범주형자료분석팀

2팀
정희철
김민서
이주형
심수현

INDEX

1

혼동행렬

혼동행렬

예측값과 **실제값**을 비교하기 위한 행렬

분류모델의 성능을 평가하는 지표

		관측값(Y)	
		$Y = 1$	$Y = 0$
예측값($\hat{Y}$)	$\hat{Y} = 1$	TP	FP
	$\hat{Y} = 0$	FN	TN

혼동행렬

		관측값(Y)	
		$Y = 1$	$Y = 0$
예측값($\hat{Y}$)	$\hat{Y} = 1$	TP	FP (1종 오류)
	$\hat{Y} = 0$	FN (2종 오류)	TN

TP(true positive): **맞다고 예측**($\hat{Y} = 1$)한 것 중 **실제로 맞는**($Y = 1$) 경우

혼동행렬

		관측값(Y)	
		$Y = 1$	$Y = 0$
예측값($\hat{Y}$)	$\hat{Y} = 1$	TP	FP (1종 오류)
	$\hat{Y} = 0$	FN (2종 오류)	TN

TN(true negative): **틀렸다고 예측**($\hat{Y} = 0$)한 것 중 **실제로 틀린**($Y = 0$) 경우

혼동행렬

		관측값(Y)	
		$Y = 1$	$Y = 0$
예측값($\hat{Y}$)	$\hat{Y} = 1$	TP	FP (1종 오류)
	$\hat{Y} = 0$	FN (2종 오류)	TN

FP(false positive): 맞다고 예측($\hat{Y} = 1$)했지만 실제로 틀린($Y = 0$) 경우

혼동행렬

		관측값(Y)	
		$Y = 1$	$Y = 0$
예측값($\hat{Y}$)	$\hat{Y} = 1$	TP	FP (1종 오류)
	$\hat{Y} = 0$	FN (2종 오류)	TN

FN(false negative): 틀렸다고 예측($\hat{Y} = 0$)했지만 실제로 맞은($Y = 1$) 경우

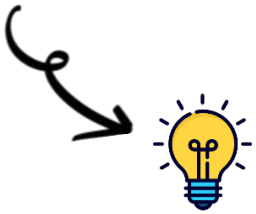
혼동행렬의 한계점

① 정보의 손실 발생

로지스틱 회귀모형: **연속적인 값**으로 예측확률 반환

VS

예측: cut-off point를 기준으로 **예측값을 0 또는 1로 범주화**



연속적인 값이 이항값으로 묶여 **정보의 손실** 발생!

혼동행렬의 한계점

② 임의적인 cut-off point 설정

대개 cut-off point를 0.5로 지정하여 분류 진행

BUT **cut-off point의 임의적 지정**은 **분석의 객관성을 떨어뜨림**

이를 해결하기 위해 **ROC 곡선** 사용!

분류 평가지표

정확도 (Accuracy)

전체 경우에서 **실제값과 예측값이 일치**하는 비율

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

		관측값(Y)	
		$Y = 1$	$Y = 0$
예측값($\hat{Y}$)	$\hat{Y} = 1$	TP	FP
	$\hat{Y} = 0$	FN	TN

1 혼동행렬

분류 평가지표

정확도 (Accuracy)

전체 경우에서 **실제값과 예측값이 일치**하는 비율

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

얼마나 정확한지를 보는 지표 → 정확도가 **1에 가까울수록** 좋은 모형!

장점: 직관적으로 이해하기 쉽고 단순

단점: 데이터가 특정 범주에 쏠린 unbalanced일 때 해당 범주에 지나치게 의존적

1 혼동행렬

분류 평가지표

정확도 (Accuracy)

전체 경우에서 **실제값과 예측값이 일치**하는 비율

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

얼마나 정확한지를 보는 지표 → 정확도가 **1에 가까울수록** 좋은 모형!

장점: 직관적으로 이해하기 쉽고 단순

단점: 데이터가 특정 범주에 쏠린 unbalanced일 때 **해당 범주에 지나치게 의존적**

분류 평가지표

정밀도 (Precision)

맞다고 예측한 것 중 실제로 맞는 것의 비율

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

		관측값(Y)	
		$Y = 1$	$Y = 0$
예측값($\hat{Y}$)	$\hat{Y} = 1$	TP	FP
	$\hat{Y} = 0$	FN	TN

분류 평가지표

정밀도 (Precision)

맞다고 예측한 것 중 실제로 맞는 것의 비율

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

$\hat{Y} = 1$이라고 말했던 것 중 실제로 $Y = 1$인 경우의 비율

Unbalanced data일 때 특정 범주에 대한 의존성이 줄어듦

FP가 더 critical한 경우 사용

ex) 재판에서 결백한 사람을 유죄로 판결하는 경우

분류 평가지표

정밀도 (Precision)

맞다고 예측한 것 중 실제로 맞는 것의 비율

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

$\hat{Y} = 1$이라고 말했던 것 중 실제로 $Y = 1$인 경우의 비율

Unbalanced data일 때 특정 범주에 대한 의존성이 줄어듦

FP가 더 critical한 경우 사용

ex) 재판에서 결백한 사람을 유죄로 판결하는 경우

분류 평가지표

민감도 (Sensitivity) / 재현율 (Recall)

실제로 맞는 것 중에서 **맞다고 예측한 것**의 비율

$$Sensitivity(recall) = \frac{TP}{TP + FN}$$

		관측값(Y)	
		$Y = 1$	$Y = 0$
예측값($\hat{Y}$)	$\hat{Y} = 1$	TP	FP
	$\hat{Y} = 0$	FN	TN

분류 평가지표

민감도 (Sensitivity) / 재현율 (Recall)

실제로 맞는 것 중에서 **맞다고 예측한 것**의 비율

$$Sensitivity(recall) = \frac{TP}{TP + FN}$$

실제로 $Y = 1$ 인 것 중 $\hat{Y} = 1$이라고 예측한 것의 비율

민감도가 높음 = 맞는 것을 잘 맞춤 → 1에 가까울수록 좋음

Unbalanced data일 때 특정 범주에 대한 의존성이 줄어듦

FN이 더 critical한 경우 사용

ex) 코로나에 걸린 사람을 음성이라고 판단하는 것

1 혼동행렬

분류 평가지표

민감도 (Sensitivity) / 재현율 (Recall)

실제로 맞는 것 중에서 **맞다고 예측한 것**의 비율

$$Sensitivity(recall) = \frac{TP}{TP + FN}$$

실제로 $Y = 1$ 인 것 중 $\hat{Y} = 1$이라고 예측한 것의 비율

민감도가 높음 = 맞는 것을 잘 맞춤 → **1에 가까울수록 좋음**

Unbalanced data일 때 특정 범주에 대한 의존성이 줄어듦

FN이 더 critical한 경우 사용

ex) 코로나에 걸린 사람을 음성이라고 판단하는 것

분류 평가지표

특이도 (Specificity)

실제로 부정인 것 중에서 **부정으로 예측**한 것의 비율

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

		관측값(Y)	
		$Y = 1$	$Y = 0$
예측값($\hat{Y}$)	$\hat{Y} = 1$	TP	FP
	$\hat{Y} = 0$	FN	TN

분류 평가지표

특이도 (Specificity)

실제로 부정인 것 중에서 **부정으로 예측**한 것의 비율

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

부정인 것을 잘 맞춘 정도

$$FPR = \frac{FP}{TN + FP} = 1 - Specificity$$

↘ 특이도는 1에 가까울수록, FPR은 0에 가까울수록 모델의 성능이 좋다고 판단

분류 평가지표

F1-Score

정밀도와 재현도의 조화평균

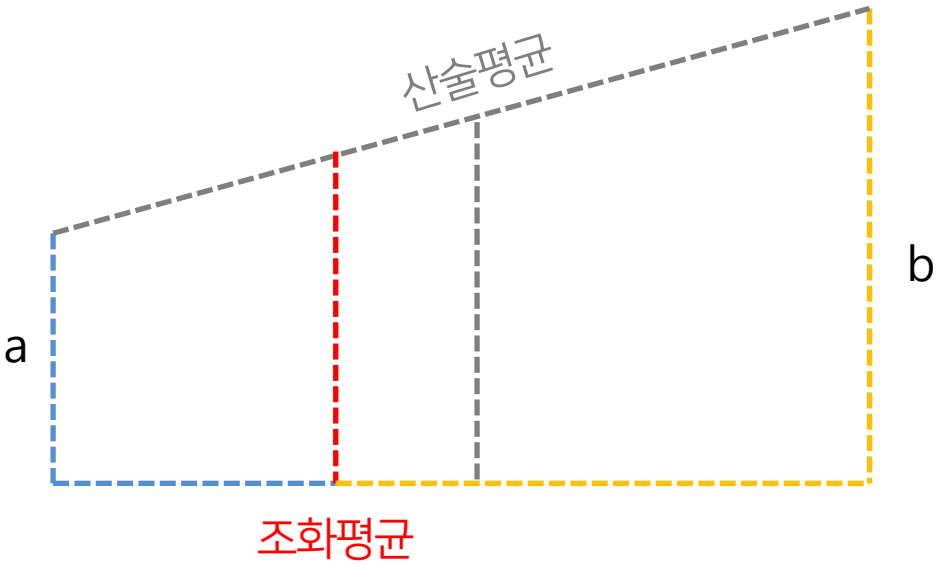
$$F1\ Score = \frac{2}{\frac{1}{precision} + \frac{1}{recall}} = 2 \times \frac{precision \times recall}{precision + recall} = \frac{2TP}{2TP + FN + FP}$$

