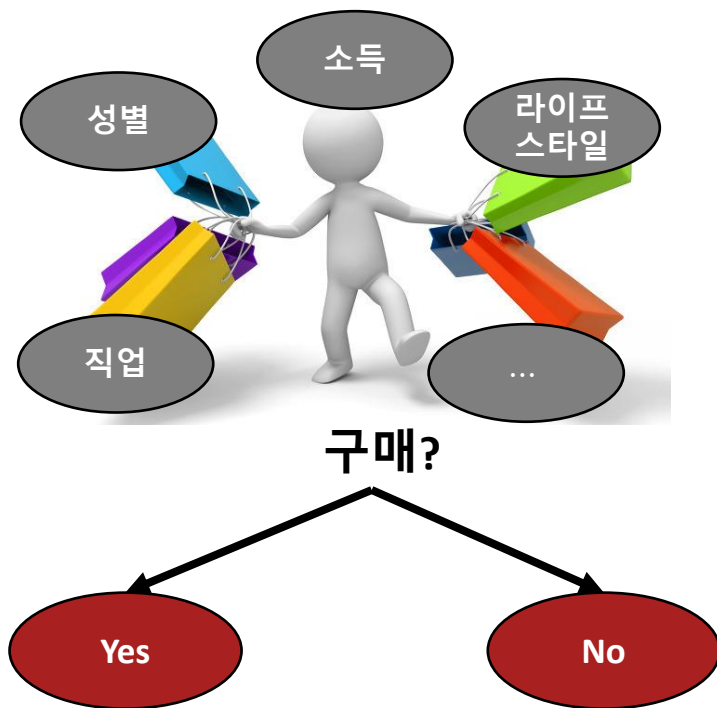
피처 엔지니어링은 주어진 피처(Feature)들을 이용해
해당 도메인에 대한 지식 및 특성 등을 미리 알거나
탐색적 분석을 통해 알게 된 사실을 바탕으로
유의미한 변수를 생성, 선택 및 변환하는 과정

Lecture 8-2

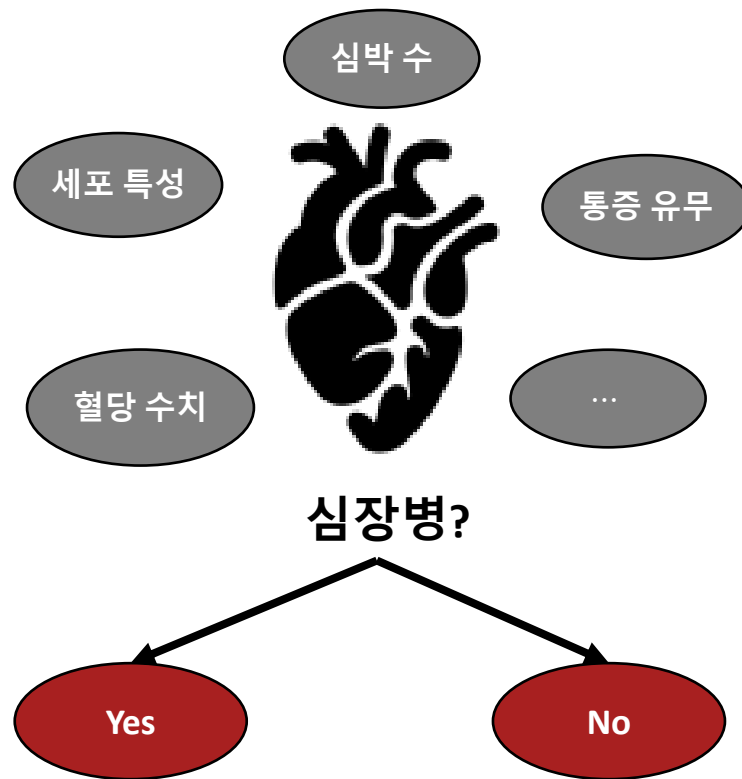
로지스틱 회귀란?

분류모형의 동기(Motivation)

A 고객이우리 회사 제품을
구매할 확률이 어떻게 될까?



이 환자가 심장병일 확률은 얼마나 될까?



선형 회귀모형처럼 직접 값을 예측하는 경우도 있지만, 더 많은 현실문제들은 분류 문제에 직면해 있다.

분류 모형(Classification Model)

레이블(Label)(Y)이
있는 분류모형

- 로지스틱 회귀모형
- 판별분석
- 최근접 이웃법(K-NN)
- 서포트벡터머신(SVM)
- 의사결정나무
- 랜덤포레스트
- 부스팅

레이블(Label)(Y)이
없는 분류모형

- 군집분석(Clustering)
- 잠재계층분석(Latent Class Model)

로지스틱 회귀모형(Logistic Regression)

