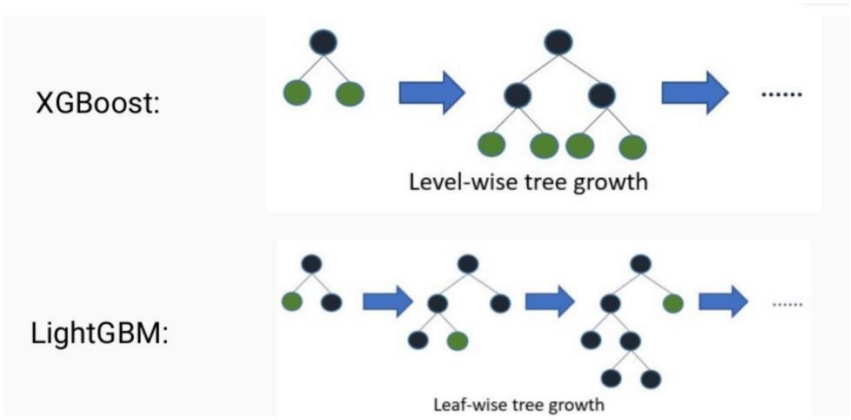
1. XGBoost (= Extra Gradient Boost)

- GBM(Gradient Boosting Machine)에서 파생된 tree based 앙상블 학습 방법으로, 실행 속도 개선을 위한 Parallelization(병렬 처리)이 가능하여 GBM 대비 학습 시간이 적게 걸림
- Tree Pruning (가지치기), Regularization (규제화), Early stopping (초기 중단)과 같은 기능이 탑재되어 있음
  - Tree Pruning: criterion-first 방식이 아닌 'depth-first' 방식으로, 'max_depth' 파라미터를 정함
  - Regularization: Lasso(L1)와 Ridge(L2) 규제를 통해서 모델에 패널티를 부여하여 Overfitting을 잡을 수 있음
  - Early stopping: 초기 중단과 같은 규제를 통해 오버피팅을 방지할 수 있음
- Cross-validation이 빌트인 되어있음
- 핵심 라이브러리가 C/C++로 작성되어 있음 (파이썬 패키지도 제공됨)

2. LightGBM

- XGBoost와 마찬가지로 GBM 기반의 알고리즘으로, 학습 시간은 느리다는 XGBoost의 단점을 보완하여 속도와 성능이 개선된 알고리즘
- GOSS(Gradient-based One-Side Sampling)이라는 메인 기술을 적용함. GOSS 는 Information gain 을 계산할 때, 큰 Gradients 를 가진 데이터 인스턴스는 유지하고 작은 Gradients 를 가진 데이터 인스턴스는 무작위로 샘플링하여 Multiplier 상수를 도입함

- GBM 계열의 트리는 level-wise (균형 트리 분할) 방식을 채택하는 반면, LightGBM 은 leaf-wise (리프 중심 트리 분할) 방식을 채택하여 트리가 깊어지면서 소요되는 시간과 메모리를 절약함



출처: https://rohitgr7.github.io/lightgbm-another-gradient-boosting/

- 대용량 데이터 처리가 가능하고 메모리를 적게 사용하며 빠르지만, 적은 수의 데이터에 대한 Overfitting 이 일어나기 쉽다는 단점이 있음

3. Catboost

1) 특징

- **Level-wise Tree**: Light GBM 은 Leaf-wise 인 반면에, XGBoost 와 Catboost 는 Level-wise 로 트리를 만들어나간다. (Level-wise 는 BFS 같이 트리를 만들어나가는 형태고, Leaf-wise 는 DFS 같이 트리를 만들어나가는 형태)

- **Ordered Boosting**: 기존의 부스팅 모델이 일괄적으로 모든 훈련 데이터를 대상으로 잔차계산을 했다면, Catboost 는 일부만 가지고 잔차계산을 한 뒤 이걸로 모델을 만들고, 그 뒤에 데이터의 잔차는 이 모델로 예측한 값을 사용

- **범주형(categorical) 피처 처리를 위한 알고리즘 도입**: 범주형 피처를 처리하는 가장 유명한 테크닉으로는 one-hot encoding 이 있음. 낮은 Cardinality 를 가지는 범주형 변수에 한해서, 기본적으로 One-hot encoding 을 시행. 예를 들어, `one_hot_max_size = 3` 으로 준 경우, Cardinality 가 3 이하인 범주형 변수들은 Target Encoding 이 아니라 One-hot 으로 Encoding 됨. 문제는 cardinality 가 높으면 어마무시하게 많은 새로운 피처가 생김. 이를 해결하기 위해 범주를 제한된 수의 클러스터로 그룹화한 다음 one-hot encoding 을 적용할 수 있음. 일반적인 방법은 각 범주의 기대 목표값을 추정하는 목표 통계량(TS)별로 범주를 그룹화하는 것.

- **Random Permutation**: Catboost 는 데이터를 셔플링하여 뽑아냄. 뽑아낼 때도 역시 모든 데이터를 뽑는게 아니라, 그 중 일부만 가져오게 할 수 있음. 이 모든 기법이 다 오버피팅 방지를 위해, 트리를 다각적으로 만들려는 시도임.

- **Optimized Parameter tuning**: XGBoost 나 LightGBM 은 파라미터 튜닝에 매우 민감하지만, Catboost 는 파라미터가 기본적으로 최적화가 잘 되어있기 때문에, 파라미터 튜닝에 크게 신경쓰지 않아도 됨. Catboost 는 이를 내부적인 알고리즘으로 트리의 다형성과 오버피팅 문제를 해결하고 있음.

2) 한계점

- Sparse 한 Matrix 는 처리하지 못함
- 데이터 대부분이 수치형 변수인 경우, Light GBM 보다 학습 속도가 느림

참고 자료:

https://dailyheumsi.tistory.com/136

https://heeya-stupidbutstudying.tistory.com/m/43