조화평균 사용 → 데이터가 **불균형**한 경우에도 성능을 정확히 평가

F1-Score값이 1에 가까울수록 모델의 성능이 우수하다고 판단

분류 평가지표

조화평균



큰 값을 가지는 수치에 **패널티**를 주어 **작은 값에 가까운 평균을 산출**

분류 평가지표

F1-Score

정밀도와 재현도의 조화평균

조화평균 사용이유

① 불균형 데이터에서 **관측값이 많은 클래스에 패널티 부여**

→ 해당 클래스에 대한 **의존성 감소**

정확도가 불균형 데이터에 갖는 한계를 보완한 지표

분류 평가지표

F1-Score

정밀도와 재현도의 조화평균

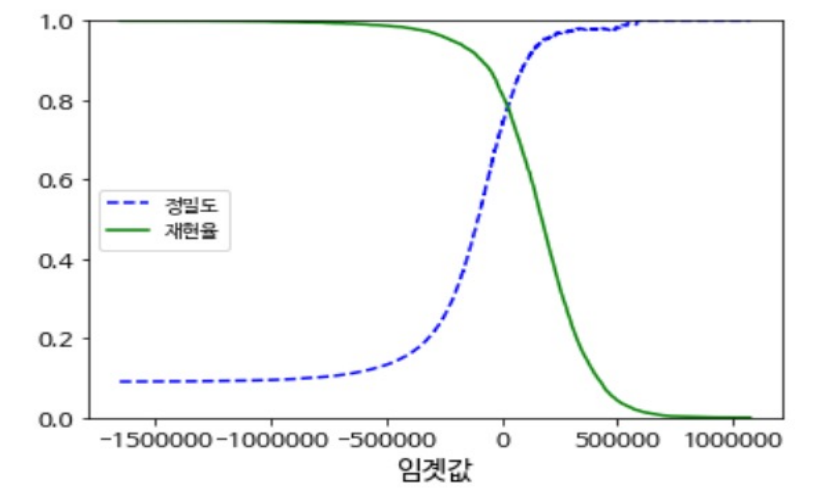
조화평균 사용이유	② 정밀도와 재현도를 모두 균형 있게 반영

1 혼동행렬

정밀도와 재현도의 상충관계 (Trade-off)

분류 평가지표

F1-Score



Ex) 임계값이 낮아지면 positive로 예측하는 값이 많아짐

조화평균 사용이유 = 재현도의 분모를 구성하는 FN이 낮아지면 재현율 ↑

= 정밀도의 분모를 구성하는 FP가 높아지면 정밀도 ↓

정밀도가 커지면 재현도 ↓ ,재현도가 커지면 정밀도 ↓

1 혼동행렬

분류 평가지표

F1-Score

정밀도와 재현도의 조화평균

F1-Score 한계

TN(True Negative) 고려 X

↓

Negative로 올바르게 예측한 값이 많아지더라도 반영 X

분류 평가지표

MCC (매튜상관계수/파이계수)

$$MCC = \frac{(TP \times TN) - (FP \times FN)}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$$

혼동행렬의 모든 부분 사용 → **가장 균형 잡힌 척도**

상관계수 형식 → **-1 ~ 1사이의 값**

1에 가까울 때: **완전예측**

-1에 가까울 때: **완전 역예측** (=완전예측)

0에 가까울 때: **랜덤 예측**

1 혼동행렬

분류 평가지표

MCC (매튜상관계수/파이계수)

$$MCC = \frac{(TP \times TN) - (FP \times FN)}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$$

혼동행렬의 모든 부분 사용 → **가장 균형 잡힌 척도**

상관계수 형식 → **-1 ~ 1사이의 값**

1에 가까울 때: **완전예측**

-1에 가까울 때: **완전 역예측** (=완전예측)

0에 가까울 때: **랜덤 예측**

1 혼동행렬

F1 Score vs MCC

		관측값(Y)	
		$Y = 1$	$Y = 0$
예측값($\hat{Y}$)	$\hat{Y} = 1$	92	4
	$\hat{Y} = 0$	3	1

		관측값(Y)	
		$Y = 1$	$Y = 0$
예측값($\hat{Y}$)	$\hat{Y} = 1$	1	3
	$\hat{Y} = 0$	4	92

1 혼동행렬

F1 Score vs MCC

혼동행렬	F1-Score	MCC
왼쪽	$\frac{2 * 92}{2 * 92 + 4 + 3} = 0.96$	$\frac{(92 * 1) - (4 * 3)}{\sqrt{(92 + 4)(92 + 3)(1 + 4)(1 + 3)}} = 0.18$
오른쪽	$\frac{2 * 1}{2 * 1 + 3 + 4} = 0.22$	$\frac{(92 * 1) - (4 * 3)}{\sqrt{(92 + 4)(92 + 3)(1 + 4)(1 + 3)}} = 0.18$

F1-Score 차이 :

0.96-0.22=0.74

1 혼동행렬

F1 Score vs MCC

혼동행렬	F1-Score	MCC
왼쪽	$\frac{2*92}{2*92+4+3} = 0.96$	$\frac{(92*1)-(4*3)}{\sqrt{(92+4)(92+3)(1+4)(1+3)}} = 0.18$
오른쪽	$\frac{2*1}{2*1+3+4} = 0.22$	$\frac{(92*1)-(4*3)}{\sqrt{(92+4)(92+3)(1+4)(1+3)}} = 0.18$

F1-Score 차이 :
0.96-0.22=0.74

MCC 차이:
0.18-0.18=0
→ 동일한 값

1 혼동행렬

F1 Score vs MCC

혼동행렬	F1-Score	MCC
왼쪽	$\frac{2 * 92}{2 * 92 + 4 + 3} = 0.96$	$\frac{(92 * 1) - (4 * 3)}{\sqrt{(92 + 4)(92 + 3)(1 + 4)(1 + 3)}} = 0.18$
오른쪽	$\frac{2 * 1}{2 * 1 + 3 + 4} = 0.22$	$\frac{(92 * 1) - (4 * 3)}{\sqrt{(92 + 4)(92 + 3)(1 + 4)(1 + 3)}} = 0.18$

F1 Score는 TN 사용 X → **TN값의 차이가 크면 F1 Score는 큰 차이**를 보임

MCC은 모든 성분 사용 → 값의 **차이 없음**

분석의 목적에 따른 평가지표 선택

F1 Score vs MCC

혼동행렬	F1-Score	MCC
왼쪽	$\frac{2*92}{2*92+4+3} = 0.96$	$\frac{(92*1)-(4*3)}{\sqrt{(92+4)(92+3)(1+4)(1+3)}} = 0.18$
오른쪽	$\frac{2*1}{2*1+3+4} = 0.22$	$\frac{(92*1)-(4*3)}{\sqrt{(92+4)(92+3)(1+4)(1+3)}} = 0.18$