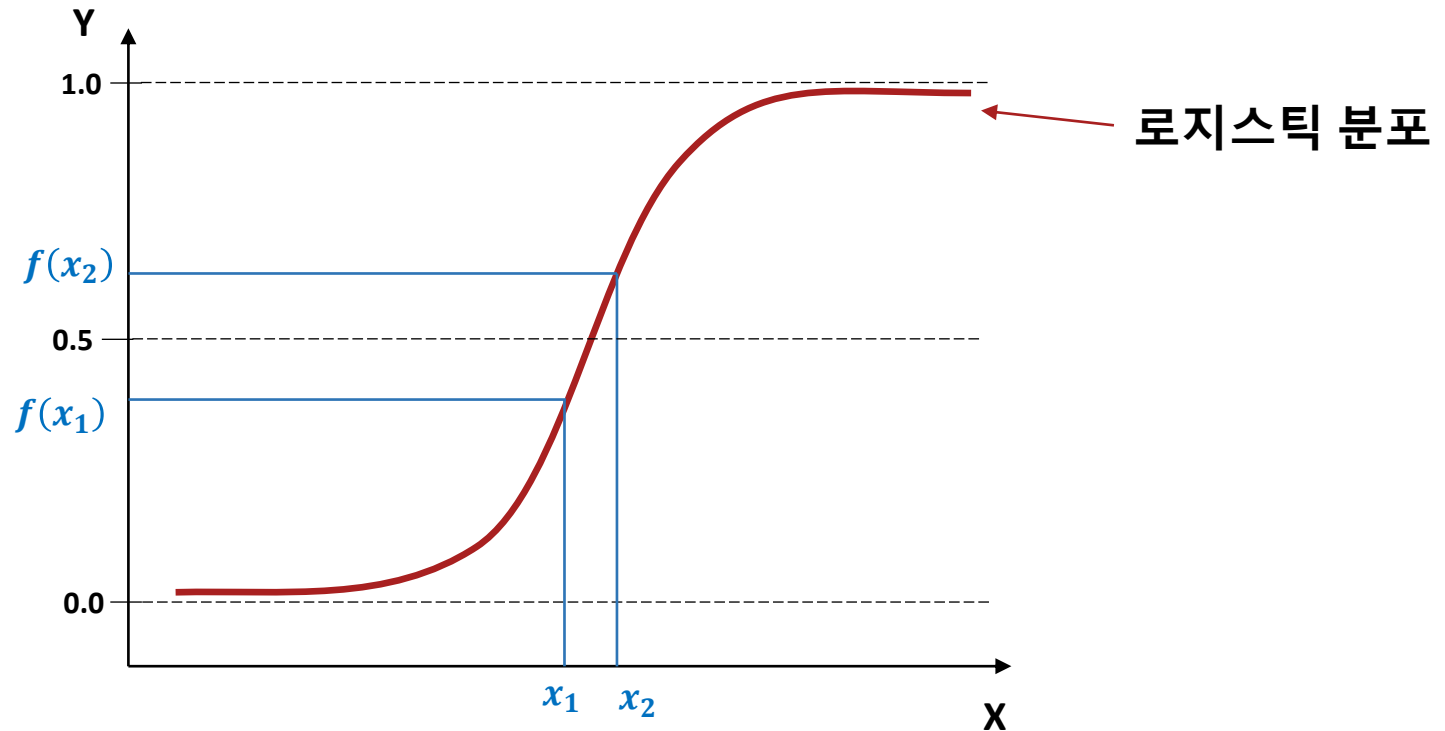
- 로지스틱 회귀분석은 선형회귀분석과 마찬가지로 예측변수와 결과변수 사이의 관계를 특정하는 모형임
- 다만, 로지스틱 회귀분석은 결과변수인 Y 가 연속형(Continuous)가 아니라 범주형 (Categorical)인 경우의 문제를 푸는 데 사용됨
- 로지스틱 회귀모형은 사건이 발생할 ‘확률(Probability)’를 도출함으로써 “발생” 또는 “미발생” 의 경우로 분류할 수 있음.

$p = P(\text{Success} \mid X \text{ data})$: 주어진 x 데이터 하에서 “성공”할 확률

$1 - p = P(\text{Fail} \mid X \text{ data})$: 주어진 x 데이터 하에서 “실패”할 확률

- 로지스틱 회귀모형은 새로운 관측치가 어떤 클래스에 속할 지를 예측하기 위해 각 클래스에 속할 성향(=확률)을 계산해 해당 클래스로 분류하는 데 많이 활용됨. 따라서, 로지스틱 회귀모형은 분류(Classification) 문제에서 매우 다양하게 적용되고 있음
- 로지스틱 회귀모형은 “로지스틱 함수”로부터 이름이 유래됨

로지스틱 함수(Logistic Function)



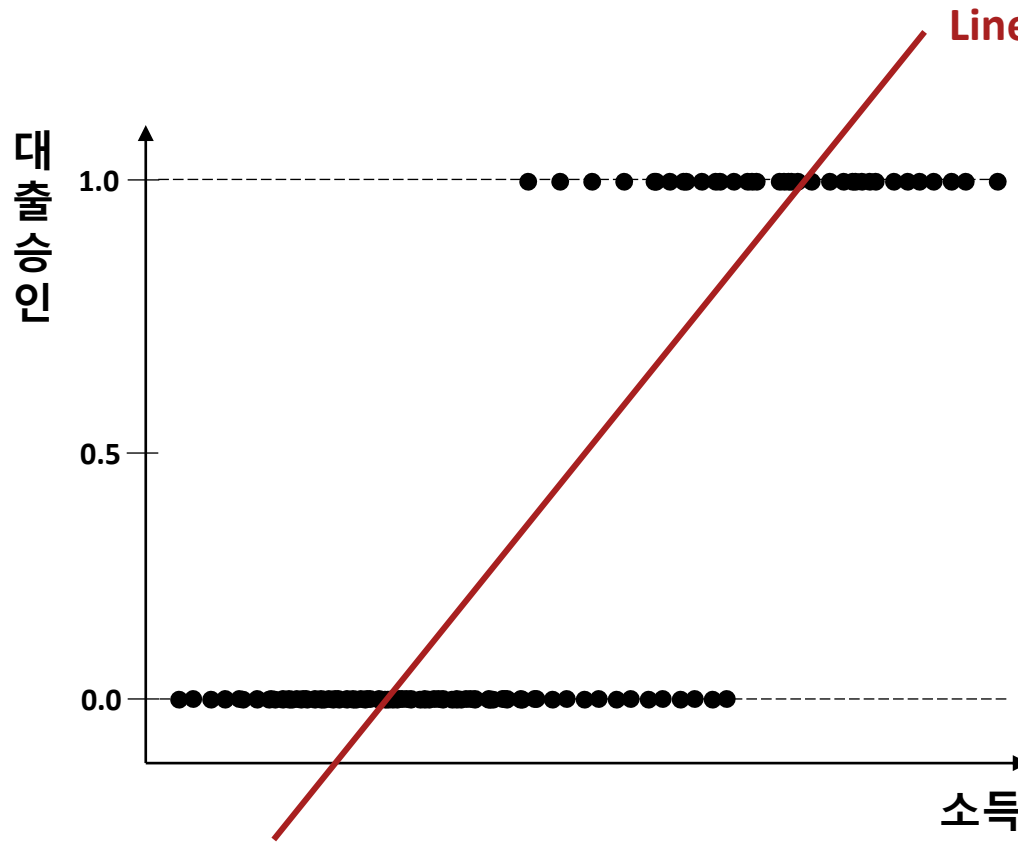
로지스틱 함수 :

$$y = f(x) = \frac{1}{1 + e^{-x}} = \frac{e^x}{1 + e^x}$$

로지스틱 함수 값은 항상()에서() 사이
값을 가진다.

로지스틱 회귀모형(Logistic Regression) 원리

선형회귀분석이 연속형(Continuous) 종속변수를 예측한다면, 로지스틱 회귀분석은 **이산형(Discrete) 변수인 종속변수를 예측함**. 가령, E-mail이 스팸인지 아닌지 혹은 은행고객에게 대출을 할지 말지 등을 결정함



$$y \in \{0, 1\}$$

Class 0 ($y = 0$): *not accepted* (대출미승인)

Class 1 ($y = 1$): *accepted* (대출승인)

만약, 선형회귀모형으로
적합(fitting)시키면 어떻게 될까?

