2025.01.25

16기 신입교육세션

2주차 - 분류 모델링

B.a.f

CONTENTS

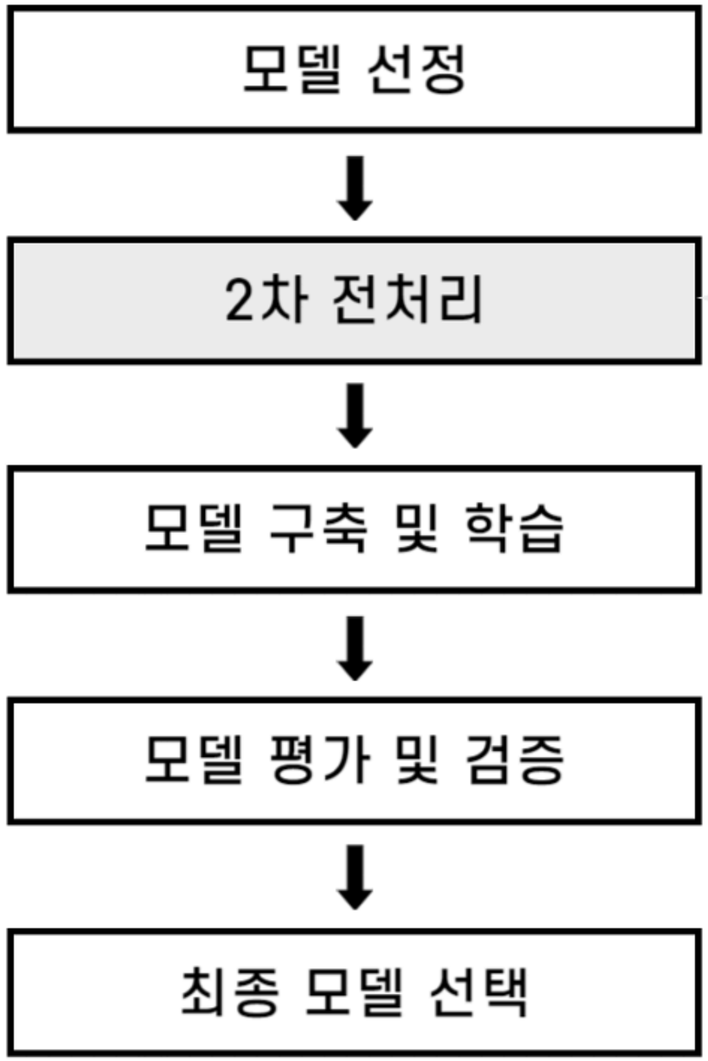
02 모델링 개요

1. 모델링 개요

2. 분류 모델링

B.a.f

모델링 과정



STEP1) 범주형 데이터 수치화

STEP2) 변수 선택

STEP3) train, valid, test 셋 분리

STEP4) 변수 스케일링 (필요시)

이 순서는 꼭 지키기!

지도 학습

지도 학습이란 ?

종속 변수 y가 데이터에 있는 경우, y를 예측하기 위한 학습 방법

좋은 모델이란 ?

test data

train data로 학습한 모델이, 새로운 데이터가 주어져도 정확히 예측하는 것

- 일반화
- 과대적합 (overfitting) & 과소적합 (underfitting)

분류 문제

binary

- 이메일이 피싱 메일은 아닐까 ?
- 고객이 제품을 계속 사용할까 ?
- 사용자가 광고를 클릭할까 ?

☑ 타이타닉 데이터는 이진 분류 문제

categorical

- 고객의 대출 등급은 무엇일까 ?
- 사용자가 제일 좋아하는 음악 장르는 무엇일까 ?

모델 종류

- 로지스틱 회귀, SVM, 랜덤포레스트, XGBoost 등

분류 문제

혼동행렬

Predicted

Actual		Positive	Negative
	Positive	TP (True Positive)	FN (False Negative)
	Negative	FP (False Positive)	TN (True Negative)

- TP : 모델이 Positive라고 예측한 것이 정답인 샘플
- FP : 모델이 Positive라고 예측한 것이 오답인 샘플 (1종 오류)
- FN : 모델이 Negative라고 예측한 것이 오답인 샘플 (2종 오류)
- TN : 모델이 Negative라고 예측한 것이 정답인 샘플

분류 문제

Accuracy

Predicted

Actual		Positive	Negative
	Positive	TP (True Positive)	FN (False Negative)
	Negative	FP (False Positive)	TN (True Negative)

