Ensemble

(Voting, Stacking)

ML Session 7차시

CONTENTS.

01. Ensemble

- 지난 과정 복습
- Ensemble

02. Voting

- Voting
- Hard Voting
- Soft Voting

03. Stacking

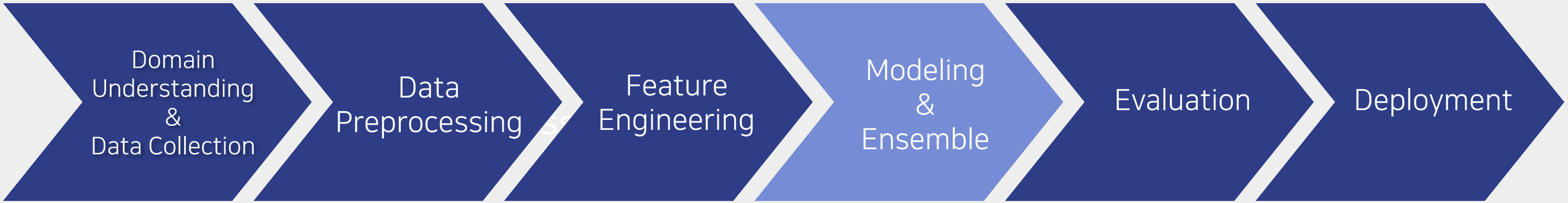
- Stacking
- CV 기반 Stacking

04. Submission Ensemble

- Submission Ensemble

Ensemble

지난 과정 복습



Ensemble

앙상블 (Ensemble)이란?

- 여러 모델을 생성하고 예측을 결합함으로써 보다 정확한 최종 예측을 도출하는 기법
    - ·여러 모델을 사용하여 예측력을 높이기 위해 사용
        - 단일 모델보다 성능이 좋음
        - 단일 모델/알고리즘 약점 보완
    - ·하나가 아닌 여러 모델을 사용하여 과적합 방지
        - 다양한 데이터와 관점에서 나온 예측값 이용
        - 더 나은 일반화

Ensemble

Ensemble 사용 시 예측력이 올라가는 이유

- 모델 학습 시 Bias와 Variance에 따라 다음과 같이 모델 유형을 나눌 수 있음

Model 1: 정확도가 가장 낮아 가치 X

Model 2: 추정값들이 전체적으로 정답과 유사하지만 그 안에서 분산이 큰 모델 → 모델 비교적 복잡 (Overfitting)

Model 3: 추정값 간 분산은 작지만 정답과 떨어진 모델 → 모델 비교적 단순

Model 4: 추정값 간 분산도 작고 정답고 유사한 모델 = 가장 좋은 모델

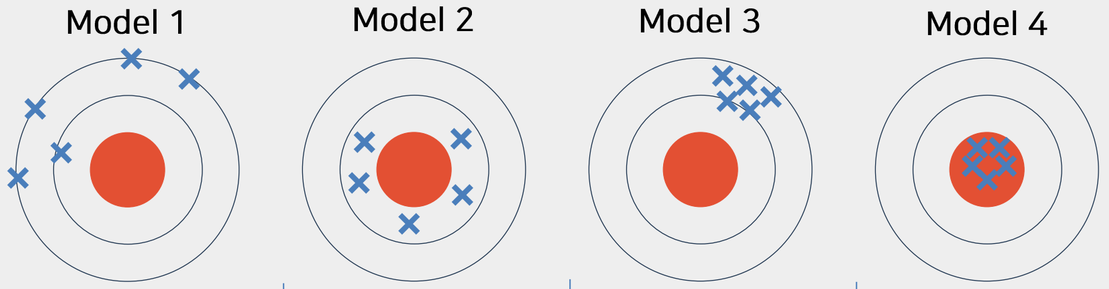
→ 앙상블을 통해 2번과 3번 모델을 결합하여 4번의 모델에 가까워지도록 하기 위함!

Bagging: 여러 서브셋으로 개별 모델 학습 후, 예측의 결과를 결합하여 최종 예측 출력

- 분산을 줄이는 방법 (2번 → 4번)