예측결과를 해석할 수 없음!

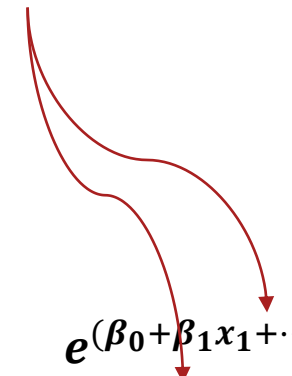
로지스틱 회귀모형(Logistic Regression) 원리

- 선택지가 0 또는 1인 경우, OLS를 이용한 선형회귀모형으로 추정해보자. 어떤 문제가 생기는가?
- 선형회귀모형으로 추정한 \hat{y} 이 0 또는 1의 값을 갖지 못하는 문제가 발생함

선형회귀모형으로 추정 :

$$\hat{y} = \beta_0 + \beta_1 x_1 + \cdots + \beta_q x_q \quad (X)$$

로지스틱 모형 :

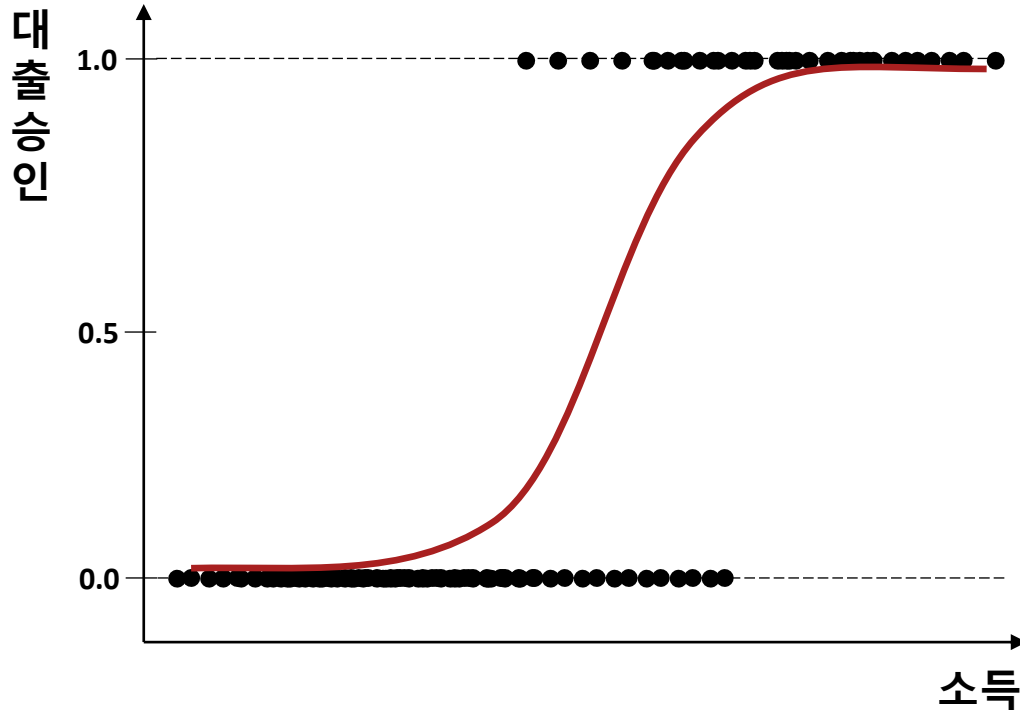
$$0 < \hat{y} = Pr(y = 1|x) = \frac{e^{(\beta_0 + \beta_1 x_1 + \cdots + \beta_q x_q)}}{1 + e^{(\beta_0 + \beta_1 x_1 + \cdots + \beta_q x_q)}} < 1 \quad (O)$$


선형 회귀모형의 회귀방정식을 로지스틱 모형의 x 자리에 넣어서 모형을 추정하므로 변수 간 관계가 설명되면서, 확률값을 얻을 수 있어 로지스틱 + 회귀 모형 이라고 부른다.

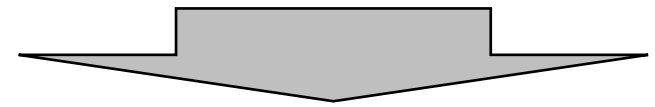
로지스틱 회귀모형(Logistic Regression) 원리

로지스틱 모형에서는 분석결과가 “사건이 발생할 확률”로 도출이 되도록 함수를 만들어 추정함.
따라서, 로지스틱 모형의 결과는 확률로 주어짐

Logistic Regression Model



$$0 < f(x) = \frac{1}{1 + e^{-\beta_0 - \beta_1 x_1}} < 1$$



Class 1(대출 승인)에 속할 확률과
Class 0(대출 미승인)에 속할 확률을 예측!

예측값(Predicted value) ≥ 0.5 이면, Class 1에 분류
예측값(Predicted value) < 0.5 이면, Class 0에 분류

※참고 - 로지스틱 모형 유도

➤ 로지스틱 회귀모형을 오즈비(Odds ratio)와 로짓함수(Logit function)을 이용해 유도해보자.

$$\text{Odds ratio} = \frac{\Pr(y = 1|x)}{1 - \Pr(y = 1|x)}$$

: 어떤 사건이 일어나지 않을 확률 대비 일어날 확률
Pr이 1에 가까울수록 오즈비는 무한대(∞), 0에 가까울수록 오즈비는 0에 가까워짐

양변에 Log

$$\log(\text{odds}) = \log\left(\frac{\Pr(y = 1|x)}{1 - \Pr(y = 1|x)}\right) = \beta_0 + \beta_1 x_1 + \cdots + \beta_q x_q$$

