앙상블, 랜덤포레스트

Contents

01 앙상블(Ensemble)

서로 다른 여러 모델의 예측 결과를 바탕으로 새로운 모델을 만들어 더 정확한 예측 결과를 도출해내는 방법

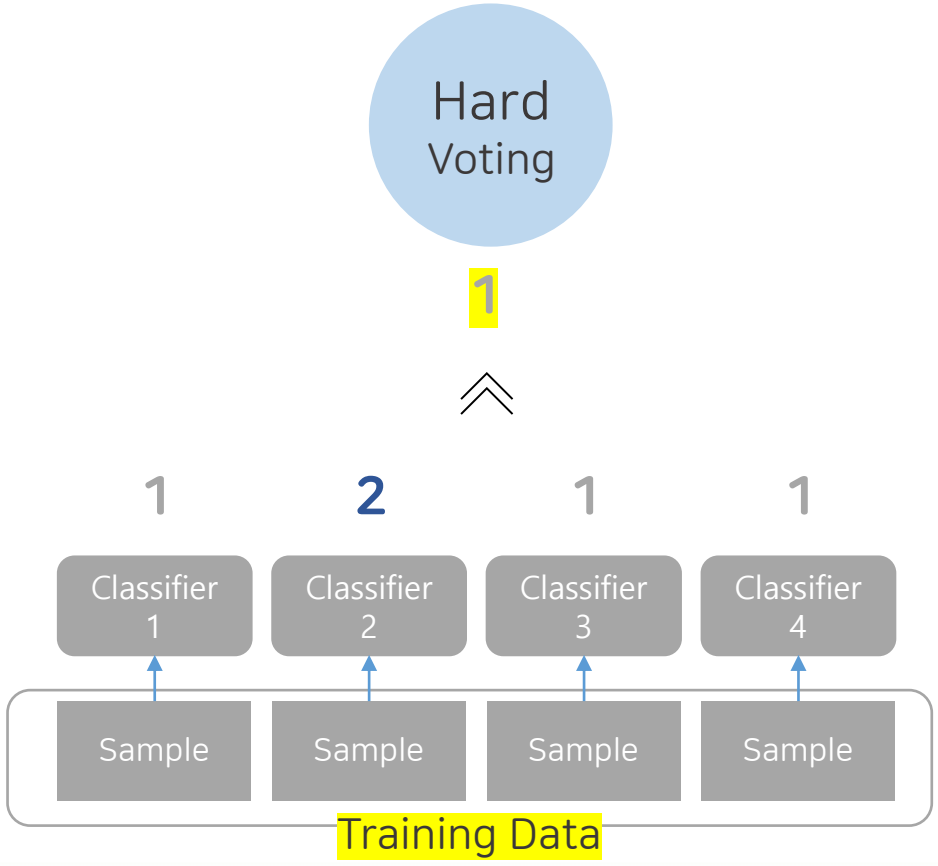
→ **이용 목적**: 단일 모델보다 신뢰성이 높은 예측 결과를 얻는 것!