Boosting: 이전 모델에서 오분류된 샘플에 가중치를 부여하여 성능 향상

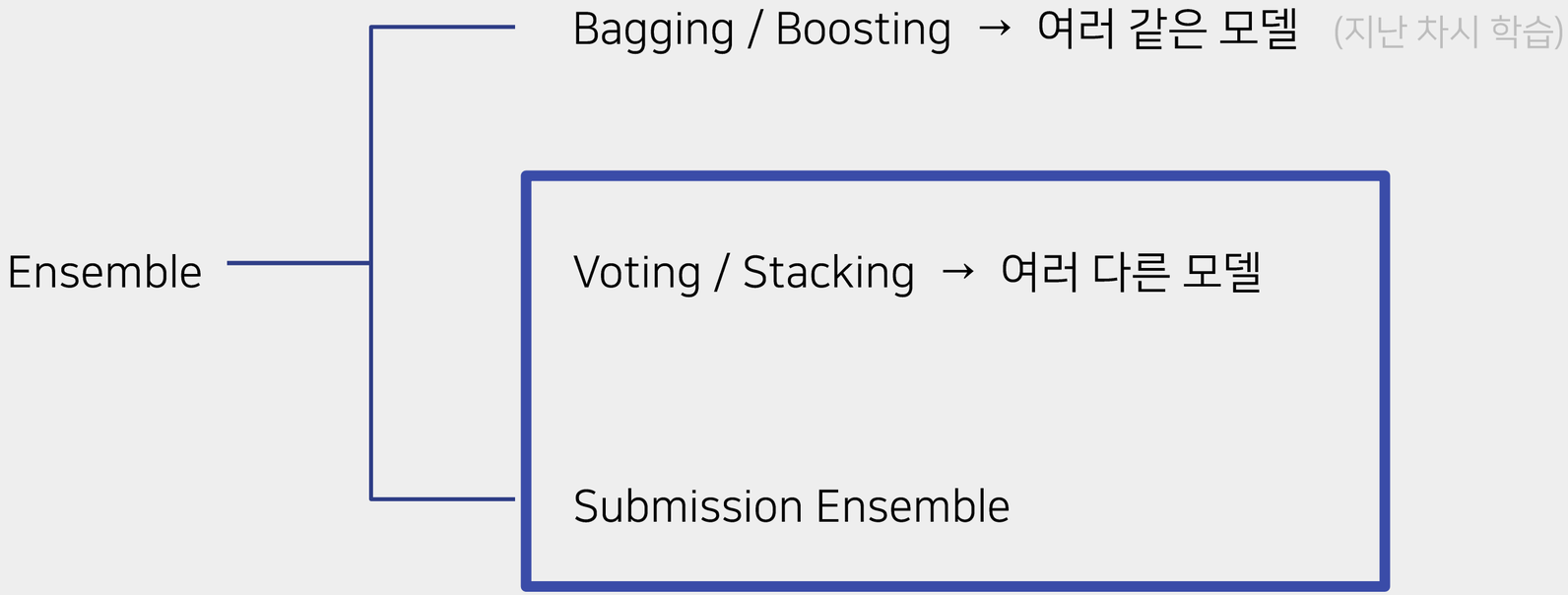
- 편향을 줄이는 방법 (3번 → 4번)



	Model 1	Model 2	Model 3	Model 4
Bias(편향)	↑	↓	↑	↓
Variance(분산)	↑	↑	↓	↓

Ensemble

Ensemble 유형



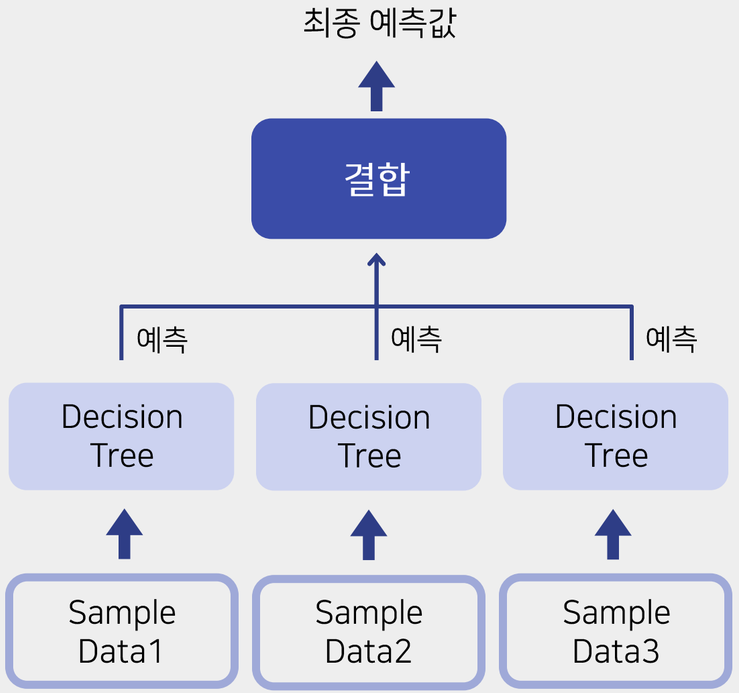
Voting

Voting이란?