: 이제, $\log(\text{odds})$ 값이 $-\infty$ 에서 $+\infty$ 값을 가지므로 선형회귀모형식을 가져올 수 있음

양변에 Exp

“Logit” 변환

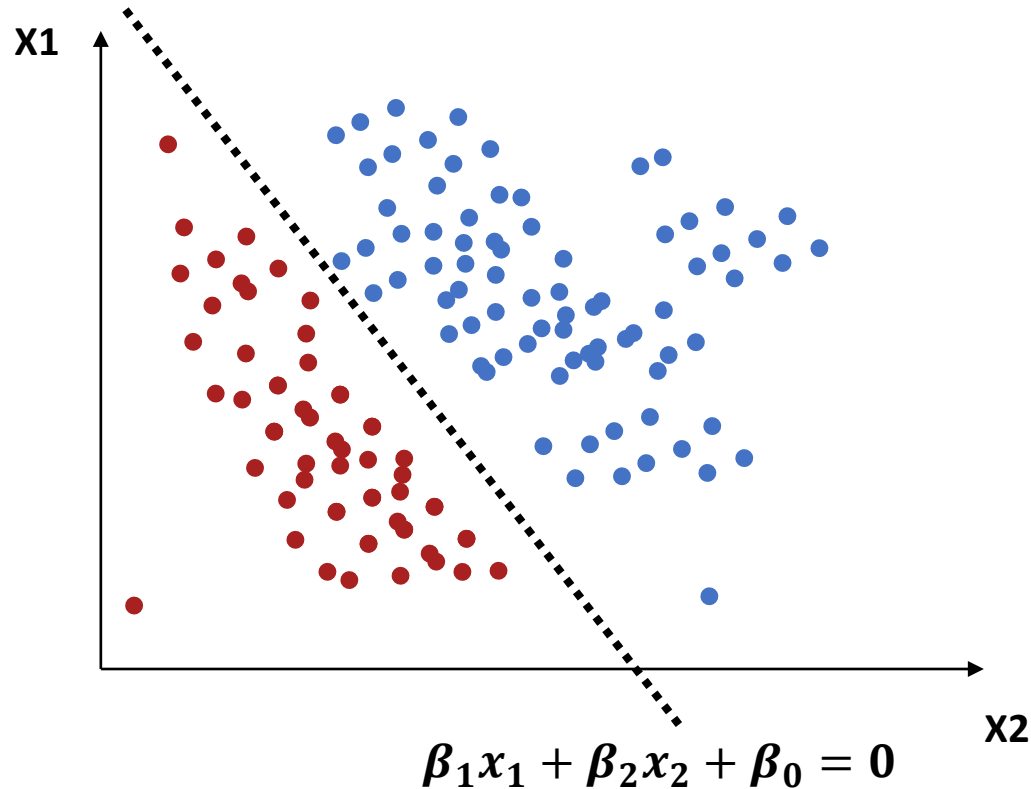
$$\frac{\Pr(y = 1|x)}{1 - \Pr(y = 1|x)} = e^{\beta_0 + \beta_1 x_1 + \cdots + \beta_q x_q}$$

$$\Leftrightarrow \Pr(y = 1|x) = (1 - \Pr(y = 1|x))e^{\beta_0 + \beta_1 x_1 + \cdots + \beta_q x_q}$$

$$\Leftrightarrow \Pr(y = 1|x) + \Pr(y = 1|x) * e^{\beta_0 + \beta_1 x_1 + \cdots + \beta_q x_q} = e^{\beta_0 + \beta_1 x_1 + \cdots + \beta_q x_q}$$

$$\Leftrightarrow \Pr(y = 1|x) = \frac{e^{\beta_0 + \beta_1 x_1 + \cdots + \beta_q x_q}}{1 + e^{\beta_0 + \beta_1 x_1 + \cdots + \beta_q x_q}} = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x_1 + \cdots + \beta_q x_q)}} \quad : \text{Logistic Model}$$

로지스틱 함수에 선형 회귀식이 들어간다는 의미는 ?



$\beta_1 x_1 + \beta_2 x_2 + \beta_0 > 0$: 파란색

$\beta_1 x_1 + \beta_2 x_2 + \beta_0 < 0$: 빨간색

선형 회귀식이 로지스틱 함수(Logistic function) 내에 들어간다는 말은 “선형 분류 경계”를 만들어 클래스(Class)(ex. 생존 vs 사망, 이탈 vs 유지 등)를 분류한다는 의미임. 즉, 로지스틱 모형은 선형 분류모형 !

Lecture 8-3

분류모형의 성능 평가

혼동행렬(Confusion Matrix)

혼동행렬(Confusion Matrix)는 지도학습 분류기법의 성능을 검증하기 위한 척도(Measure)로 혼동행렬을 통해 정확도(Accuracy), 민감도(Sensitivity), 특이도(Specificity), 정밀도(Precision)을 측정할 수 있음

		실제 결과 (Actual)	
		참 (TRUE)	거짓 (FALSE)
분류예측 결과 (Predicted)	참 (TRUE)	TP (True Positive)	FP (False Positive) ← “ Type I Error ”
	거짓 (FALSE)	FN (False Negative) ← “ Type II Error ”	TN (True Negative)

Type I error vs. Type II error

제2종 오류: **실제** 환자가 **암 환자**인데, 진단결과 **암 환자가 아니라고 분류**하는 경우

제1종 오류: **실제** 환자가 **암이 아닌데**, 진단결과 **암 환자라고 분류**하는 경우

		실제 결과 (Actual)	
		참 (TRUE)	거짓 (FALSE)
분류예측 결과 (Predicted)	참 (TRUE)	TP (True Positive)	FP (False Positive) ← “ Type I Error ”
	거짓 (FALSE)	FN (False Negative) ← “ Type II Error ”	TN (True Negative)

분류모형 검증 지표(Index)

구분	정의	측정
정확도 (Accuracy)	전체 예측결과중 올바르게 예측한 것의 비율	$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$ $Errorrate = \frac{FP + FN}{TP + TN + FP + FN} = 1 - Accuracy$
민감도 (Sensitivity)	실제로 참(True)인 것 중에서 참(True)으로 분류한 비율	$Sensitivity = \frac{TP}{TP + FN}$
특이도 (Specificity)	실제로 거짓(False)인 것 중에서 거짓(False)으로 분류한 비율	$Specificity = \frac{TN}{TN + FP}$
정밀도 (Precision)	참(True) 이라고 예측한 것 중에 실제로 참(True)인 비율 혹은 거짓(False) 이라고 예측한 것 중에 실제로 거짓(False)인 비율	$Precision = \frac{TP}{TP + FP} = PositivePredictValue$ $Precision = \frac{TN}{FN + TN} = Negative PredictValue$

왜 정확도(Accuracy) 외 지표를 보는가?