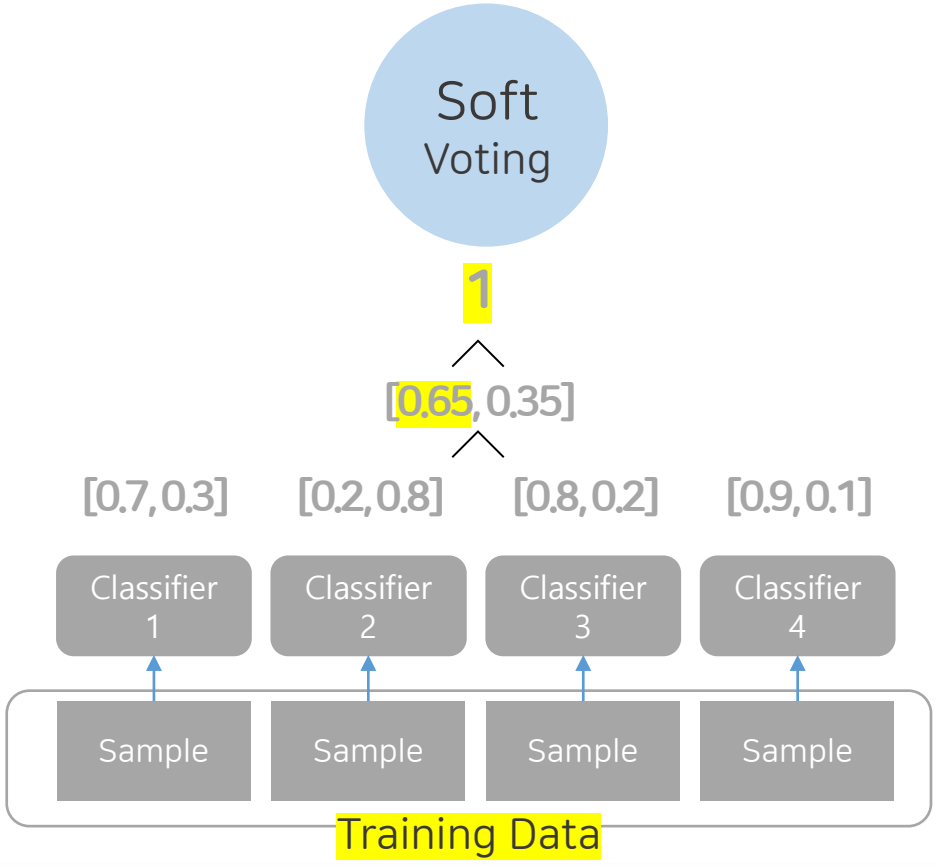
<앙상블의 종류>

- 보팅 (Voting)
- 배깅 (Bagging)
- 부스팅 (Boosting)
- 스태킹 (Stacking)

01.1 보팅(Voting)

서로 다른 알고리즘의 결과에 대해 투표로 최종 예측 결과를 결정





01.1 보팅(Voting)

사이킷런의 VotingClassifier 이용

```
from sklearn.ensemble import VotingClassifier
```

```
voting_classifier = VotingClassifier(estimators=[('LR', lr_clf), ('KNN', knn_clf),
voting='soft')]
```

Default: hard

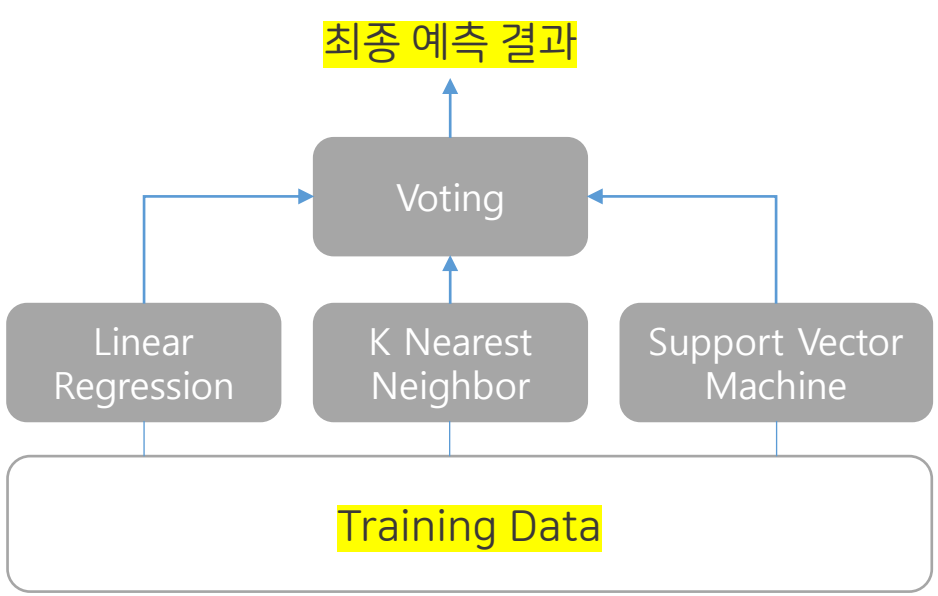
보팅에 사용할 Classifier 객체들
튜플 형식으로 입력 받음.