클래스에 대한 균형적 평가

희귀질환처럼 중요하지만 관측치가 적은 경우

MCC

F1 Score는 TN 사용 X → TN값의 차이가 크면 F1 Score는 큰 차이를 보임

MCC은 모든 성분 사용 → 값의 차이가 없음

해당 클래스를 Positive로 두고

F1 Score

2

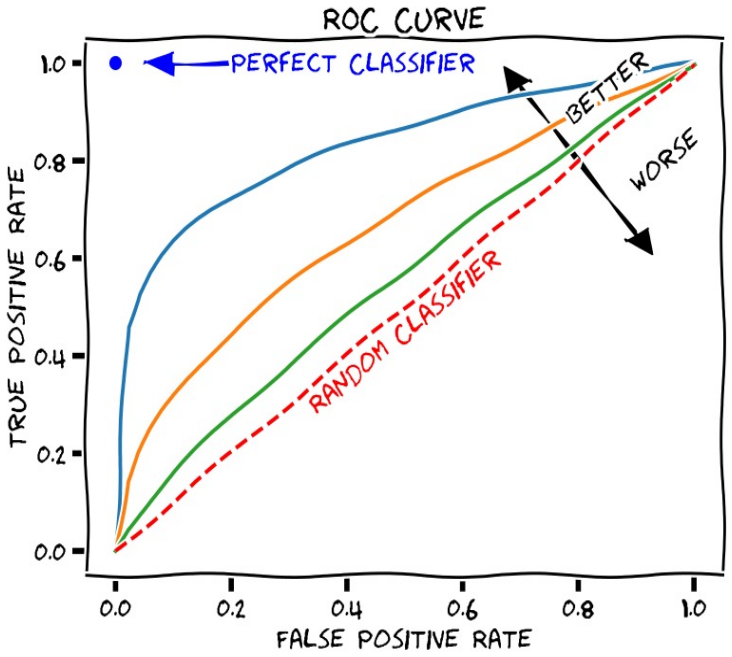
ROC 곡선

2 ROC 곡선

ROC 곡선

가능한 모든 cut-off point에 대하여

재현율을 (1-특이도)의 함수로 나타낸 곡선



모든 cut-off point에 대한

혼동행렬을 구하고,

이를 통해 구한 재현율과 1-특이도 값을

2차원 상의 점으로 찍어 이를 연결한 형태

2 ROC 곡선

ROC 곡선

가능한 모든 cut-off point에 대하여

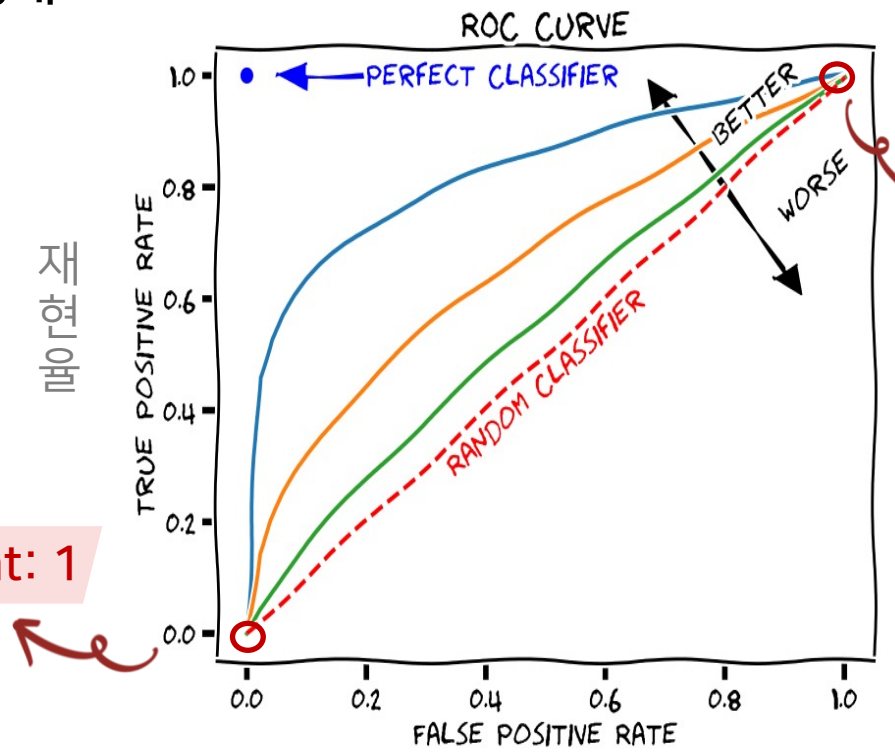
재현율을 (1-특이도)의 함수로 나타낸 곡선

① **혼동행렬**보다 더 많은 정보

② 주어진 모형에서 **가장 적합한** cut-off point를 찾을 수 있음

2 ROC 곡선

ROC 곡선의 형태



재현율

1-특이도

cut-off point: 0

cut-off point: 1

(0,0)에서 (1,1)을 이어주며 **위로 볼록한** 우상향 형태

2 ROC 곡선

ROC 곡선의 형태

cut-off point: 0

cut-off point: 1

Cut-off point값이 **1에 가까울수록** (0,0)에
0에 가까울수록 (1,1)에 가까워지기 때문

(0,0)에서 (1,1)을 이어주며 **위로 볼록한 우상향** 형태

2 ROC 곡선

ROC 곡선의 형태

Cut-off point가 **0에 가까운** 값

↓

대부분의 관측치를

Y=1(positive)로 예측

↓

TP, FP **증가** & TN, FN **감소**

↓

TPR&FPR 모두 **1에 가까운** 값

Cut-off point가 **1에 가까운** 값

↓

대부분의 관측치를

Y=0(negative)로 예측

↓

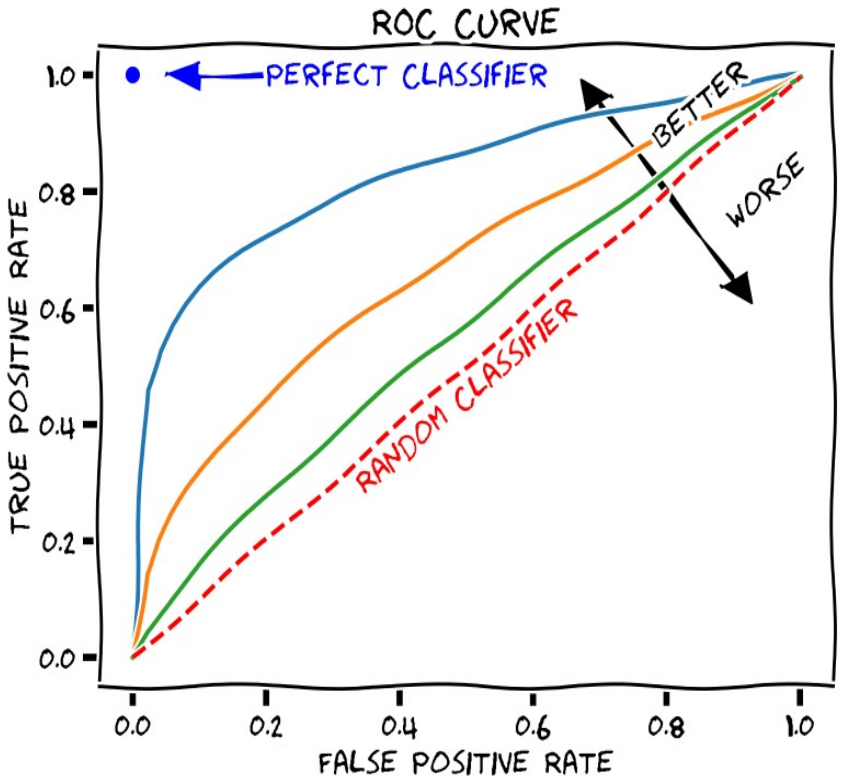
TP, FP **감소** & TN, FN **증가**

↓

TPR&FPR 모두 **0에 가까운** 값

2 ROC 곡선

ROC 곡선 해석



TPR은 1에 가까울수록,

FPR은 0에 가까울수록

성능이 우수함

↓

같은 Y값 → X값이 작을수록

같은 X값 → Y값이 클수록

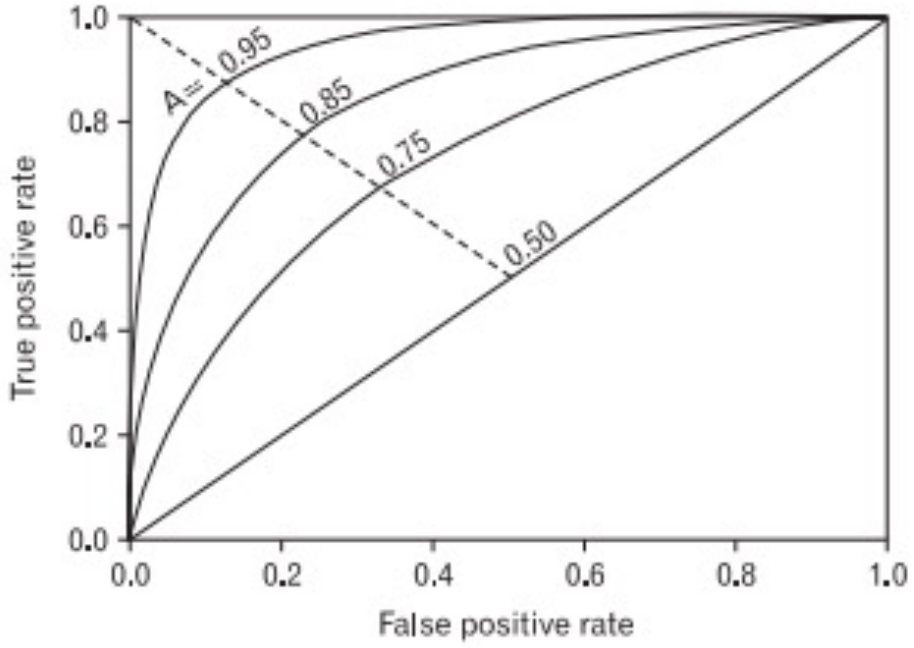
좋은 cut-off point!