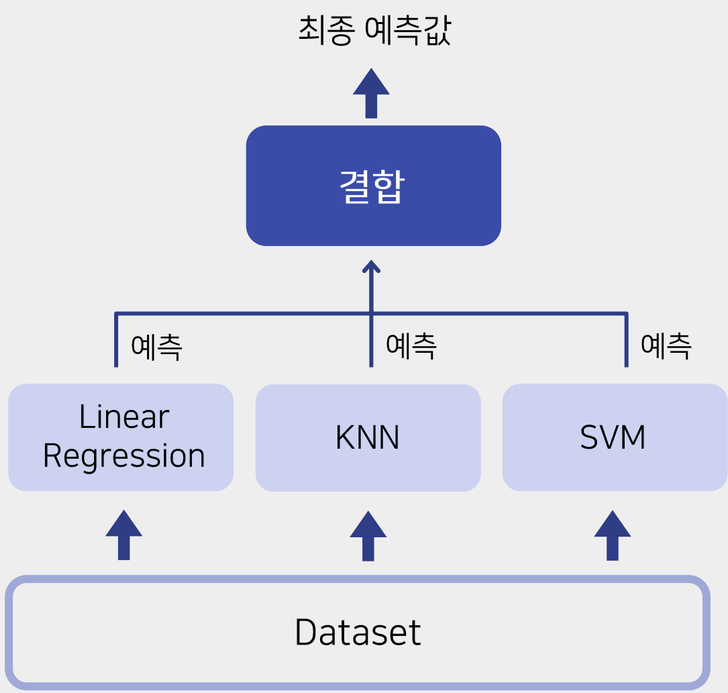
- 여러 개의 예측기가 투표를 통해 최종 예측 결과를 결정하는 방식
  - Hard Voting
  - Soft Voting

Bagging vs Voting

- 공통점: Aggregation (분류 – 최빈값 / 회귀 - 평균)
- 차이점: 모델 종류
  - Bagging: 각 모델이 같은 유형의 알고리즘
  - Voting: 각 모델이 서로 다른 알고리즘



[Bagging]



[Voting]

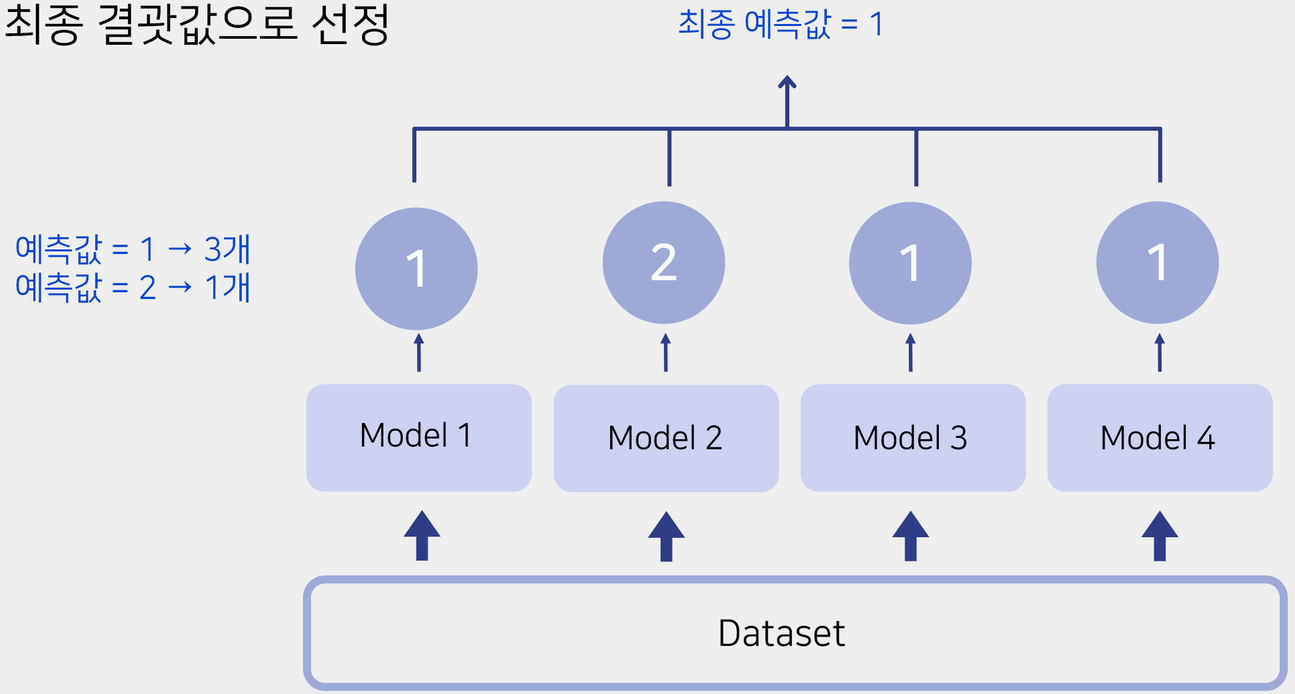
Voting

Hard Voting

Hard Voting이란?