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

- 전체 샘플 중 정답을 맞춘 비율
- Accuracy만 가지고 성능을 판단해서는 안됨
- 불균형 데이터에서는 Accuracy로 성능 판단 X

  ex) 100명 중 1명이 암환자인 데이터
  샘플 모두 음성이라 예측해도 accuracy는 99%

분류 문제

Precision (정밀도)

Predicted

Actual		Positive	Negative
	Positive	TP (True Positive)	FN (False Negative)
	Negative	FP (False Positive)	TN (True Negative)

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

- True라 예측한 것중 진짜 True인 비율
- Precision이 높다 : 정말 확실한 경우에만 참이라 예측
- Precision이 낮다 : 참이 아닌데 참이라 예측한 샘플 수가 많다
  ex) 스팸메일이 아닌데 스팸메일이라 판단해 차단함

분류 문제

Recall (재현율)

Predicted

Actual		Positive	Negative
	Positive	TP (True Positive)	FN (False Negative)
	Negative	FP (False Positive)	TN (True Negative)

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

- 실제 True 샘플 중 True라 예측한 비율
- Recall이 높다 : True라 예측한 샘플이 많다
- Recall이 낮다 : True인데 못찾은 샘플이 많다

** 질병 유무를 판단할 때에는 recall이 더욱 중요

분류 문제

f1-score

Predicted

Actual		Positive	Negative
	Positive	TP (True Positive)	FN (False Negative)
	Negative	FP (False Positive)	TN (True Negative)

Precision

Recall

Trade off →

$$F1\ Score = 2 \times \frac{recall \times precision}{recall + precision}$$

03 앙상블 기법

1. 배깅 (Bagging)

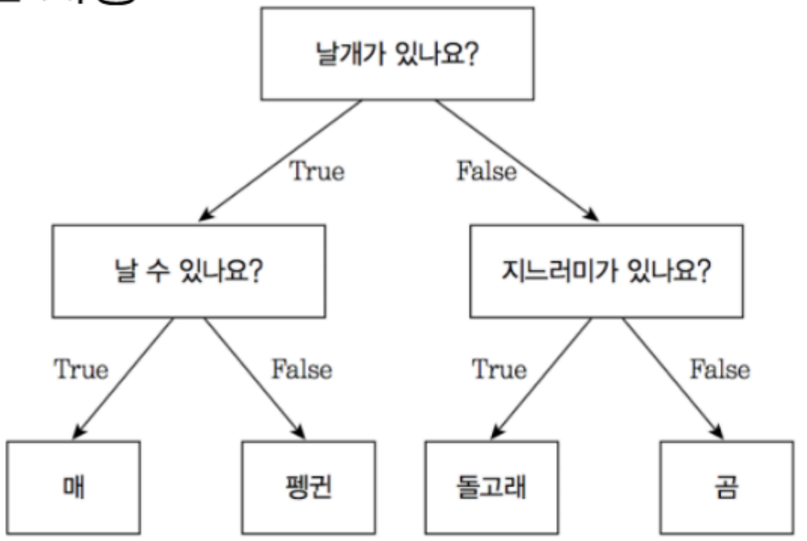
2. 부스팅 (Boosting)

B.a.f

앙상블 (Ensemble)

: 여러 개의 분류기(=모델)를 생성한 후, 각 분류기들의 예측 결과를 결합함으로써
보다 정확한 예측을 도출하는 기법

- 여러 개의 약한 분류기를 결합하여 강한 분류기 생성
- 일반적으로 의사결정 트리(Decision Tree)를 기본 알고리즘으로 사용



▶ 의사결정 트리(Decision Tree)

(1) 배깅 (Bagging)