2 ROC 곡선

AUC(Area Under the Curve)

AUC

ROC 곡선 밑의 면적으로, 0 ~ 1의 값 가짐



2 ROC 곡선

AUC(Area Under the Curve)

AUC

ROC 곡선 밑의 면적으로, 0 ~ 1의 값 가짐

AUC=1 : 100% 완벽하게 예측 / 모델이 과적합되었을 가능성 고려

AUC=0.5 : 50%만 맞춘 예측 / 랜덤하게 예측한 결과

보통 0.5 이상의 값을 가져야 정상

(0.5 미만의 값을 갖는 경우 → 분류군을 반대로 처리한 경우)

AUC=0 : 100% 반대로 예측

(완벽히 틀린 것은 완벽히 맞는 것과 같은 의미 → AUC=1과 AUC=0은 같음)

2 ROC 곡선

AUC(Area Under the Curve)

AUC

ROC 곡선 밑의 면적으로, 0 ~ 1의 값 가짐

AUC=1 : 100% 완벽하게 예측 / 모델이 과적합되었을 가능성 고려

AUC=0.5 : 50%만 맞춘 예측 / 랜덤하게 예측한 결과

보통 0.5 이상의 값을 가져야 정상

(0.5 미만의 값을 갖는 경우 → 분류군을 반대로 처리한 경우)

AUC=0 : 100% 반대로 예측

(완벽히 틀린 것은 완벽히 맞는 것과 같은 의미 → AUC=1과 AUC=0은 같음)

2 ROC 곡선

AUC(Area Under the Curve)

AUC

ROC 곡선 밑의 면적으로, 0 ~ 1의 값 가짐

AUC=1 : 100% 완벽하게 예측 / 모델이 과적합되었을 가능성 고려

AUC=0.5 : 50%만 맞춘 예측 / 랜덤하게 예측한 결과

보통 0.5 이상의 값을 가져야 정상

(0.5 미만의 값을 갖는 경우 → 분류군을 반대로 처리한 경우)

AUC=0 : 100% 반대로 예측

(완벽히 틀린 것은 완벽히 맞는 것과 같은 의미 → AUC=1과 AUC=0은 같음)

3

샘플링

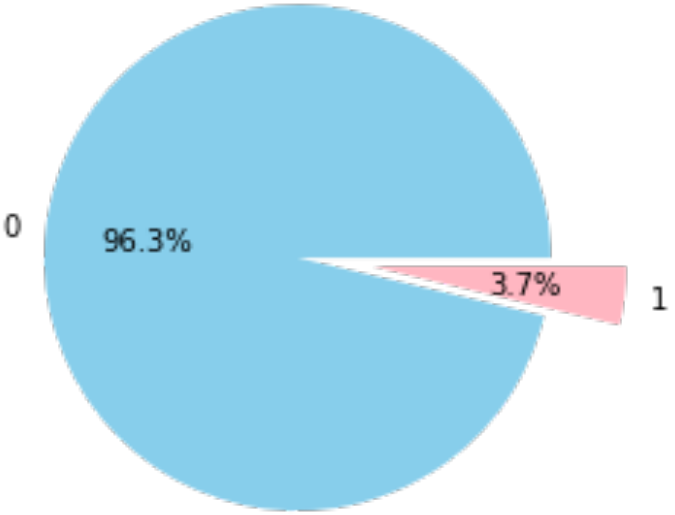
3 샘플링

클래스 불균형

클래스 불균형

어떤 범주형 변수의 각 수준이 가지고 있는

데이터의 양에 큰 차이가 있는 경우



→

수준 0과 수준 1의 데이터 구성비가

96.3 : 3.7로 매우 큰 차이를 보임

=> **클래스 불균형 !**

3 샘플링

클래스 균형의 필요성

Y변수의 클래스 비율의 차이가 크다면?

↓

우세한 클래스만으로 예측하여도

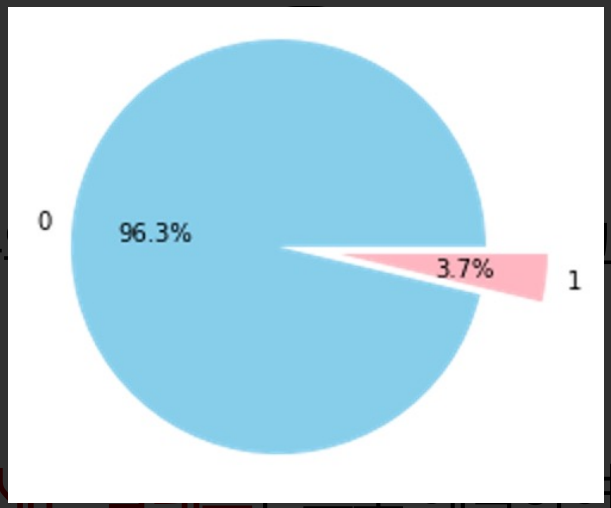
정확도 자체는 높게 나타남

↓

모델의 성능을 판별하기 어려움 !

왜 모델 성능 판별이 어려울까?

클래스 균형의 필요성



Y변수 ... 다면?

우... 도

정확도 자체는 높게 나타남

모든 예측치를 0 클래스로 예측시 96.3%의 높은 정확도를 가지는 모델이지만

소수 클래스의 재현율이 매우 낮아지기 때문 !

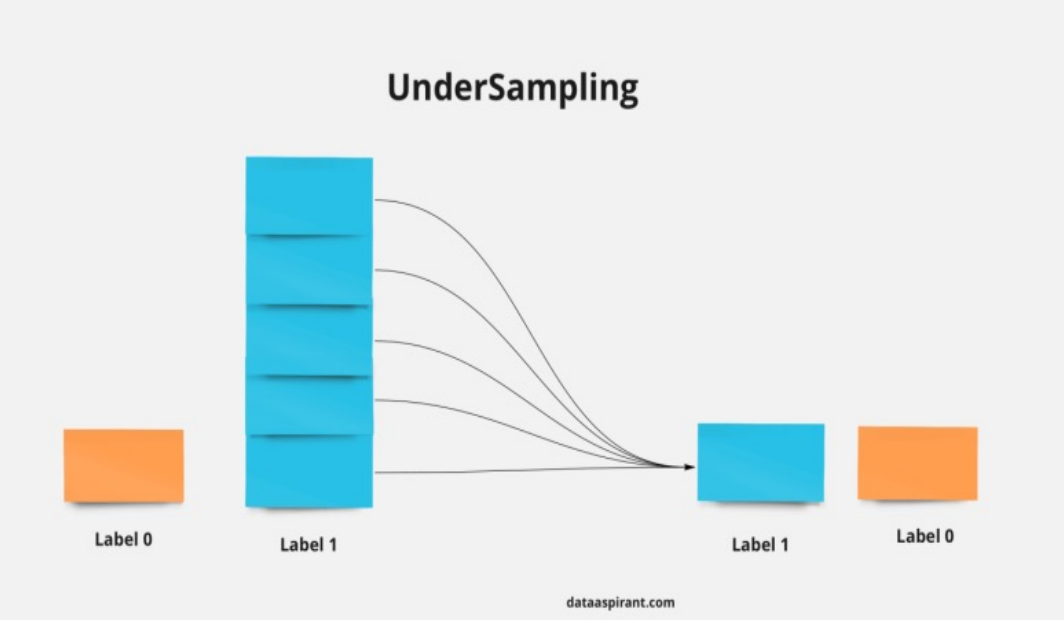
=> 샘플링을 통해 해결 가능 !