- 예측한 결과 값들 중 다수의 분류기가 결정한 예측값을 최종 결괏값으로 선정

  ≒ 다수결 원칙



Voting

Soft Voting

Soft Voting이란?

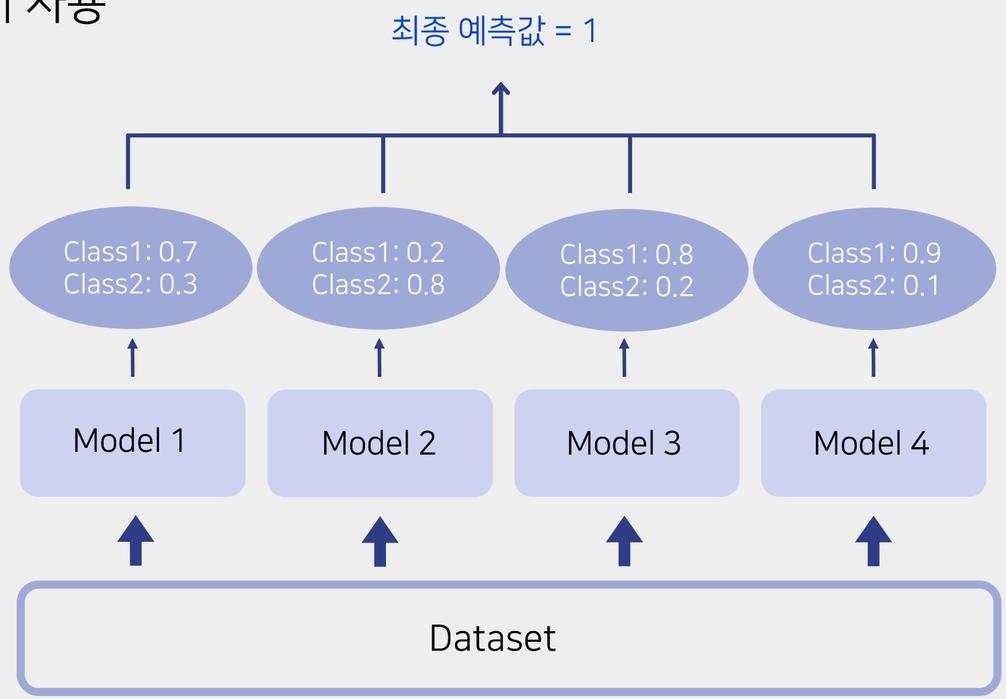
- 분류기들의 클래스 값 결정 확률의 평균을 구했을 때 확률이 가장 높은 클래스 값을 최종 결괏값으로 선정

  → 일반적으로 Hard Voting보다 Soft Voting이 성능이 좋아 더 많이 사용

1로 예측한 확률: $\frac{0.7+0.2+0.8+0.9}{4} = 0.65$

∨

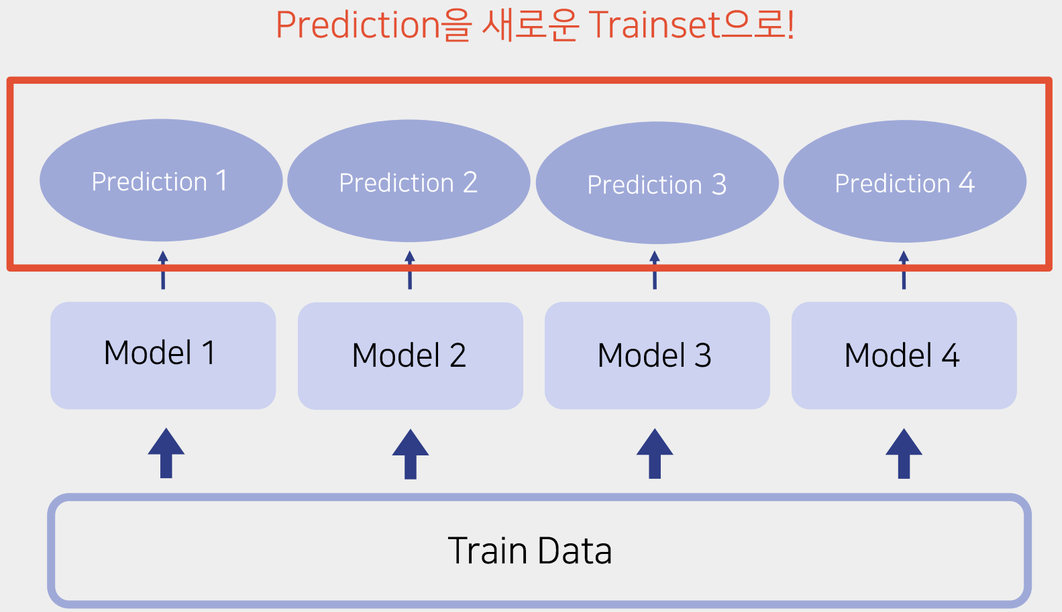
2로 예측한 확률: $\frac{0.3+0.8+0.2+0.1}{4} = 0.35$