: 여러 개의 모델을 독립적으로 학습시킨 후, 그 결과를 투표 또는 평균을 통해 종합하는 방식

Bootstrap

- 데이터셋에서 여러 개의 부분 데이터셋을 뽑아 모델에 할당 후,

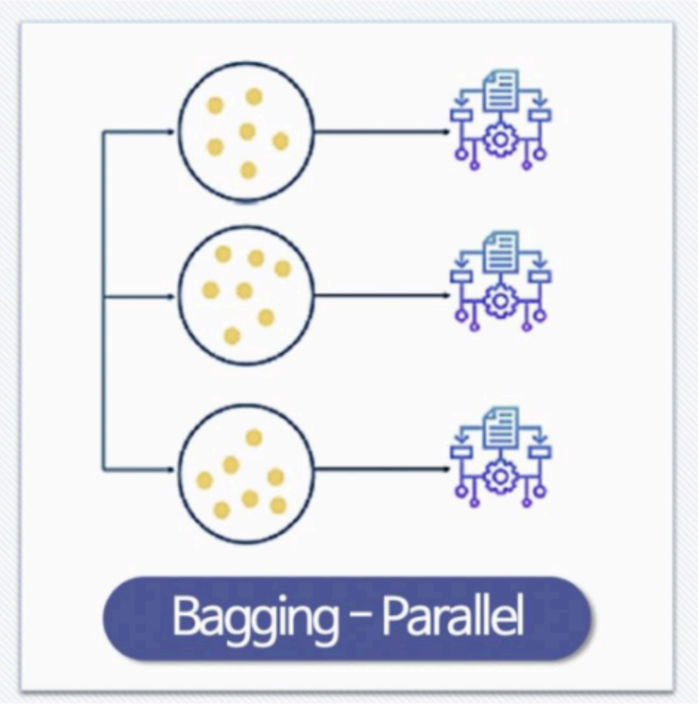
  각 모델을 학습시켜 결과물을 집계함 *Aggregation*

- 독립적인 결정트리가 각각 예측값을 출력하고,

  그 값들을 집계해 최종 결과값을 예측

- 병렬 방식

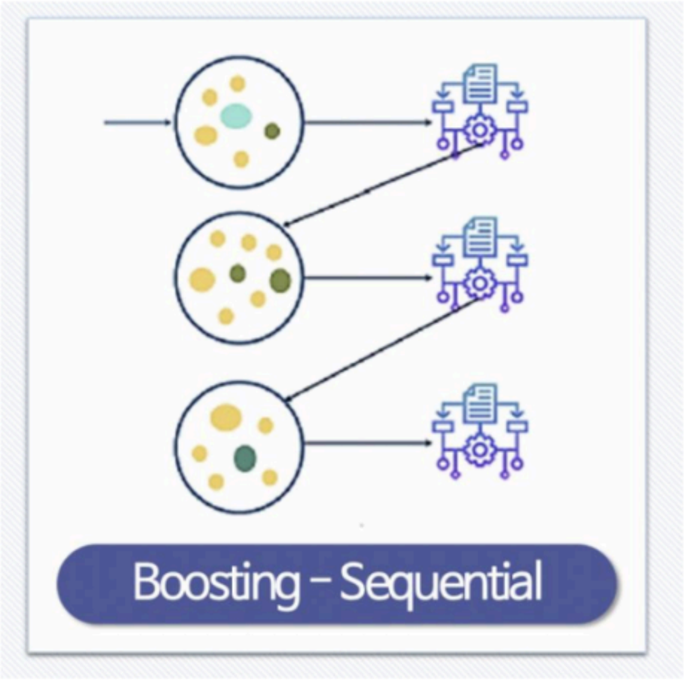
- 대표적인 모델: Random Forest



(2) 부스팅 (Boosting)

: 각 모델이 순서대로 학습되면서 오답에 높은 가중치를 주어 오차를 보완해나가는 방식

- 가중치를 활용하여 약한 분류기(=모델)를 강한 분류기로 만듦
- 배깅과 달리, 부스팅은 모델 간 팀워크가 이루어짐
- 일반적으로 배깅에 비해 성능이 좋음
- 모델 학습에 순서가 있어 속도가 느림
- 순차적 방식
- 대표적인 모델: XGBoost, Gradient Boost 등



04. 2차 전처리

1. 범주형 데이터 수치화
2. train, valid, test 셋 분리
3. 변수 스케일링

B.a.f

범주형 데이터 수치화

- 컴퓨터는 숫자를 인식 (= 모델은 수치형 데이터만 입력 받음)
- 범주형 데이터는 수치형으로 변환해줘야 함

원핫 인코딩 (One-Hot Encoding)

: 해당 변수 내에 존재할 수 있는 n개의 값들을 각각 n개의 벡터(열)로 표현하는 방식

- 명목형 변수(ex. 성별, 혈액형 등)인 경우에 사용
- 0과 1의 값만을 갖는 더미변수를 생성
- 크기의 의미가 없음
- n의 크기가 클수록 차원이 증가함

ID	과일
1	사과
2	바나나
3	체리

One-Hot Encoding

ID	사과	바나나	체리
1	1	0	0
2	0	1	0
3	0	0	1

라벨 인코딩 (Label Encoding)

