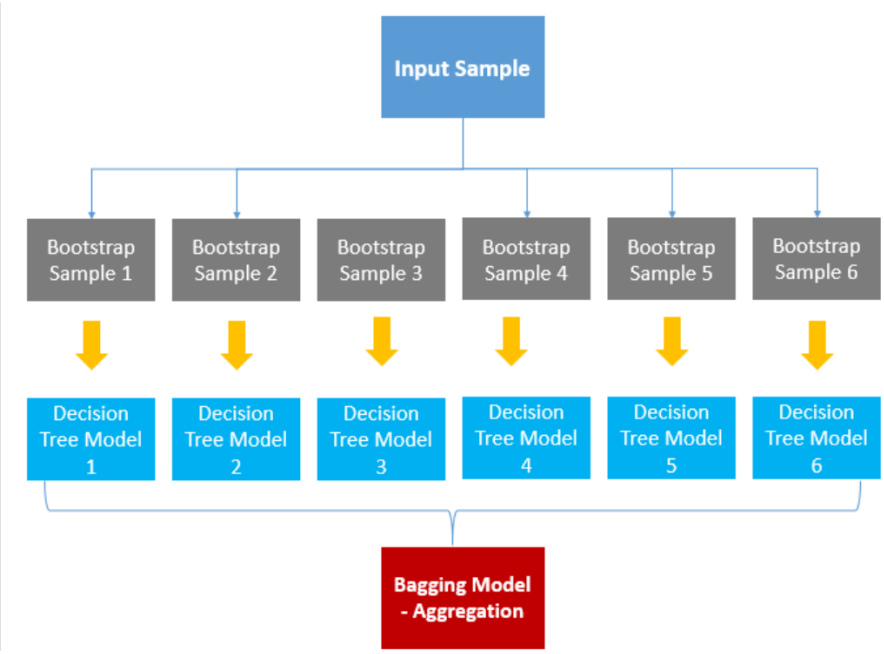
앙상블 5주차 과제 - 내용정리

앙상블

- 일반화와 강건성(강도)을 향상시키기 위해 다양한 모델의 예측 값 결합하는 방법
- 종류
  1. 평균 방법
     - 여러 개의 추정값을 독립적으로 구한 뒤 평균 취함
     - 결합 추정값은 분산이 줄어들기 때문에 단일 추정값보다 성능 우위
  2. 부스팅 방법
     - 순차적으로 모델 생성
     - 결합된 모델의 편향을 감소 시키기 위해 노력
     - 부스팅 방법의 목표 → 여러 개의 약한 모델들을 결합, 하나의 강력한 앙상블 모델 구축

1. 배깅

- Bootstrap aggregating의 약자
- 샘플을 여러 번 뽑아(Bootstrap) 각 모델을 학습시켜 결과물을 집계(Aggregration)하는 방법
- 분산을 줄이고 과적합을 막음 + 강력하고 복잡한 모델에서 성능 좋음



출처 : swallow.github.io

과정

1) Training Data (D개의 데이터 셋) → 2) Bootstrap samples(n개의 데이터 표본을 무작위 추출) →

3) Model(각각 표본에 대한 분류기 생성) → 4) Aggregation(집계) → 5) Output(최종 결과 출력)

편향과 분산의 관점으로 배깅 보기

- 배깅 핵심 아이디어 : 분산이 적은 모델을 얻기 위해 여러 개의 독립적인 모델의 예측값을 평균화 하는 것.
- 데이터를 리샘플링하여 각 샘플에 대한 훈련된 k개의 모델 예측 결과의 평균 취하기 때문에 분산이 기존 단일 모델의 $1/k$이 되도록 함.

> 즉, 여러 개의 분류기를 사용하여 평균을 취해 최종 예측값을 도출하기 때문에 편향은 유지하면서 분산을 줄일 수 있다.

여기서 문제가 발생할 수 있다. 실제로 데이터를 훈련할 때는 많은 데이터 셋을 필요로 하기 때문에 이 훈련된 모델들이 “독립적이다”라고 확정지을 수 없다.

그래서 배깅의 대표적인 모델인 `랜덤 포레스트 모델` 에서는 Feature를 랜덤으로 부분 선택하여 여러 모델을 생성한다. → 모델들의 상관성을 낮추어 독립성을 보장할 수 있도록 하기 위해서이다.

> 💡 **배깅은 과적합되는 경향이 있는 모델(높은 분산 모델)에 굉장히 효과적임을 알 수 있다.**

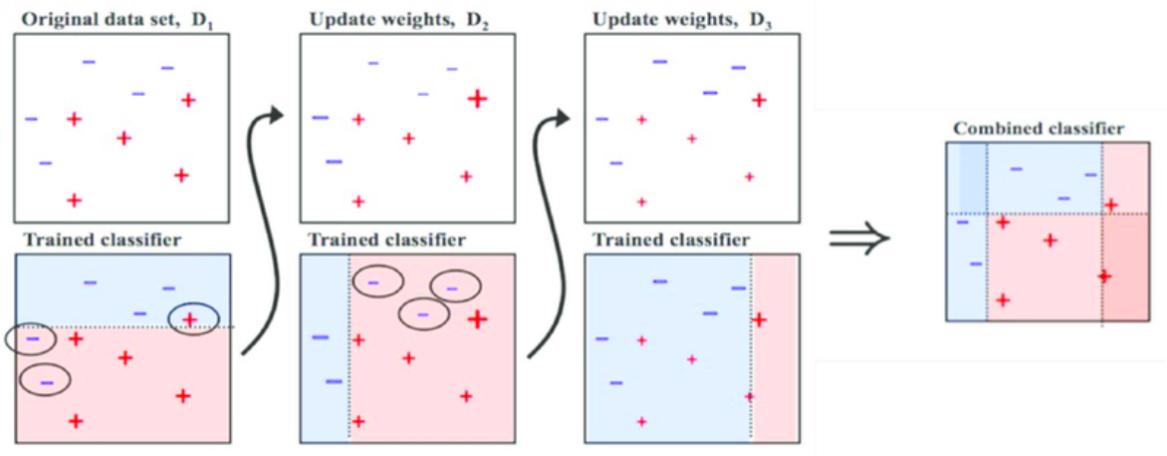
랜덤 포레스트(Random Forest)