Stacking

Stacking이란?

- 서로 다른 모델의 예측 결괏값을 학습 데이터로 만든 후 다른 모델로 재학습시켜 결과 예측



Stacking

Stacking 기본 구조

- 여러 개의 모델에 대한 예측값을 쌓은 뒤, 이에 대한 예측을 다시 수행

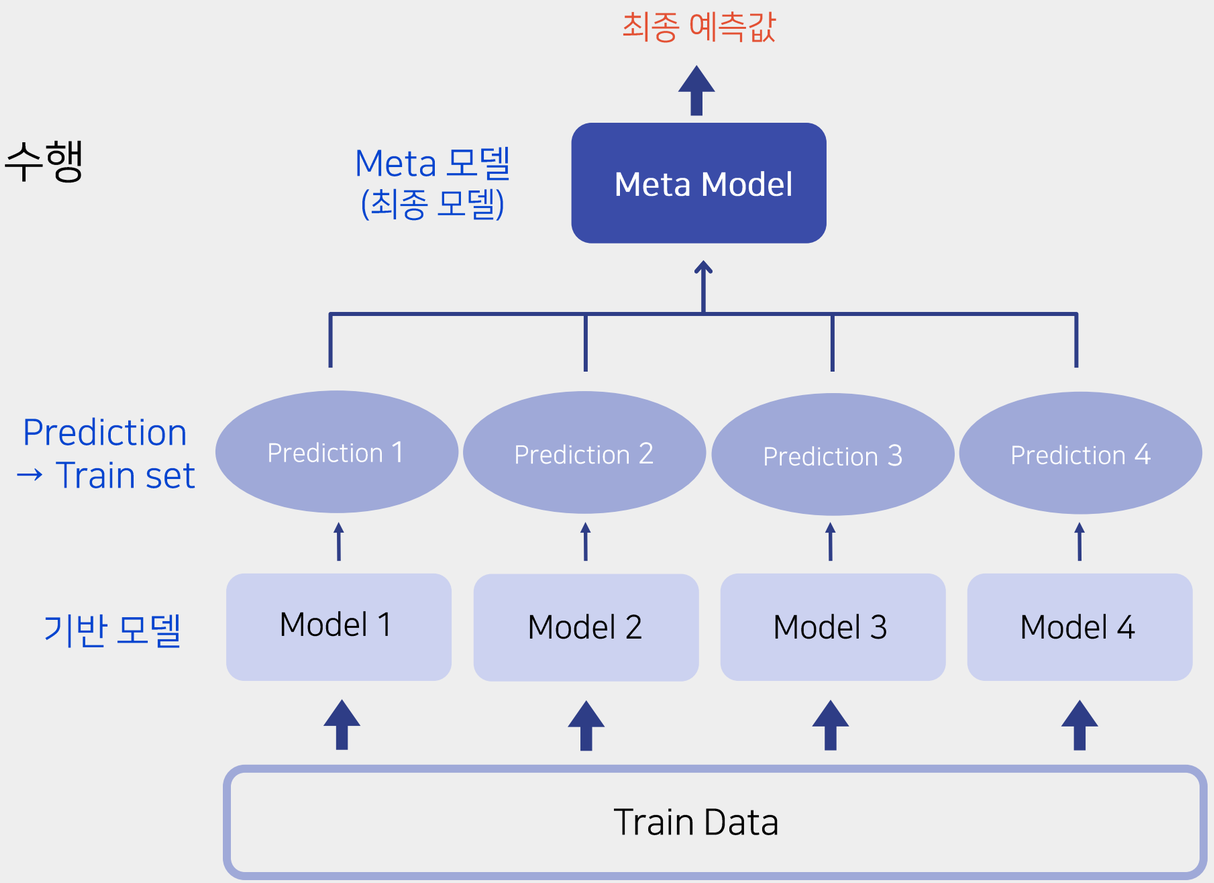
• Meta Model 활용

- 성능 좋은 단일 모델
- Voting / Stacking (→ 2 Layers Ensemble)