: 해당 변수 내에 존재할 수 있는 n개의 값들을 각각 0 ~ (n-1)의 연속적인 수치로 변환하는 방식

- 순서형 변수(ex. 성적 등급, 학년 등)인 경우에 사용
- 순위가 보존되어 크기의 의미가 있음
- 변수의 총 개수는 그대로 유지

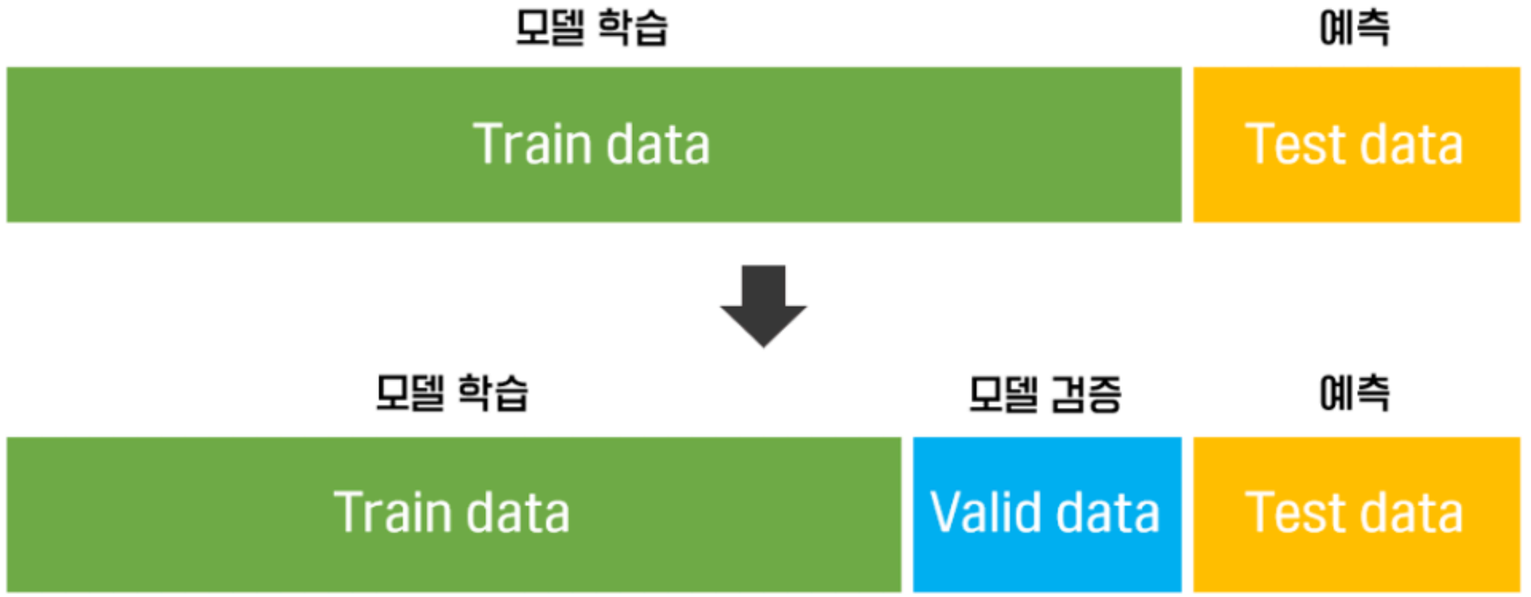
LabelEncoder

ID	성적
1	C
2	B
3	A

→

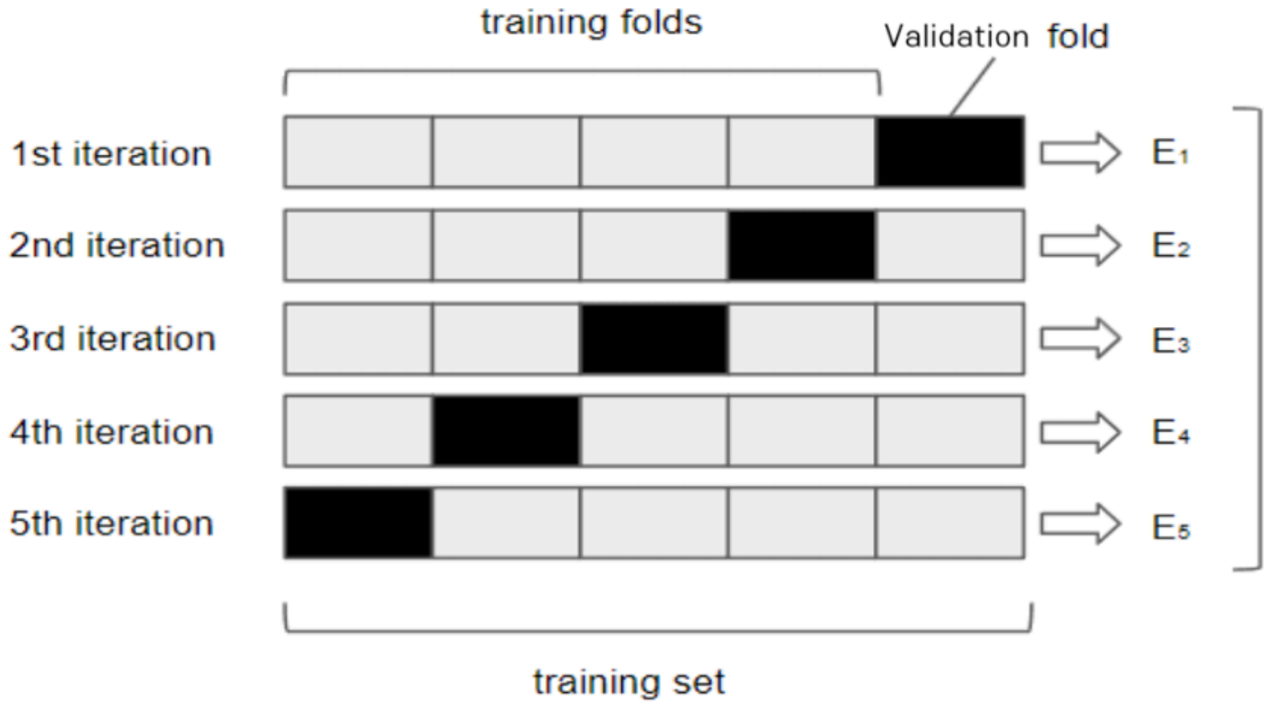
ID	성적
1	0
2	1
3	2

train, valid, test 셋 분리



- 데이터 분할을 통해 모델의 성능을 높일 수 있음
- train : valid : test = 6 : 2 : 2 or 7 : 2 : 1
- test 셋은 모델 학습에 절대 사용 X

K-fold 교차 검증(Cross Validation)



$$E = \frac{1}{5}\sum_{i=1}^{5} E_i$$

- 데이터를 k개로 분할: k-1개는 train 셋 / 1개는 valid 셋
- 위 과정을 k번 반복하고, k개의 성능의 평균값 계산
- 데이터셋의 크기가 작을 때 유용함

변수 스케일링 (Feature Scaling)

- 변수들의 범위(scale)를 통일시켜주는 작업
- 대부분의 모델(Tree 기반 모델 제외)에서 필요한 단계
  - (ex. 특정 회귀 모형, 거리 기반 모델 등)
  - 특정 변수의 값의 범위가 클수록 영향력이 큰 변수라고 잘못 인식

(주의 1) 스케일링된 데이터로 모델 학습 시, 예측값은 원래 스케일로 변환하기

(주의 2) train, valid, test 셋 분리한 후, 각각 스케일링 진행하기