01.2 배깅(Bagging)

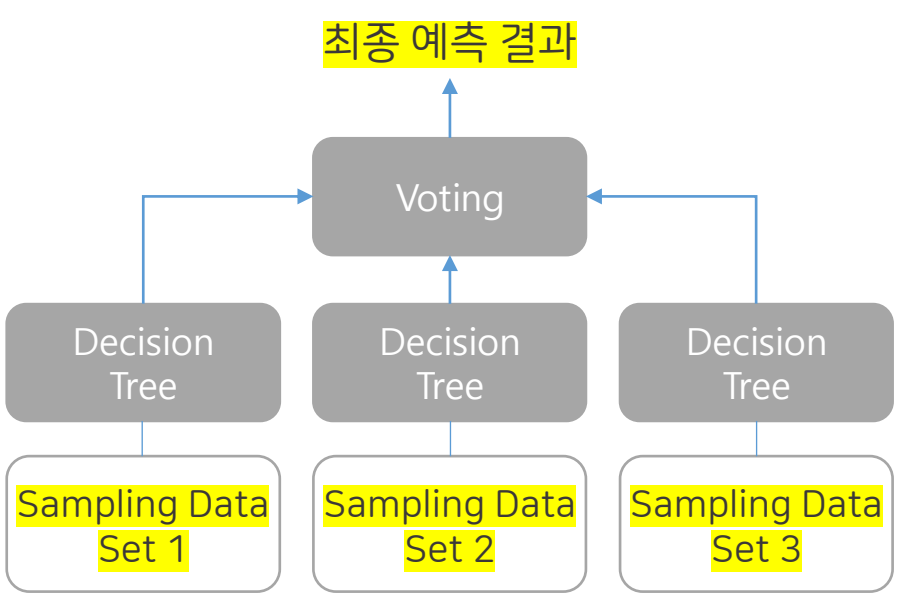
같은 알고리즘 여러 개에 각각 다른 데이터 샘플을 사용하여 학습 후 보팅으로 최종 예측 결과 결정

<보팅과의 차이점>

<Voting>



<Bagging>

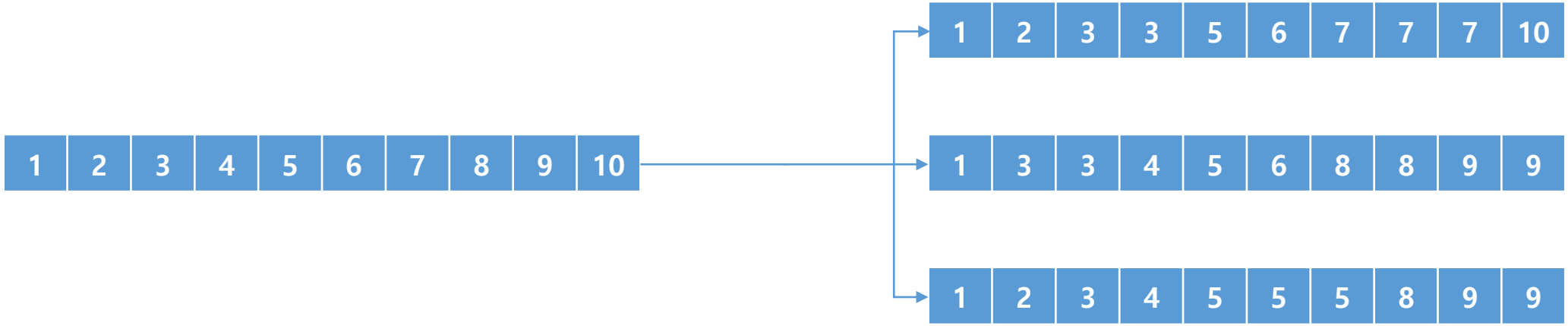


01.2 배깅(Bagging)