•단점

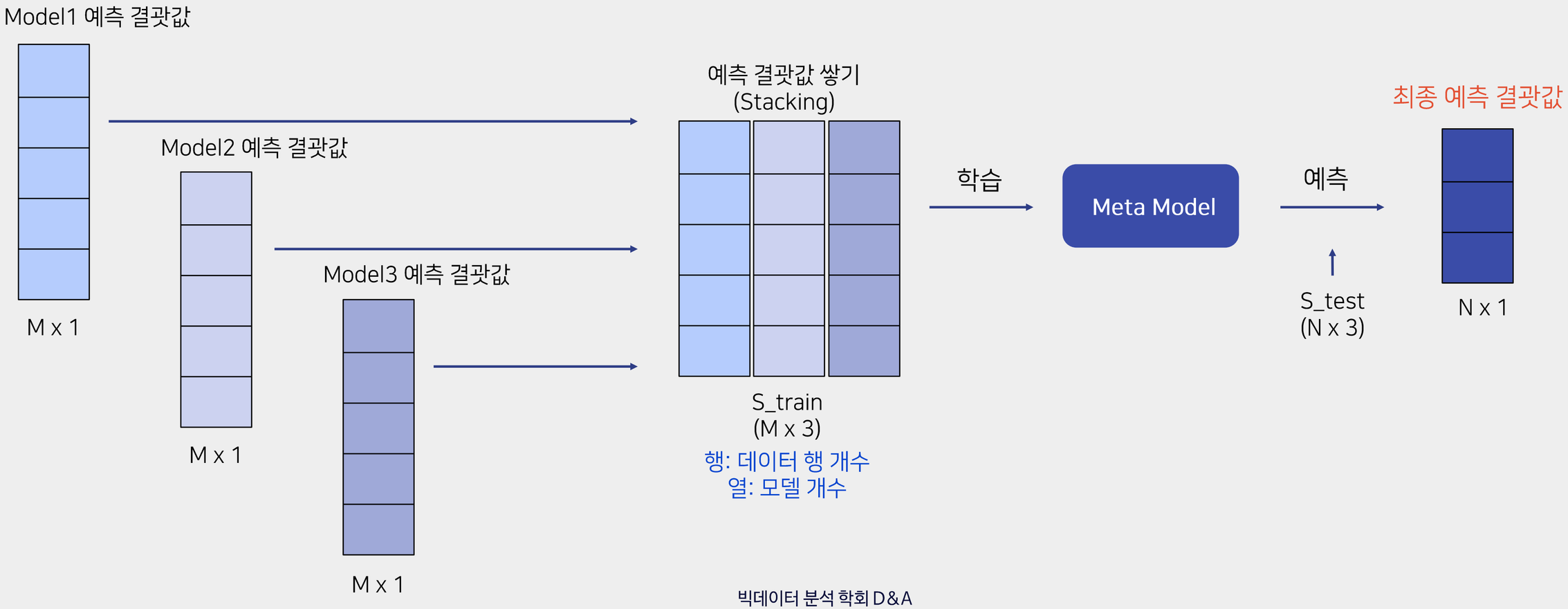
: 동일한 Train에 대한 Prediction을 학습했기 때문에

과적합 발생 가능성 ↑



Stacking

Stacking 기본 구조



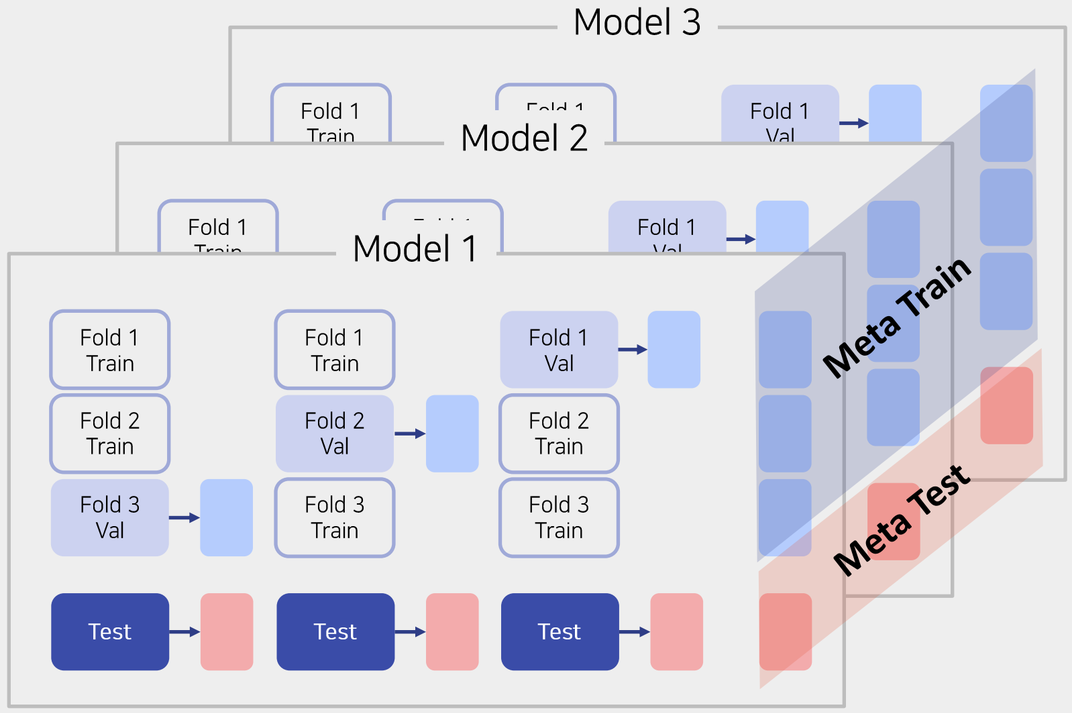
CV기반 Stacking

CV 기반 Stacking

과적합 문제를 해결하기 위한 방법
→ 최종 메타 모델을 위한 데이터셋을 만들 때 교차 검증 기반으로 예측된 결과 사용

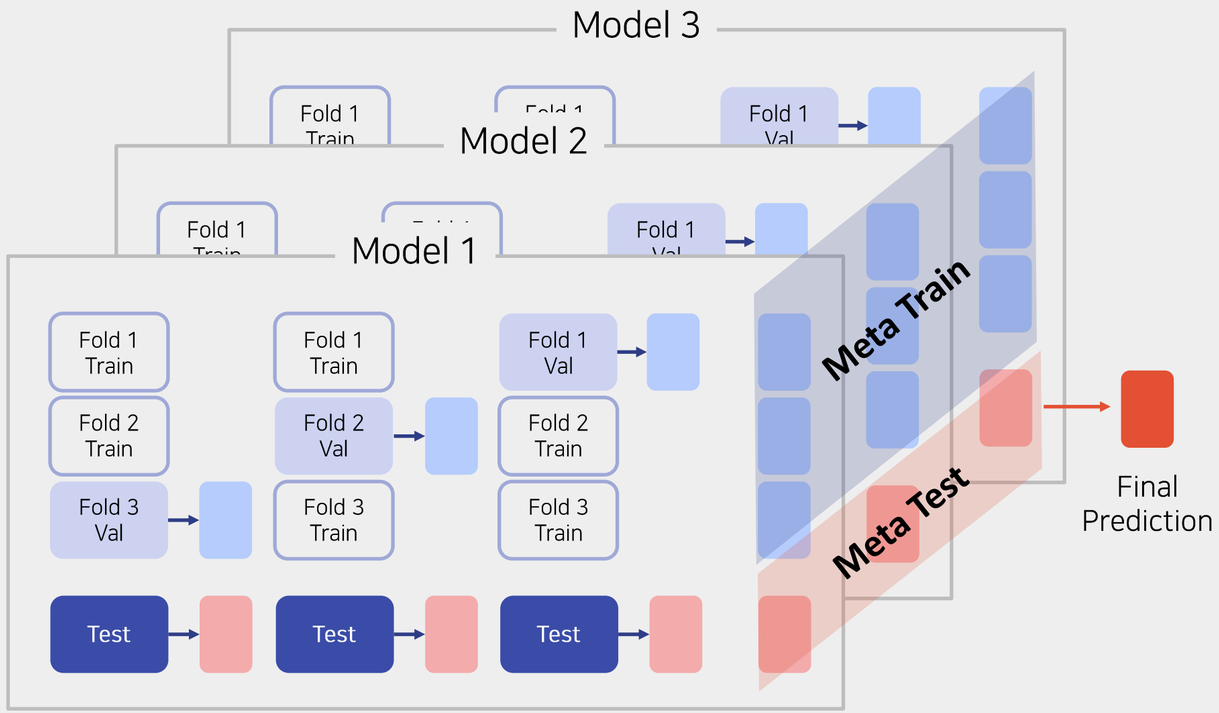
•방법

1. Train Set을 N개의 fold로 나눈다.
2. 학습에 사용하려는 (N-1)개로 개별 모델 학습,
   1개의 검증용 fold로 데이터 예측 후 결과를 저장한다.
   → N번 반복하여 메타 모델의 학습 데이터로 사용
3. fold별 Test Set에 대한 예측값의 평균으로 최종 결과값을 생성한다.
   → 메타 모델의 테스트 데이터로 사용
4. 각 모델들이 위의 과정으로 생성한 학습 데이터와 테스트 데이터를 모두 합쳐 메타 모델의 학습 및 예측을 수행한다.