(주의 3) train 셋에 적용한 scaler로 valid, test셋에도 적용하기

(1) Standard Scaler

: 값의 분포를 평균이 0, 분산이 1인 정규분포로 스케일링하는 방식

- 가장 일반적인 방법
- 이상치에 민감 (평균과 분산에 영향을 주기 때문)

(2) Min-Max Scaler

: 값의 분포를 0과 1 사이의 값으로 스케일링하는 방식

- 최솟값은 0, 최댓값은 1
- 이상치에 민감 (이상치가 극값이 되어 데이터가 분포가 비정상적으로 좁아지기 때문)

(3) Max Abs Scaler

: 값의 절댓값이 0과 1사이가 되도록 스케일링하는 방식

- 모든 값은 -1과 1사이로 표현됨
- 데이터가 양수이면, Min-Max Scaling과 동일
- 이상치에 민감

(4) Robust Scaler

: Standard Scaler에서 평균과 분산 대신에 중앙값과 사분위값을 사용하여 스케일링하는 방식

- 이상치에 강함 (중앙값과 사분위값을 사용하기 때문)
- 보통 Standard Scaler에 비해 데이터가 더 넓은 범위로 분포됨

05. 실습 & 과제

B . a . f

분류 모델링 과제 : 해당 모델에 맞게 전처리 및 모델링 해보기

(1) 조별 모델 배정

- 조별로 해당 모델에 대해 공부한 후, 어떻게 전처리할 지 회의해보기
- 과제 제출은 개별로 GitHub에 제출

조	조원				모델
1	김현	안재혁	최연식		RandomForest
2	서정유	유영우	지승우		Gradient Boosting
3	김민정	박서연	이용혁	함주헌	Logistic Regression

(2) 회귀 모델 예습

[파이썬으로 시작하는 데이터사이언스] 2-2. (기본) 5. 회귀모델 만들기 예습 수강

감사합니다

B.a.f