같은 알고리즘 여러 개에 각각 다른 데이터 샘플을 사용하여 학습 후 보팅으로 최종 예측 결과 결정

부트스트래핑(Bootstrapping) 분할 방식

: 중복을 허용하여 원본데이터로부터 데이터를 랜덤 샘플링. → 여러 개의 서브세트 생성



01.2.1 랜덤 포레스트(Random Forest)

배깅의 대표적인 알고리즘.

결정 트리를 기반 알고리즘으로 사용.

장점

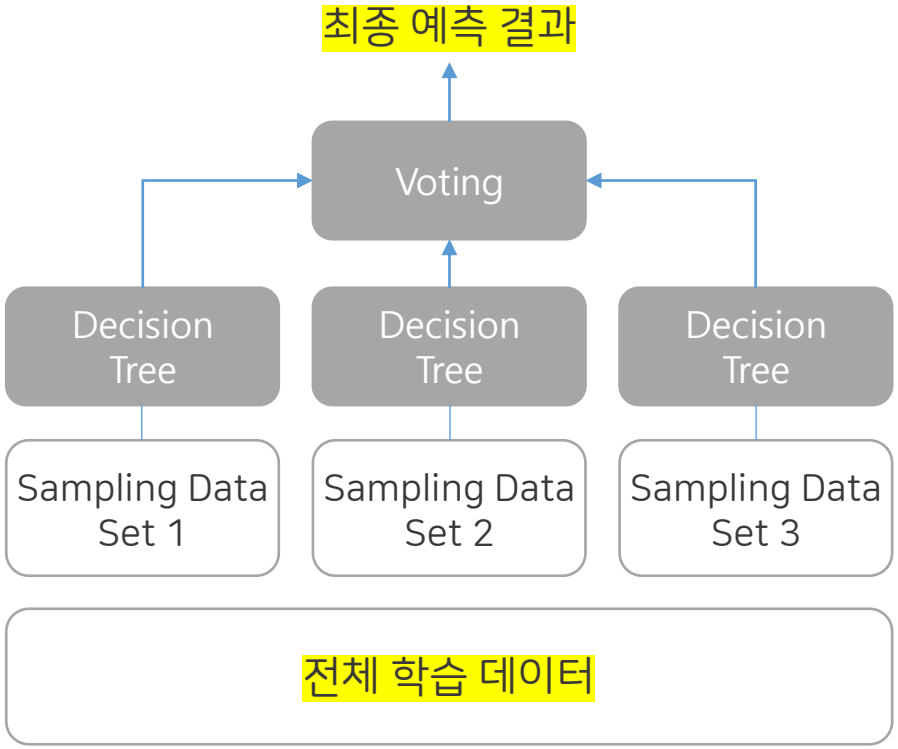
-비교적 빠른 수행 속도

-높은 예측 성능

단점

-너무 많은 하이퍼 파라미터

→튜닝에 많은 시간 소모



01.2.1 랜덤 포레스트(Random Forest)

사이킷런의 RandomForestClassifier 이용

<하이퍼 파라미터>

`n_estimators` 결정 트리의 개수 지정.
Default: 10

`max_features` 결정 트리에 사용된 max_features와 같은 파라미터. 최대 피처 개수
Default: auto 즉 sqrt (전체 피처 중 $\sqrt{\text{전체 피처 개수}}$)

이 외에도 결정 트리에서 사용되는 파라미터가 똑같이 적용될 수 있음.

Thank you