언더 샘플링

언더 샘플링

소수의 클래스는 변형하지 않고

다수의 클래스를 소수의 클래스에 맞춰 관측치를 감소시키는 방법



언더 샘플링 - 특징

언더 샘플링

소수의 클래스는 변형하지 않고

다수의 클래스를 소수의 클래스에 맞춰 **관측치를 감소**시키는 방법

✓ **장점**

데이터 사이즈가 줄어

메모리 및 처리속도면에서 **유리**

✓ **단점**

관측치가 손실되므로

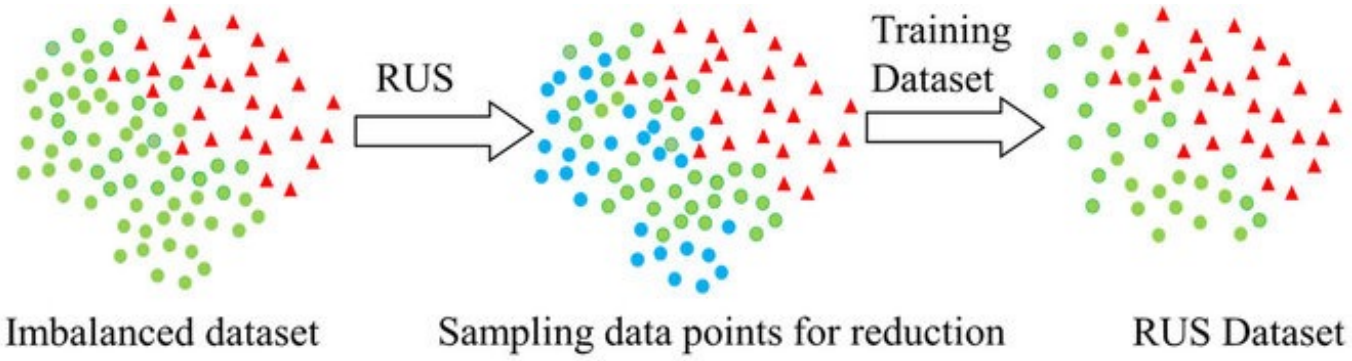
정보 누락 문제 발생

언더 샘플링 - 기법

Random Under Sampling

임의적으로 다수의 클래스의 **데이터를 제거**하여

관측치 수를 **감소**시키는 방법



언더 샘플링 - 기법

Random Under Sampling

임의적으로 다수의 클래스의 **데이터를 제거**하여

관측치 수를 **감소**시키는 방법

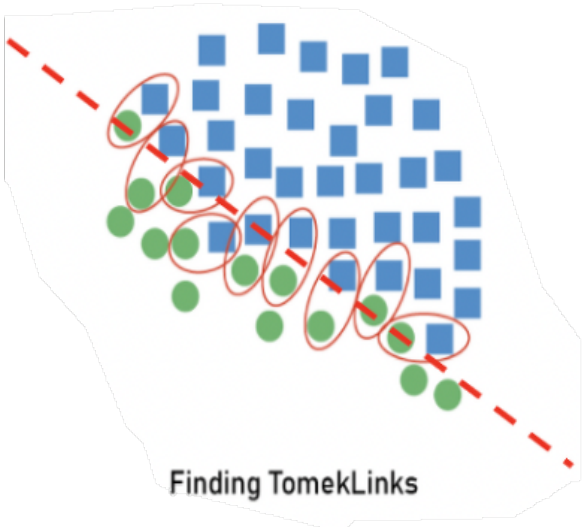
임의적으로 제거한 **데이터의 정보가 누락**

- > 기존 데이터의 **대표성**을 띄지 못할 가능성이 많음

언더 샘플링 - 기법

Tomek Links Method

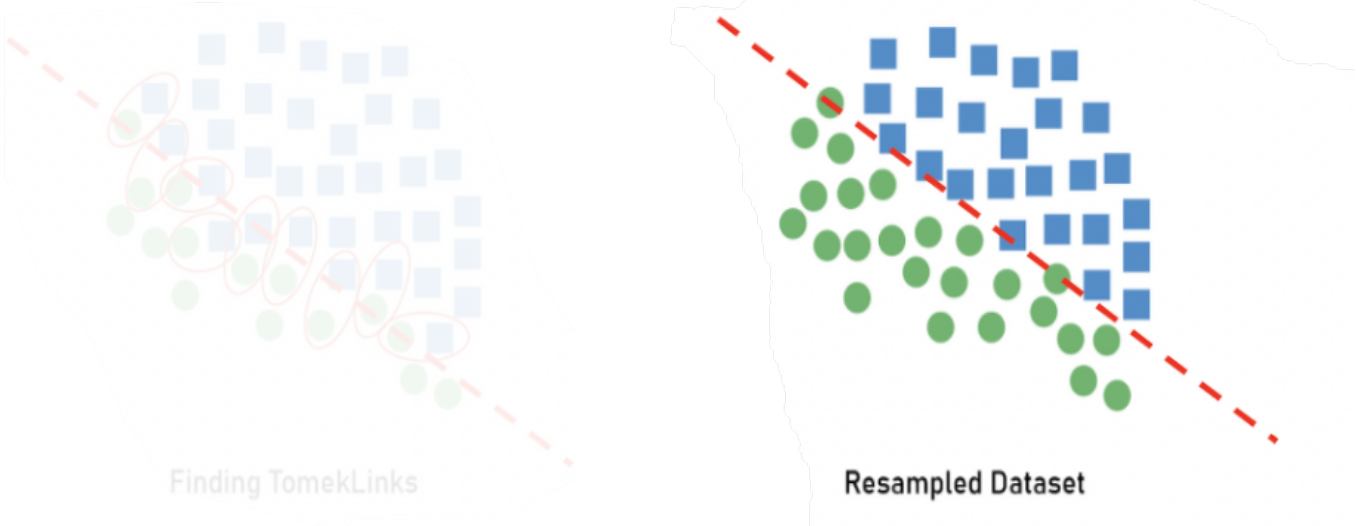
① 임의로 **서로 다른 클래스**의 데이터의 **두 점**을 선택하여 **연결**



언더 샘플링 - 기법

Tomek Links Method

② 연결된 쌍에서 **다수 클래스**에 속하는 관측치 **제거**



언더 샘플링 - 기법

Tomek Links Method

② 연결된 쌍에서 **다수 클래스**에 속하는 관측치 **제거**

경계 주위의 정보를 잃어버릴 수 있음

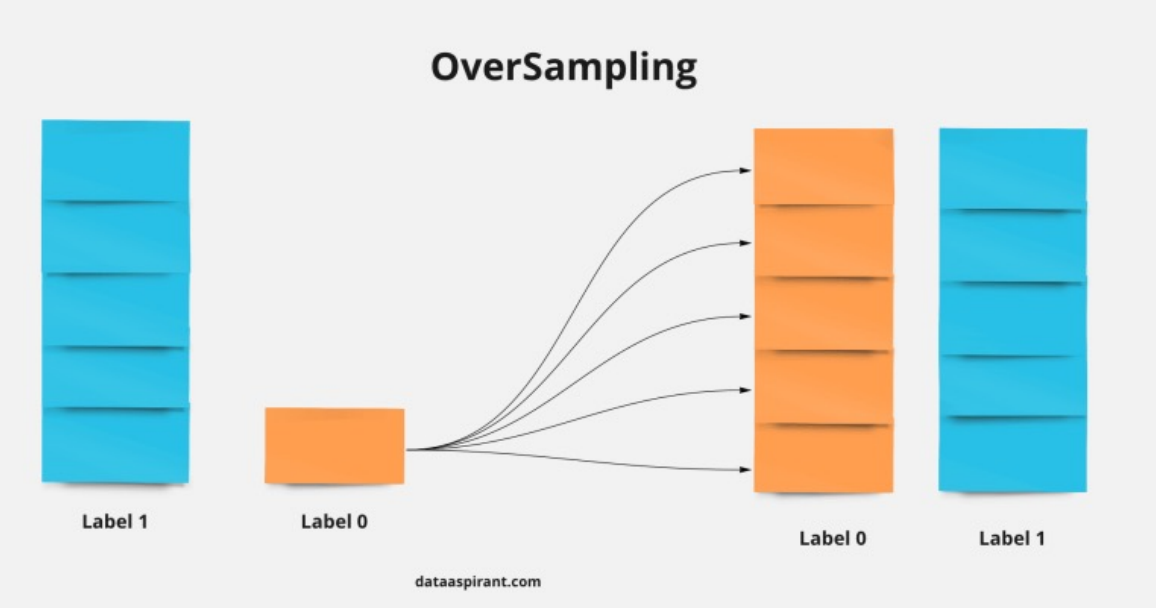
클래스 불균형이 심할수록 효과가 없음

오버 샘플링

오버 샘플링

다수의 클래스는 변형하지 않고

소수의 클래스를 다수의 클래스에 맞춰 **관측치를 증가**시키는 방법



오버 샘플링 - 특징

오버 샘플링

다수의 클래스는 변형하지 않고

소수의 클래스를 다수의 클래스에 맞춰 **관측치를 증가**시키는 방법

✓ **장점**

데이터의 손실이 없기 때문에

언더 샘플링보다 **성능이 좋음**

✓ **단점**

데이터의 사이즈가 커져

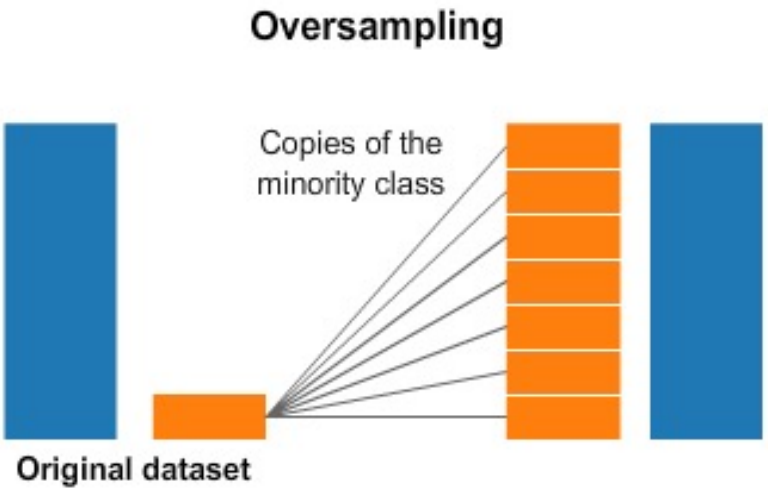
메모리나 처리속도면에서 **불리**

오버 샘플링 - 기법

Random Over Sampling

임의적으로 소수의 클래스의 **데이터를 복제**하여

관측치 수를 **증가**시키는 방법



오버 샘플링 - 기법

Random Over Sampling

임의적으로 소수의 클래스의 **데이터를 복제**하여

관측치 수를 **증가**시키는 방법

임의로 소수 클래스의 데이터를 복제하여 관측치 증가

-> 동일한 데이터의 수가 늘어나 **과적합** 발생 가능성이 큼 !

오버 샘플링 - 기법

SMOTE

① 소수의 클래스 중 임의로 관측치 하나를 선택

오버 샘플링 - 기법

SMOTE

② 선택된 관측치를 기준으로 KNN 알고리즘을 이용하여

가장 가까운 K개 관측치를 선택

오버 샘플링 - 기법

SMOTE

③ 선택된 관측치와 선정된 K개의 관측치 사이에 **직선**을 그려 직선 상의 **가상의 소수 클래스 데이터** 생성

오버 샘플링 - 기법

SMOTE

③ 선택된 관측치와 선정된 K개의 관측치 사이에 **직선**을 그려
직선 상의 **가상의 소수 클래스 데이터** 생성

가상의 데이터를 생성하기 때문에
Random Over Sampling보다 **과적합 발생 가능성이 적음 !**

오버 샘플링 - 기법

SMOTE

③ 선택된 관측치와 선정된 K개의 관측치 사이에 **직선**을 그려
직선 상의 **가상의 소수 클래스 데이터** 생성

데이터 생성 과정서 다수 클래스 데이터 고려 X
-> 생성된 데이터가 기존 데이터와 겹치거나 노이즈 발생 가능
=> **고차원 데이터**일 경우 **부적합** !