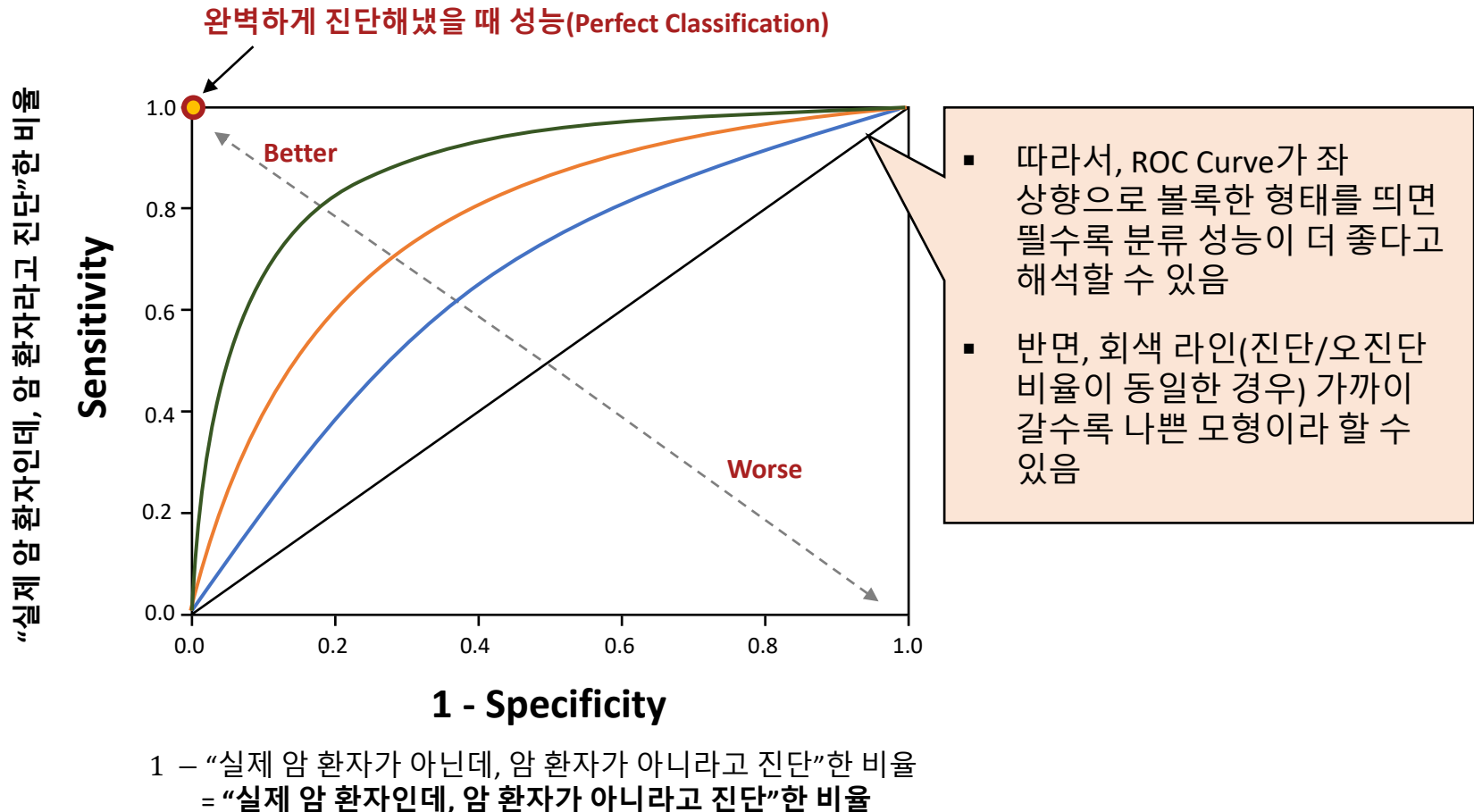
		실제 결과 (Actual)	
		암	암 아님
분류예측 결과 (Predicted)	암	13	3
	암 아님	28	24,826

Accuracy	$\frac{13 + 24,826}{13 + 3 + 28 + 24,826} = 99.88\%$
Sensitivity	$\frac{13}{13 + 28} = 31.71\%$
Specificity	$\frac{24,826}{3 + 24,826} = 99.99\%$
Precision (Positive Predicted value)	$\frac{13}{13 + 3} = 81.25\%$

우리가 Target으로 하는 True 보다 False의 Case가 압도적으로 많은 경우들이 있기 때문에, 정확도를 절대적 기준으로 삼으면 치명적인 오류를 범할 수 있음