- 배깅 기법의 대표적인 모델 (DT 기반)
- 변수 랜덤하게 선택하여 사용한다.

2. 부스팅

- bootstrap data 샘플링 기법을 사용하는 점에서 배깅과 같다. 하지만 배깅과 달리 데이터를 독립적으로 샘플링하는 것은 X
- 배깅은 여러 분류기를 한 번에 만들어 결합하는 병렬 학습과 달리 부스팅은 순차 학습을 진행

> 💡 부스팅은 여러 개의 분류기가 순차적으로 학습 수행하되, 앞에서 학습한 분류기가 예측이 틀린 데이터에 대해서는 올바르게 예측할 수 있도록 다음 분류기에서는 가중치(weight)를 부여하면서 학습과 에측 진행한다.



출처 : Medium (Boosting and Bagging explained with examples)

과정

1) 랜덤 데이터 샘플 D1으로 learner M1 생성하여 성능 판단

2) 새로 생성된 랜덤 데이터 샘플 D2에 M1에서 오분류된 데이터 포함하여 learner M2 생성

3) 이때 M1에서 오분류된 데이터에 가중치를 부여하여 M2 학습하기 때문에 M2에서는 오분류 데이터에 집중하여 학습 진행

4) 위 방식대로 반복하여 learner M3 생성

5) 모든 분류기가 다 생성되었을 때, 각 분류기의 성능에 따라서 다시 분류기의 가중치 생성 → 성능이 좋은 분류기에는 더 큰 가중치를 부여

6) 각 분류기의 예측값에 각 가중치를 곱하여 가중평균을 낸 최종 예측값 출력

편향과 분산의 관점으로 부스팅 보기

- 배깅은 분산을 줄이는 것에 초점 두고 있지만, 부스팅은 편차를 줄이는 것에 초점을 맞춤
- 부스팅의 핵심 아이디어는 모델이 잘 예측하지 못하는 데이터를 맞추는 것에 집중 따라서 train data set에 대한 예측 성능이 뛰어날 수 있지만, overfitting 단점 존재

부스팅은 편향이 큰 모델에 굉장히 효과적이다.

AdaBoost (adaptive + Boost)

- 매 스텝마다 가중치를 이용하여 약한 모형으로부터 시작하여 이전 모형의 약점을 보완한 새로운 모형들의 선형 결합을 통해 강한 모형을 생성시키는 알고리즘

Gradient Boost (tree 기반)

- 각 분류기의 loss function을 계산하여 잔차를 최소화하여 최적화 방법 사용
- 속도가 느리고, Overfitting의 단점 존재, 이를 개선하기 위한 XGBoost, LightGBM 알고리즘 나옴.

XGBoost (Extreme Gradient Boosting)

- 기존 Gradient Boost 알고리즘에 과적합 방지를 위한 기법이 추가된 지도 학습 알고리즘
- 확실히 과적합 방지가 잘 되고, 이에 따라 예측 성능이 좋아지는 장점 존재
- 하지만, 작은 데이터(Small Data)에 대해 과적합될 가능성이 있다. 뿐만 아니라 입력 변수에 대하여 출력 변수가 어떻게 변하는지에 대한 해석이 어렵다.

LightGBM 알고리즘

- 장점
  - 굉장히 빠른 속도로 학습 하는데에 걸리는 시간이 적음
  - GPU 학습 지원
  - Categorical Features의 자동 변환과 최적 분할
- 단점
  - XGBoost와 같이 작은 data set에 대해서 과적합 가능성 있다. (일반적으로는 10,000개 이하의 데이터를 적다 라고 판단한다.)

Summary

앙상블 유형	Bagging	Boosting
공통점	전체 데이터 집합으로부터 복원 랜덤 샘플링(Bootstrap)으로 훈련 데이터 집합 생성	전체 데이터 집합으로부터 복원 랜덤 샘플링(Bootstrap)으로 훈련 데이터 집합 생성
차이점	병렬학습	순차학습

앙상블 유형	Bagging	Boosting
특징	균일한 확률분포에 의해 훈련 데이터 집합 생성(분산이 큰 모델에 적합)	분류하기 어려운 훈련 데이터 집합 생성(편차가 큰 모델에 적합)

▼ 참고자료

week5_Ensemble 강의자료.pdf - 투빅스

https://bkshin.tistory.com/entry/머신러닝-11-앙상블-학습-Ensemble-Learning-배깅Bagging과-부스팅Boosting

https://eatchu.tistory.com/entry/앙상블Ensemble-bias-variance-관점에서의-유형-정리-Voting-Bagging-Boosting

https://zephyrus1111.tistory.com/