CV기반 Stacking

CV 기반 Stacking



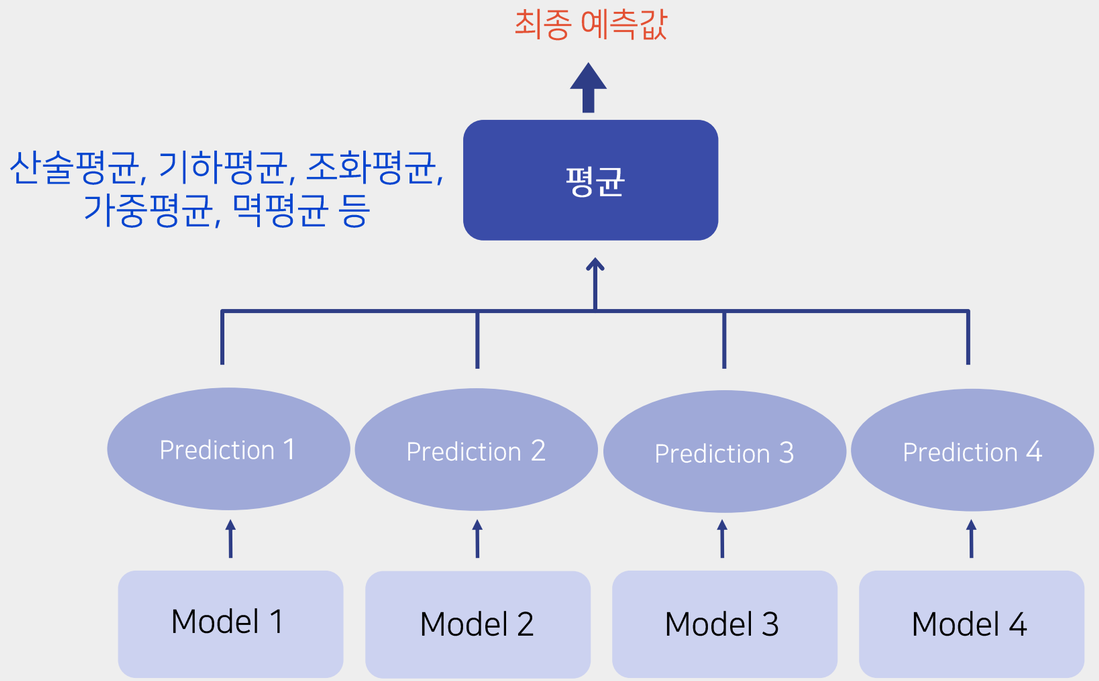
Submission Ensemble

Submission Ensemble

- Submission 파일끼리 산술평균, 기하평균 등을 활용하여 합치는 방법

•좋은 성능을 내기 위해서는?

- 모델이 학습한 데이터가 조금씩 달라야 한다.
- 학습 모델들이 서로 달라야 한다.
- 성능이 어느 정도 유사한 Prediction끼리 합쳐야 한다.



Submission Ensemble

다양한 평균

- 산술평균: 모델들의 예측값을 단순히 더한 후 평균을 내는 방식
- 기하평균: 각 모델의 예측값의 곱의 제곱근을 구하는 방식

  → 극단값에 덜 민감하게 반응해, 안정적인 결과 제공 가능
- 조화평균: 낮은 값에 더 민감하게 반응하는 평균 계산 방식으로, 극단적으로 낮은 예측값이 있을 때 전체 평균에 강하게 영향을 미침
- 가중평균: 모델의 중요도에 따라 가중치를 달리 주어 평균을 계산하는 방식

$$\text{산술평균} = \frac{a+b}{2}$$

$$\text{기하평균} = \sqrt{ab}$$

$$\text{조화평균} = \frac{1}{\frac{\frac{1}{a}+\frac{1}{b}}{2}} = \frac{2ab}{a+b}$$

REFERENCE

- Stacking 관련

https://data-analysis-science.tistory.com/61

https://m.blog.naver.com/winddori2002/221848433173

THANK YOU

ML Session 7차시

빅데이터 분석 학회 D&A