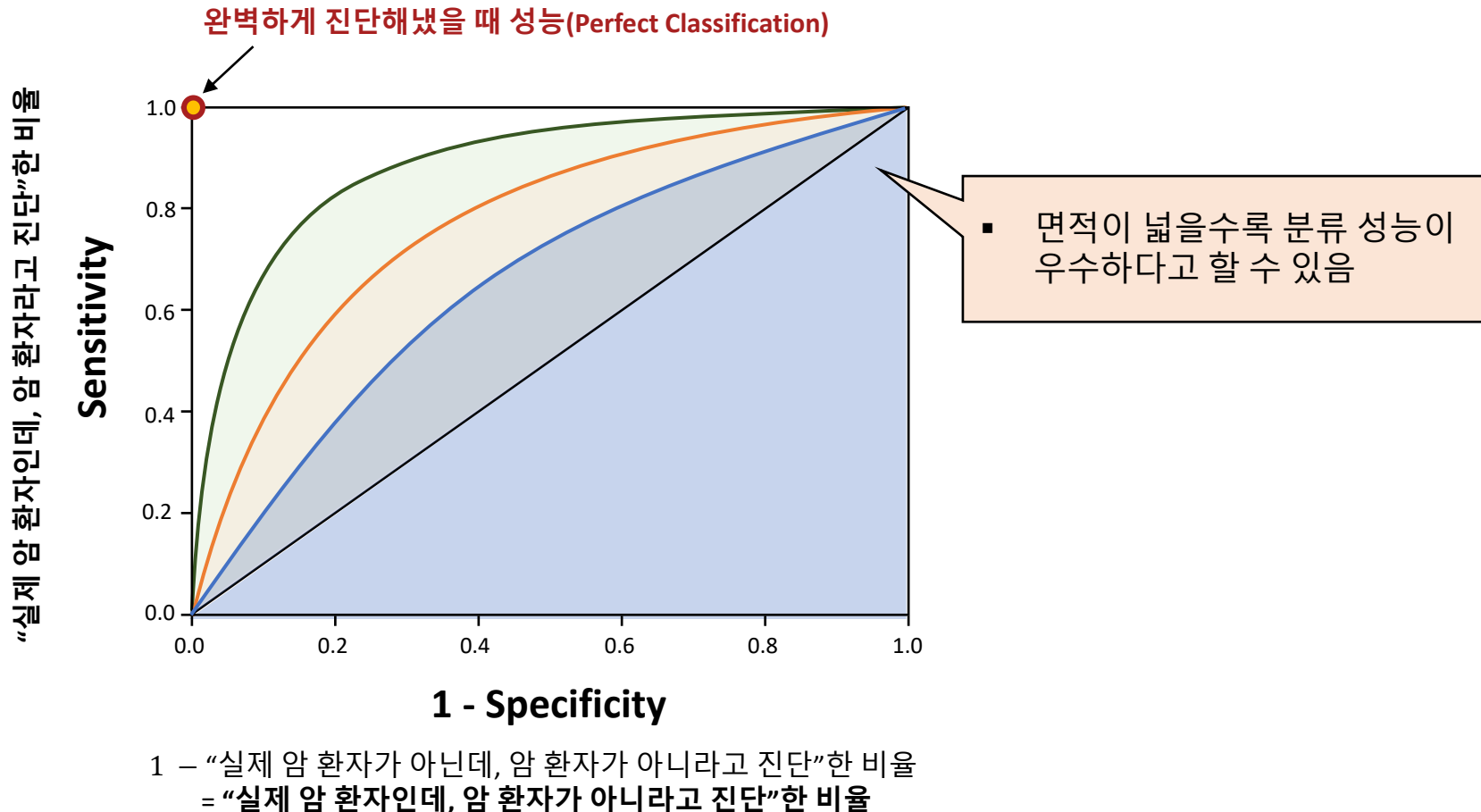
ROC Curve와 AUC

- 정확도(Accuracy)만으로 모델 성능을 평가할 수 없으므로 보완적으로 봐야될 성능 지표로 민감도와 특이도를 시각화한 것이 ROC(Receiver Operating Characteristic curve) curve임



ROC Curve와 AUC

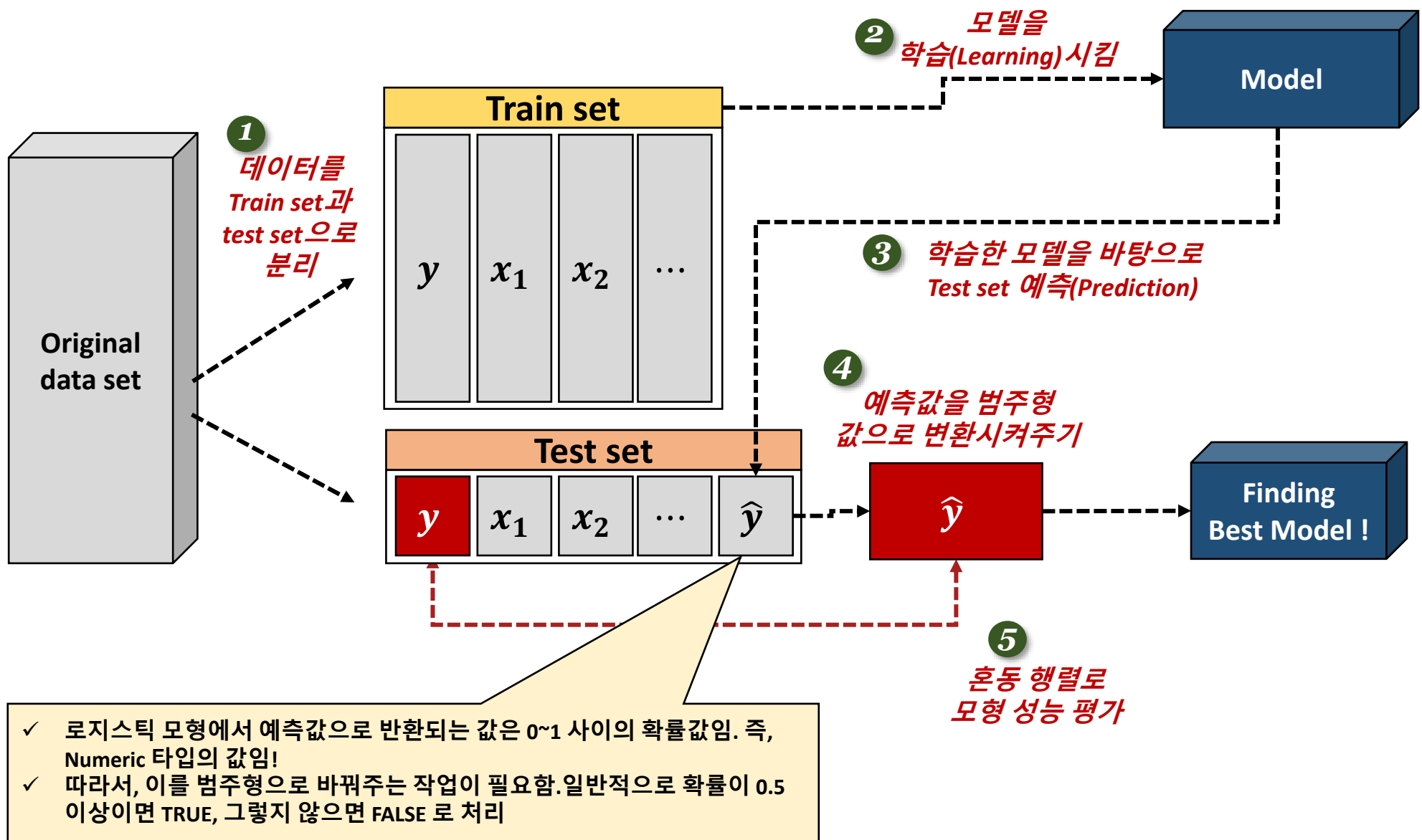
- AUC(Area Under Curve)는 곡선 아래 면적을 나타내는 것으로, 좌상향으로 볼록할수록 면적이 넓어지므로 면적이 넓을수록 성능이 좋다고 할 수 있음. 즉, ROC Curve와 AUC는



Lecture 8-4

분류모형
학습/예측/평가
Process

로지스틱 회귀모형 어떻게 추정하고, 예측하는가?



Lecture 8-5

R에서
Logistic Model
구현하기

예시#1 - 개인대출상품 승낙여부

Universal bank의 Loan campaign에 대해 고객들은 어떻게 반응했을까?