4

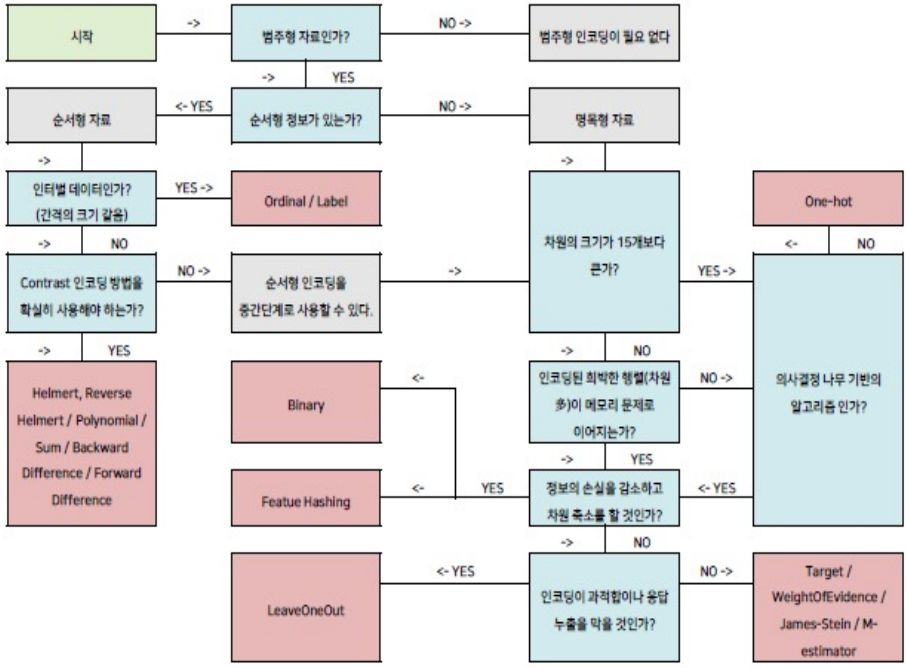
인코딩

인코딩

인코딩 (Encoding)

문자나 기호들을 **컴퓨터가 이용할 수 있는 신호**로 만드는 과정

주어진 데이터의 특징에 따라 방법이 다름

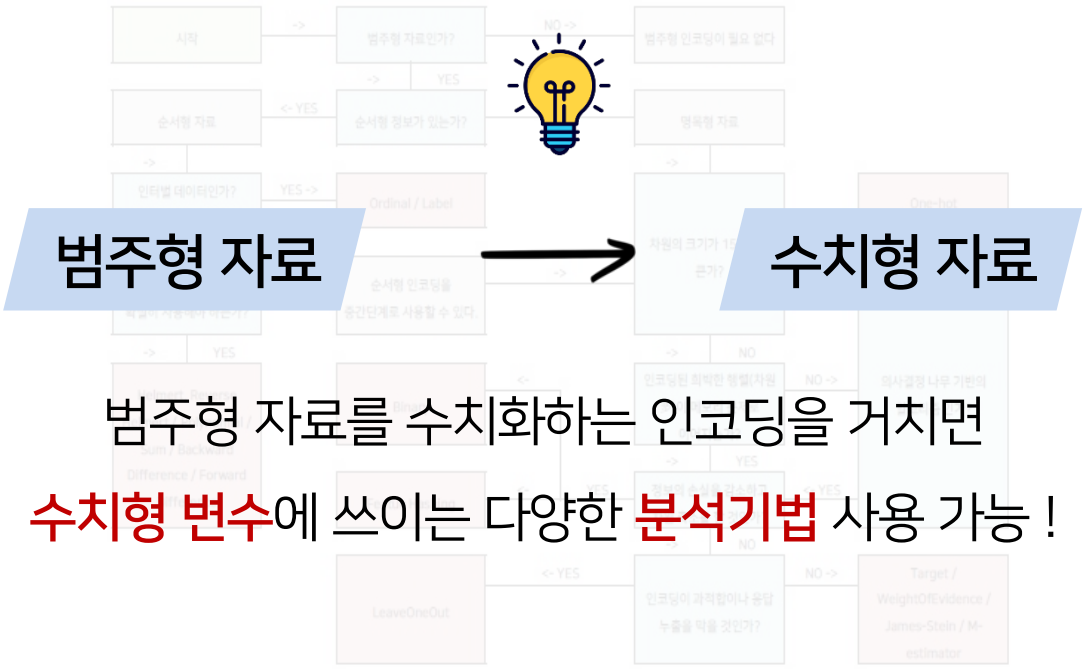


인코딩

인코딩 (Encoding)

문자나 기호들을 **컴퓨터가 이용할 수 있는 신호**로 만드는 과정

주어진 데이터의 특징에 따라 방법이 다름



범주형 자료를 수치화하는 인코딩을 거치면

수치형 변수에 쓰이는 다양한 **분석기법** 사용 가능 !

4 인코딩

인코딩의 종류

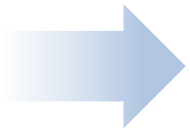
표시한 부분에 대해서 중점적으로 다룰 예정!

Classic	Contrast	Bayesian	기타
Ordinal	Simple	**Mean(Target)**	Frequency
One-Hot	Sum	**Leave One Out**	
Label	Helmert	Weight of Evidence	
Binary	Reverse Helmert	Probability Ratio	
BaseN	Forward Difference	James Stein	
Hashing	Backward Difference	M-estimator	
	Orthogonal Polynomial	**Ordered Target**	

One-Hot Encoding (Dummy Encoding)

가변수(Dummy variable)를 만들어주는 인코딩 방법

계절
봄
여름
가을
겨울



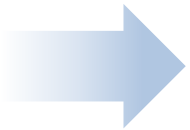
봄	여름	가을	겨울
1	0	0	0
0	1	0	0
0	0	1	0
0	0	0	1

가변수 형성 후 해당 범주에는 1, 그 외는 0 입력

One-Hot Encoding (Dummy Encoding)

가변수(Dummy variable)를 만들어주는 인코딩 방법

계절
봄
여름
가을
겨울



봄	여름	가을	겨울
1	0	0	0
0	1	0	0
0	0	1	0
0	0	0	1

기준이 되는 열(ex: 봄) 삭제 → 모든 가변수가 0이면 기준 범주를 의미하기 때문

4 인코딩

One-Hot Encoding (Dummy Encoding)

가변수(Dummy variable)를 만들어주는 인코딩 방법

계절
봄
여름
가을
겨울

J개의 수준을 갖는 **범주형 변수**는

J-1개의 **가변수**로 충분히 설명 가능!

봄	여름	가을	겨울
1	0	0	0
0	1	0	0
0	0	1	0
0	0	0	1

기준이 되는 열(ex: 봄) 삭제 → 모든 가변수가 0이면 기준 범주를 의미하기 때문

4 인코딩

One-Hot Encoding (Dummy Encoding)

장점

① 기준 범주의 정보가 intercept로 존재 → 해석 용이

② 명목형 변수 값을 잘 반영함

③ 한 변수가 다른 변수로 설명되는 **다중공선성** 해결

회귀팀 3주차 클린업 참고!

단점

범주형 변수의 수준이나 개수가 너무 많을 경우,

데이터의 차원이 늘어나는 문제 발생!

↳ 모델 학습속도 ↓ , 많은 computing power 요구

4 인코딩

One-Hot Encoding (Dummy Encoding)

장점

① 기준 범주의 정보가 intercept로 존재 → 해석 용이

② 명목형 변수 값을 잘 반영함

③ 한 변수가 다른 변수로 설명되는 **다중공선성** 해결

회귀팀 3주차 클린업 참고!

단점

범주형 변수의 수준이나 개수가 너무 많을 경우,

데이터의 차원이 늘어나는 문제 발생!

↳ 모델 학습속도 ↓ , 많은 computing power 요구

4 인코딩

Label Encoding

각 범주를 나누기 위해 **점수를 할당**하는 인코딩 방법

계절	점수
봄	1
여름	2
가을	3
겨울	4

✓ **명목형 자료**에 주로 사용

할당된 점수의 숫자에 **의미/순서/연관성 X**

4 인코딩

Label Encoding

각 범주를 나누기 위해 **점수를 할당**하는 인코딩 방법

계절	점수
봄	1
여름	2
가을	3
겨울	4

1부터 시작할 필요 X

등간격 유지하지 않아도 됨

4 인코딩

Label Encoding

장점 **차원이 늘어나지 않아** 모델이 데이터를 빠르게 학습 가능

단점 할당된 점수에 순서나 연관성이 있다고 판단

→ **정보의 왜곡** 발생 가능!