속성(Feature)

Age : 나이

Experience : 직장경력

Education : 교육수준(1=under/2=grad/3=advanced)

Income : 소득

ZIPCode : 우편번호

Family size of customer : 가족 수

CAAvg : 월 평균 신용카드 사용액

Mortgage : 주택자산가치

SecuritiesAccount : 유가증권계정 유무(1=yes/0=no)

CDAccount : 양도성 예금증서 유무(1=yes/0=no)

Online : 인터넷뱅킹 사용유무(1=yes/0=no)

CreditCard : 자사 신용카드 사용유무(1=yes/0=no)

종속변수(Class)

Acceptance : 개인대출상품 승낙 여부 (1=yes/0=no)

Source : Shmueli et al. (2016).

예시#1 - 개인대출상품 승낙여부

```
Call:
glm(formula = Acceptance ~ Age + Experience + Education + Income +
     Family + CCAvg + Mortgage + SecuritiesAccount + CD.Account +
     Online + CreditCard, family = binomial(link = "logit"), data = loan_train)
```

Deviance Residuals:

Min	1Q	Median	3Q	Max
-3.1650	-0.1938	-0.0734	-0.0224	4.1274

Coefficients:

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	-9.2563740	2.2835665	-4.053	5.05e-05 ***
Age	-0.1407968	0.0862587	-1.632	0.102624
Experience	0.1404916	0.0854465	1.644	0.100134
Education2	3.8731949	0.3159415	12.259	< 2e-16 ***
Education3	4.1732738	0.3128607	13.339	< 2e-16 ***
Income	0.0577365	0.0034515	16.728	< 2e-16 ***
Family	0.5983783	0.0933718	6.409	1.47e-10 ***
CCAvg	0.2012250	0.0534108	3.767	0.000165 ***
Mortgage	0.0009504	0.0007112	1.336	0.181416
SecuritiesAccount	-0.9957104	0.3481886	-2.860	0.004241 **
CD.Account	4.0521971	0.4041555	10.026	< 2e-16 ***
Online	-0.7688495	0.1975228	-3.892	9.92e-05 ***
CreditCard	-1.1103863	0.2600455	-4.270	1.95e-05 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 2199.94 on 3499 degrees of freedom
 Residual deviance: 835.83 on 3487 degrees of freedom
 AIC: 861.83

Number of Fisher Scoring iterations: 8

나이, 직장경력, 부동산 가치를
제외한 대부분의 변수가 유의한
영향을 미침

그러나, 나이, 직장경력, 부동산
가치도 영향이 없다고 할 수 없음

예시#1 - 개인대출상품 승낙여부

모형 평가(Model Evaluation)

Confusion Matrix and Statistics

Reference		
Prediction	0	1
0	1335	46
1	18	101

Accuracy : 0.9573

95% CI : (0.9458, 0.967)

No Information Rate : 0.902

P-Value [Acc > NIR] : 8.226e-16

Kappa : 0.7363

McNemar's Test P-Value : 0.0007382

Sensitivity : 0.68707

Specificity : 0.98670

Pos Pred Value : 0.84874

Neg Pred Value : 0.96669

Prevalence : 0.09800

Detection Rate : 0.06733

Detection Prevalence : 0.07933

Balanced Accuracy : 0.83689

'Positive' Class : 1

		실제 결과 (Actual)	
		참 (TRUE)	거짓 (FALSE)
분류예측 결과 (Predicted)	참 (TRUE)	101	1
	거짓 (FALSE)	18	1335

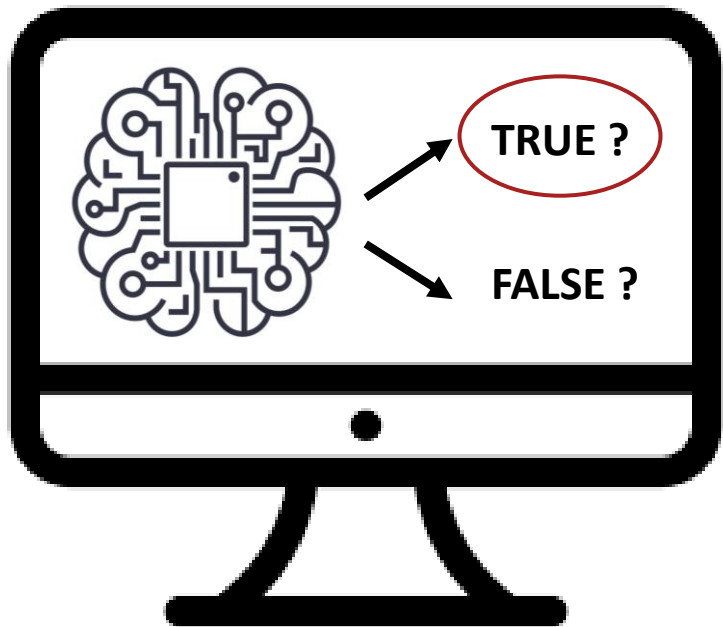
정확도 (Accuracy)	95.7%
민감도 (Sensitivity)	68.7%
특이도 (Specificity)	98.7%
정밀도 (PPV)	84.9%

“ 모형개선 작업 없이도 높은
예측정확도를 나타내고 있음 ”

Lecture 8-6

분류모형
클래스 불균형 문제

Class가 불균형하면 어떤 문제가 발생하는가?



“Agent도 잔머리를 굴린다”

		실제 결과 (Actual)	
		암	암 아님
분류예측 결과 (Predicted)	암	13	3
	암 아님	28	24,826

정확도 : 99.88%

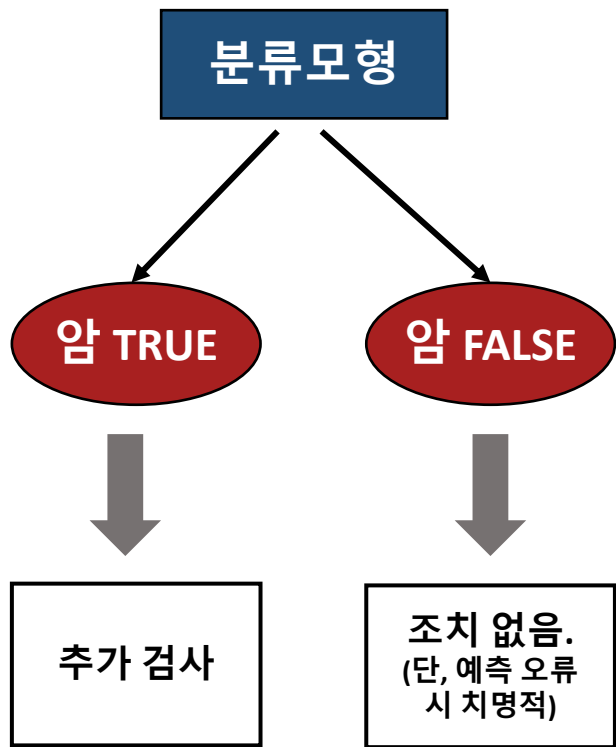


		실제 결과 (Actual)	
		암	암 아님
분류예측 결과 (Predicted)	암	0	0
	암 아님	41	24,829

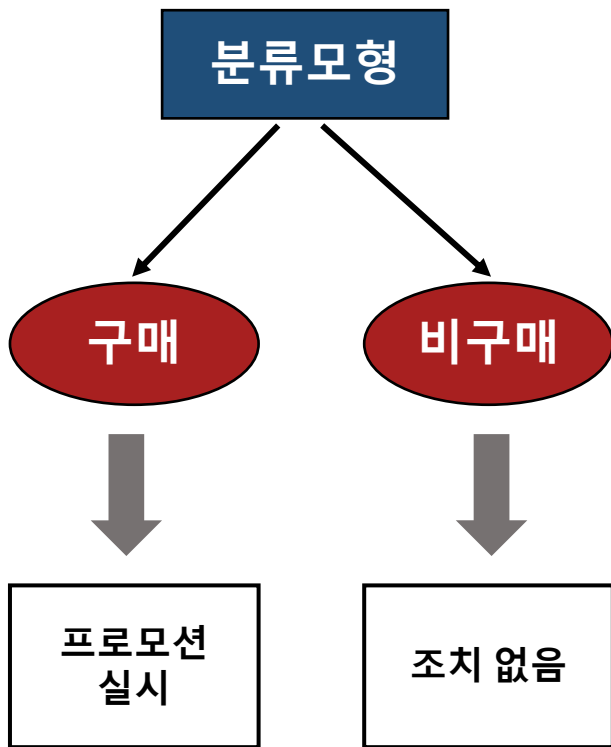
정확도 : 99.82%

Class “수”가 같다고, 의사결정의 “가치”가 같지는 않다.

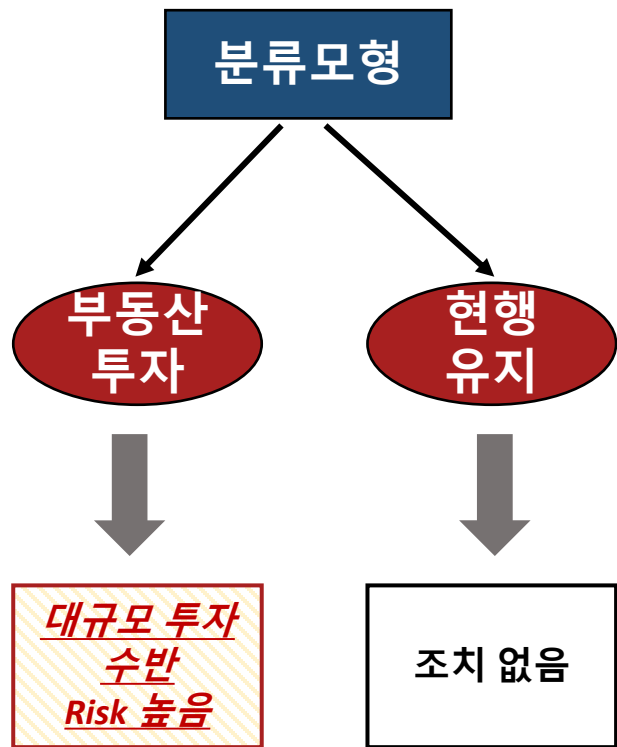
Case#1.



Case#2.



Case#3.



Class 불균형 극복 방안

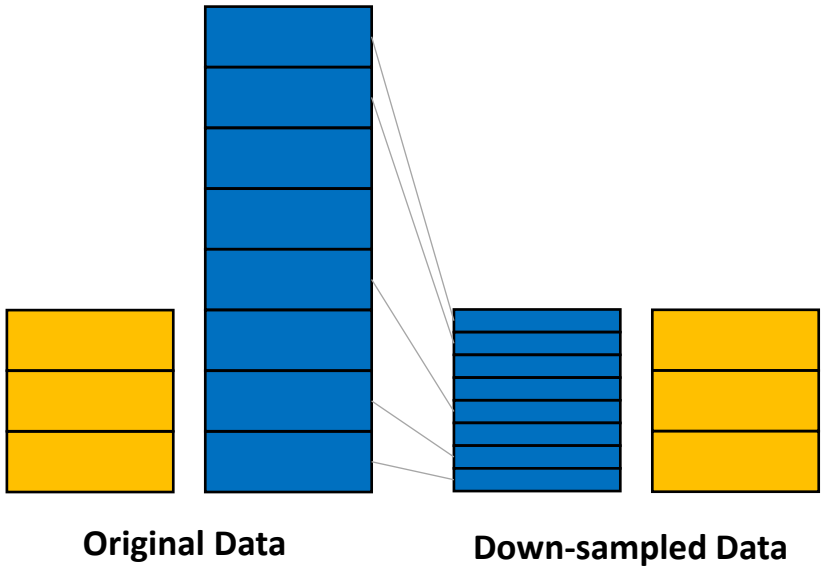
Over-sampling 혹은 Up-sampling	<ul style="list-style-type: none"> 클래스가 적은 쪽의 데이터를 많은 쪽의 데이터에 상응하게 샘플링 하는 방법
Under-sampling 혹은 Down-sampling	<ul style="list-style-type: none"> 클래스가 많은 쪽의 데이터를 적은 쪽의 데이터에 상응하게 샘플링 하는 방법
SMOTE	<ul style="list-style-type: none"> 클래스가 적은 쪽 데이터의 분포 특성에 따라 새로운 데이터를 생성해내는 방법
Weight Balancing	<ul style="list-style-type: none"> 특정 클래스로 분류될 때, 더 큰 Penalty를 줘서 “잔머리”를 굴리지 않도록 하는 방법

분포는 불균형 하더라도
어느 정도의 데이터
사이즈가 확보되어야 함

오버/언더샘플링조차
힘들 때, 그나마
대안적으로 활용가능

오버샘플링(업샘플링) & 언더샘플링(다운샘플링)

언더샘플링(다운샘플링)



오버샘플링(업샘플링)