4 인코딩

Label Encoding

장점

차원이 늘어나지 않아 모델이 데이터를 빠르게 학습 가능

단점

할당된 점수에 순서나 연관성이 있다고 판단

→ **정보의 왜곡** 발생 가능!

4 인코딩

Ordinal Encoding

순서형 자료가 주어졌을 때 사용하는 방식으로,

순서가 있는 각 수준에 대응하는 점수를 할당하는 인코딩 방법

만족도	점수
매우 나쁨	1
나쁨	2
보통	3
좋음	4
매우 좋음	5

할당된 점수의 숫자에

순서와 연관성 존재

4 인코딩

Ordinal Encoding

순서형 자료가 주어졌을 때 사용하는 방식으로,

순서가 있는 각 수준에 대응하는 점수를 할당하는 인코딩 방법

만족도	점수
매우 나쁨	1
나쁨	2
보통	3
좋음	4
매우 좋음	5

만족도의 각 수준에

1부터 순서대로 수치 할당!

4 인코딩

Ordinal Encoding

장점 **차원이 늘어나지 않아** 모델이 데이터를 빠르게 학습 가능

단점 범주 내 **수준 간 정확한 차이** 반영 어려움

Ex) 만족도의 각 수준은 1의 차이가 나지만,
보통과 좋음 수준의 차이는 그 이상일 수도 있음

4 인코딩

Ordinal Encoding

장점

차원이 늘어나지 않아 모델이 데이터를 빠르게 학습 가능

단점

범주 내 **수준 간 정확한 차이** 반영 어려움

Ex) 만족도의 각 수준은 1의 차이가 나지만,

보통과 좋음 수준의 차이는 그 이상일 수도 있음

Target Encoding

One-Hot Encoding	Label Encoding	Ordinal Encoding

각 수준을 구분하는 것이 우선, **값 자체의 의미 X**

Target Encoding

각 수준을 구분할 뿐만 아니라

설명변수 X와 반응변수 Y의 **수치적인 관계를 반영**해서 인코딩

4 인코딩

Target Encoding

One-Hot Encoding	Label Encoding	Ordinal Encoding

각 수준을 구분하는 것이 우선, **값 자체의 의미 X**

Target Encoding

각 수준을 구분할 뿐만 아니라

설명변수 X와 반응변수 Y의 **수치적인 관계를 반영**해서 인코딩

4 인코딩

Mean Encoding

각 수준에서 **반응변수의 평균**으로 수준별로 점수를 할당하는 인코딩 방법

[Y] 토익 점수	[X] 학과	[X] Mean Encoding
780	경영	855
930	경영	855
850	경영	855
870	통계	820
810	통계	820
750	통계	820
660	통계	820
980	경제	863.33
950	경제	863.33
780	경제	863.33

4 인코딩

Mean Encoding

각 수준에서 **반응변수의 평균**으로 수준별로 점수를 할당하는 인코딩 방법

[Y] 토익 점수	[X] 학과	[X] Mean Encoding
780	경영	855
930	경영	855
850	경영	855
870	통계	820
810	통계	820
750	통계	820
660	통계	820
980	경제	863.33
950	경제	863.33
780	경제	863.33

$$\frac{780 + 930 + 850}{3}$$

$= 855$ (경영학과 평균 토익 점수)

4 인코딩

Mean Encoding

장점

① 반응변수와 설명변수 간 관계 고려

→ 할당된 수치가 **당위성** 지님

② 인코딩 후에도 **데이터의 차원이 늘어나지 않음**

단점

① 새로운 수준이 test set에 등장하면 점수를 할당할 수 없음

② 설명변수 인코딩 과정에 **반응변수에 대한 정보 투입**

→ **과적합** 가능성 높음

③ **이상치**에 영향을 많이 받음

4 인코딩

Mean Encoding

장점

① 반응변수와 설명변수 간 관계 고려

→ 할당된 수치가 **당위성** 지님

② 인코딩 후에도 **데이터의 차원이 늘어나지 않음**

단점

① 새로운 수준이 test set에 등장하면 점수를 할당할 수 없음

② 설명변수 인코딩 과정에 **반응변수에 대한 정보 투입**

→ **과적합** 가능성 높음

③ **이상치**에 영향을 많이 받음

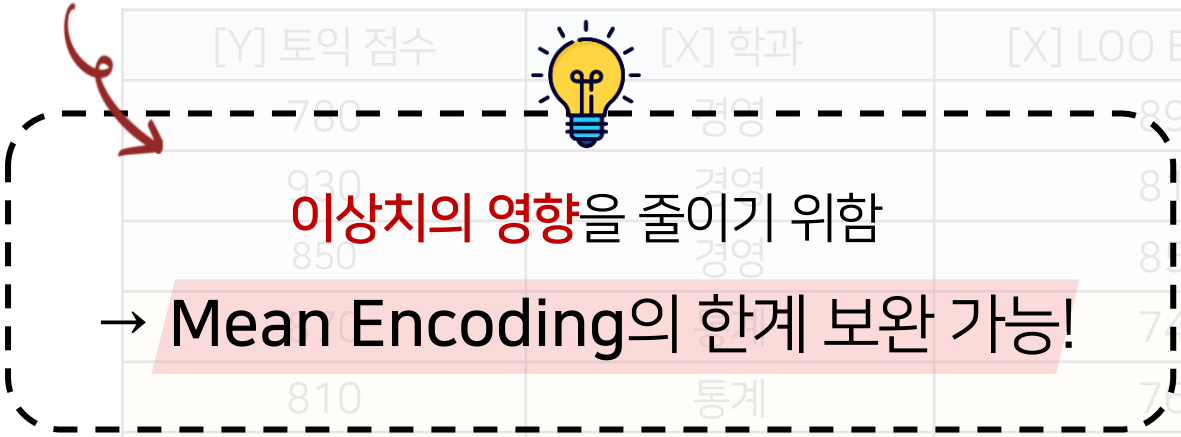
Leave One Out Encoding (LOO Encoding)

현재 행을 제외하고 평균을 구한 후 점수로 할당하는 인코딩 방법

[Y] 토익 점수	[X] 학과	[X] LOO Encoding
780	경영	890
930	경영	815
850	경영	855
870	통계	740
810	통계	760
750	통계	780
660	통계	810
980	경제	865
950	경제	880
780	경제	965

Leave One Out Encoding (LOO Encoding)

현재 행을 제외하고 평균을 구한 후 점수로 할당하는 인코딩 방법



이상치의 영향을 줄이기 위함

→ Mean Encoding의 한계 보완 가능!

[Y] 토익 점수	[X] 학과	[X] LOO Encoding
780	경영	890
930	경영	815
850	경영	815
760	경영	780
810	통계	780
750	통계	780
660	통계	810
980	경제	865
950	경제	880
780	경제	965

Leave One Out Encoding (LOO Encoding)

현재 행을 제외하고 평균을 구한 후 점수로 할당하는 인코딩 방법

[Y] 토익 점수	[X] 학과	[X] LOO Encoding
780	경영	890
930	경영	815
850	경영	855
870	통계	740
810	통계	760
750	통계	780
660	통계	810
980	경제	865
950	경제	880
780	경제	965

Leave One Out Encoding (LOO Encoding)

현재 행을 제외하고 평균을 구한 후 점수로 할당하는 인코딩 방법

[Y] 토익 점수	[X] 학과	[X] LOO Encoding
780	경영	890
930	경영	815
850	경영	855
870	통계	780
810	통계	780
750	통계	780
660	통계	810
980	통계	810
950	경제	880
780	경제	965

$$\frac{780+930}{2} = 855$$

(현재 행을 제외한 경영학과 토익 점수의 평균)

Leave One Out Encoding (LOO Encoding)

장점

① 이상치의 영향을 덜 받음

② **과적합**의 정도나 가능성이 Mean Encoding보다 **낮음**

(모든 반응변수의 정보가 다 반영되지 않기 때문)

단점

Mean Encoding과 동일

4 인코딩

Ordered Target Encoding (CatBoost Encoding)

현재 행 이전의 값들로 평균을 구한 후 점수로 할당하는 인코딩 방법

[Y] 키	[X] 학과	Mean Encoding	Ordered Target Encoding
168	경영	172	169.5
174	통계	166	169.5
165	경제	171.66	169.5
156	통계	166	174
180	경영	172	168
163	통계	166	165
180	경제	171.66	165
170	경제	171.66	172.5
168	경영	172	174
171	통계	166	164.33

4 인코딩

Ordered Target Encoding (CatBoost Encoding)

현재 행 이전의 값들로 평균을 구한 후 점수로 할당하는 인코딩 방법

[Y] 키	[X] 학과	Mean Encoding	Ordered Target Encoding
168	경영	172	169.5
174	통계	166	169.5
165	경제	171.66	169.5
156	통계	166	174
180	경영	172	168
163	통계	166	166
180	경제	171.66	165
170	경제	171.66	172.5
168	경영	172	174
171	통계	166	164.33

각 수준에서 첫 번째로 나타나는 행은

평균을 구할 동일한 수준의 이전 값이 없음

→ **전체 평균으로 할당**

4 인코딩

Ordered Target Encoding (CatBoost Encoding)

현재 행 이전의 값들로 평균을 구한 후 점수로 할당하는 인코딩 방법

[Y] 키	[X] 학과	Mean Encoding	Ordered Target Encoding
168	경영	172	169.5
174	통계	166	169.5
165	경제	171.66	171.66
156	통계	166	165
180	경영	172	171
163	통계	166	165
180	경제	171.66	165
170	경제	171.66	172.5
168	경영	172	174
171	통계	166	164.33

앞선 두 통계학과 학우들의 키의 평균으로 할당

Ordered Target Encoding (CatBoost Encoding)

현재 행 이전의 값들로 평균을 구한 후 점수로 할당하는 인코딩 방법

Mean Encoding과 비교해 보면,

범주 내 **같은 수준**에 속해도

전혀 **다른 인코딩 값** 가질 수 있음!

[Y] 키	[X] 학과	Mean Encoding	Ordered Target Encoding
168	경영	172	169.5
172	경제	166	169.5
165	경제	171.66	169.5
156	통계	166	174
180	경영	172	168
163	통계	166	165
180	경제	171.66	165
170	경제	171.66	172.5
168	경영	172	174
171	통계	166	164.33

5

대응짝 검정

5 대응짝 검정

대응짝

대응짝

하나의 subject가 **2개 이상의 값**을 가지는 데이터

비교를 위해 많이 쓰이는 방법

	효과있음	효과없음	합계
Drug	61	25	86
Placebo	22	64	86

같은 환자가 약과 플라시보 모두 복용

↓

실제 약을 복용한 환자 수와

플라시보를 복용한 환자 수가 **동일**

5 대응짝 검정

대응짝

대응짝

하나의 subject가 **2개 이상의 값**을 가지는 데이터

비교를 위해 많이 쓰이는 방법

	효과있음	효과없음	합계
Drug	61	25	86
Placebo	22	64	86

같은 환자가 약과 플라시보 모두 복용

↓

실제 약을 복용한 환자 수와

플라시보를 복용한 환자 수가 **동일**

대응짝

대응짝에는 이전의 독립성 검정 방법 사용 불가 !

하나의 subject가 2개 이상의 값을 가지는 데이터

χ^2, G^2 등의 검정들

비교를 위해 많이 쓰이는 방법

-> 각 subject들이 독립이어야 한다는 가정이 필요 !

	효과있음	효과없음	합계
Drug	61	25	86
Placebo	22	64	86

but, 대응짝 데이터는 하나의 subject에서 2개의 값이 주어짐

-> 독립성 만족 불가

같은 환자가 약과 플라시보 모두 복용

↓

실제 약을 복용한 환자 수

↓

새로운 검정 방법 필요!

맥니마 검정

맥니마 검정 (McNemar's Test)

2×2 분할표에서 사용할 수 있는 대표적인 대응짝 독립 검정 방법

$$H_0 : The\ new\ treatment\ is\ effective$$

$$H_1 : The\ new\ treatment\ is\ not\ effective$$

$$Z^2 = \frac{(n_{12} - n_{21})^2}{n_{12} + n_{21}} \sim \chi^2(1)$$

통계치가 $\chi^2_\alpha(1)$보다 클 경우 귀무가설 기각 가능 !

5 대응짝 검정

맥니마 검정 vs 카파 검정

맥니마 검정

대응짝 데이터를 2×2 혼동행렬로 나타낼 수 있을 때 처치의 효과 비교

카파 검정

대응짝 데이터를 3×3 이상의 혼동행렬로 나타낼 수 있을 때 처치의 효과 비교

카파 검정

카파 검정 (Kappa Statistics)

3×3 이상의 혼동행렬로 대응짝 데이터를 나타낼 수 있을 때 사용하는 대응짝 독립성 검정 방법

		박찬욱		
봉준호		싫음	중간	좋음
	싫음	24	8	13
	중간	8	13	11
	좋음	10	9	64

✓ 두 감독의 취향의 유사도에 관심

-> 카파검정 방법 사용 가능

카파 검정

카파 검정 (Kappa Statistics)

3×3 이상의 혼동행렬로 대응짝 데이터를 나타낼 수 있을 때 사용하는 대응짝 독립성 검정 방법

$$K = \frac{\sum_{i=1}^{I} \pi_{ii} - \sum_{i=1}^{I} \pi_{i+}\pi_{+i}}{1 - \sum_{i=1}^{I} \pi_{i+}\pi_{+i}}$$

귀무가설 하에서 π_{i+} , $\pi_{+i} = 0$ -> $K = 1$

독립성이 만족될 경우 $\sum_{i=1}^{I} \pi_{ii} = \sum_{i=1}^{I} \pi_{i+}\pi_{+i}$ -> $K = 0$

카파 검정

카파 검정 (Kappa Statistics)

3×3 이상의 혼동행렬로 대응짝 데이터를 나타낼 수 있을 때 사용하는 대응짝 독립성 검정 방법

		박찬욱		
봉준호		싫음	중간	좋음
	싫음	24	8	13
	중간	8	13	11
	좋음	10	9	64

K=0에 가까울수록 영화 취향이 다르다

K=1에 가까울수록 영화 취향이 비슷하다

카파 검정

카파 검정 (Kappa Statistics)

3×3 이상의 혼동행렬로 대응짝 데이터를 나타낼 수 있을 때 사용하는 대응짝 독립성 검정 방법

$$100(1-\alpha)\%\ CI = \hat{K} \pm Z_{\alpha} \times S.E.(\hat{K})$$

카파 계수 K를 이용하여 구한 신뢰구간

-> 신뢰구간에 0이 포함되면 귀무 가설 기각

카파 검정

카파 검정 (Kappa Statistics)

3×3 이상의 혼동행렬로 대응짝 데이터를 나타낼 수 있을 때 사용하는 대응짝 독립성 검정 방법

$$100(1-\alpha)\%\ CI = \hat{K} \pm Z_{\alpha} \times S.E.(\hat{K})$$

$$S.E.(\hat{K}) = \sqrt{\frac{P_e^2 + P_e - \frac{\sum_{i=1}^{I} R_i C_i (R_i + C_i)}{N^3}}{(1-P_e)^2 N}}$$

카파 검정

카파 검정 (Kappa Statistics)

3×3 이상의 혼동행렬로 대응짝 데이터를 나타낼 수 있을 때 사용하는 대응짝 독립성 검정 방법

$$100(1-\alpha)\%\ CI = \hat{K} \pm Z_{\alpha} \times S.E.(\hat{K})$$

$$S.E.(\hat{K}) = \sqrt{\frac{P_e^2 + P_e - \frac{\sum_{i=1}^{I} R_i C_i (R_i + C_i)}{N^3}}{(1 - P_e)^2 N}}$$

P_e : 기대 일치 비율

카파 검정

카파 검정 (Kappa Statistics)

3×3 이상의 혼동행렬로 대응짝 데이터를 나타낼 수 있을 때 사용하는 대응짝 독립성 검정 방법

$$100(1-\alpha)\%\ CI = \hat{K} \pm Z_{\alpha} \times S.E.(\hat{K})$$

$$S.E.(\hat{K}) = \sqrt{\frac{P_e^2 + P_e - \frac{\sum_{i=1}^{I} R_i C_i (R_i + C_i)}{N^3}}{(1-P_e)^2 N}}$$

R_i 각 수준에서의 행 합계

카파 검정

카파 검정 (Kappa Statistics)

3×3 이상의 혼동행렬로 대응짝 데이터를 나타낼 수 있을 때 사용하는 대응짝 독립성 검정 방법

$$100(1-\alpha)\%\ CI = \hat{K} \pm Z_{\alpha} \times S.E.(\hat{K})$$

$$S.E.(\hat{K}) = \sqrt{\frac{P_e^2 + P_e - \frac{\sum_{i=1}^{I} R_i C_i (R_i + C_i)}{N^3}}{(1-P_e)^2 N}}$$

R_i 각 수준에서의 행 합계

카파 검정

카파 검정 (Kappa Statistics)

3×3 이상의 혼동행렬로 대응짝 데이터를 나타낼 수 있을 때 사용하는 대응짝 독립성 검정 방법

$$100(1-\alpha)\%\ CI = \hat{K} \pm Z_{\alpha} \times S.E.(\hat{K})$$

$$S.E.(\hat{K}) = \sqrt{\frac{P_e^2 + P_e - \frac{\sum_{i=1}^{I} R_i C_i (R_i + C_i)}{N^3}}{(1-P_e)^2 N}}$$

C_i

각 수준에서의 열 합계

카파 검정

카파 검정 (Kappa Statistics)

3×3 이상의 혼동행렬로 대응짝 데이터를 나타낼 수 있을 때 사용하는 대응짝 독립성 검정 방법

$$100(1-\alpha)\%\ CI = \hat{K} \pm Z_{\alpha} \times S.E.(\hat{K})$$

$$S.E.(\hat{K}) = \sqrt{\frac{P_e^2 + P_e - \frac{\sum_{i=1}^{I} R_i C_i (R_i + C_i)}{N^3}}{(1-P_e)^2 N}}$$

총 합계

THANK YOU

♥ 어흥범주 클린업 끝끝끝 ♥

팀장님이 꼬옥 넣어달라 하셨습니다^^ (인증가능)