클린업 2주차

4팀 데이터마이닝

박정현
전규리
김지민
노정화
염예빈

1 차원이란?

- 1. 차원의 저주 (Curse of Dimensionality)
- 2. Tree 기반 모델

차원(dimension)의 의미

What is Dimension?

2 기하학적 관점의 차원

공간 내에 있는 점 등의 위치를 나타내기 위해 필요한 축의 개수

"선형적으로 독립인 n개의 벡터를 span하여 n차원을 만든다"

$$X = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix} = \begin{bmatrix} x_{11} & \cdots & x_{1K} \\ x_{21} & \cdots & x_{2K} \\ \vdots & \ddots & \vdots \\ x_{N1} & \cdots & x_{NK} \end{bmatrix}$$

N개의 관측치 + K개의 설명변수

Linearly independent

span

K차원의 공간

1 차원이란?

차원(dimension)의 의미

- 1. 차원의 저주 (Curse of Dimensionality)
  2. Tree 기반 모델

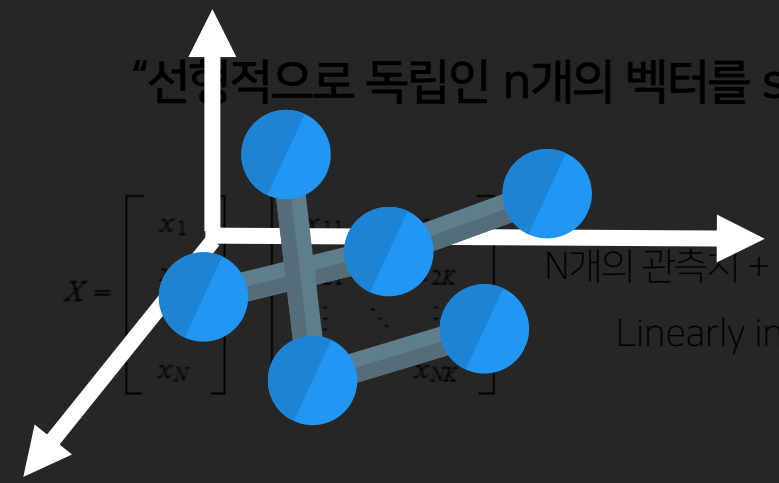
What is Dimension?

설명 변수(K)의 개수가 늘어날수록

X에 의해 span되는 **공간의 크기**가 커진다!

공간 내에 있는 점 등의 위치를 나타내기 위해 필요한 축의 개수

"선형적으로 독립인 n개의 벡터를 span하여 n차원을 만든다"



$$X = \begin{bmatrix} x_1 & & x_K \\ & \ddots & \\ x_N & & x_{NK} \end{bmatrix}$$

N개의 관측치 + K개의 설명변수

Linearly independent

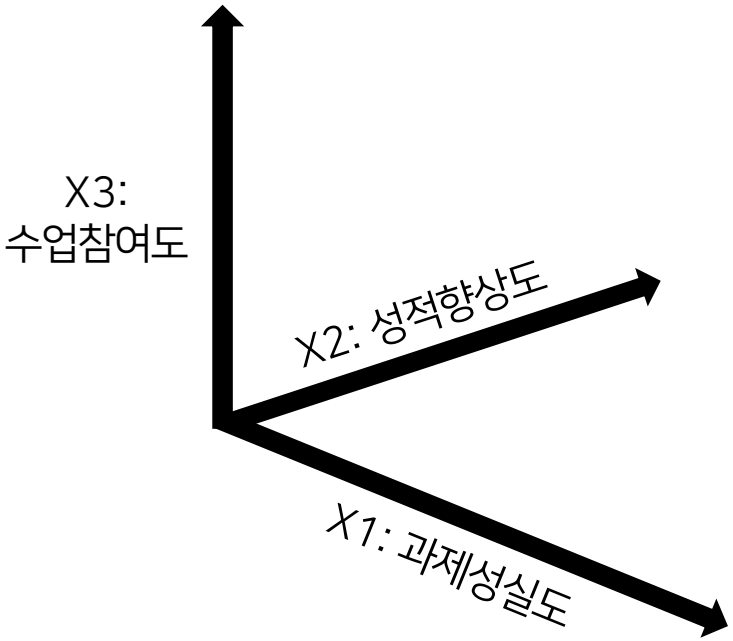
span

K차원의 공간

1 차원이란?

- 1. 차원의 저주 (Curse of Dimensionality)
- 2. Tree 기반 모델

공간의 크기가 커지는 것은 왜 문제가 되는가?



$$\frac{3}{125} \times 100 = 2.4\%$$

→ 고려해야 할 조합의 수가 125개로 늘어나
조합의 증가율이 가팔라짐

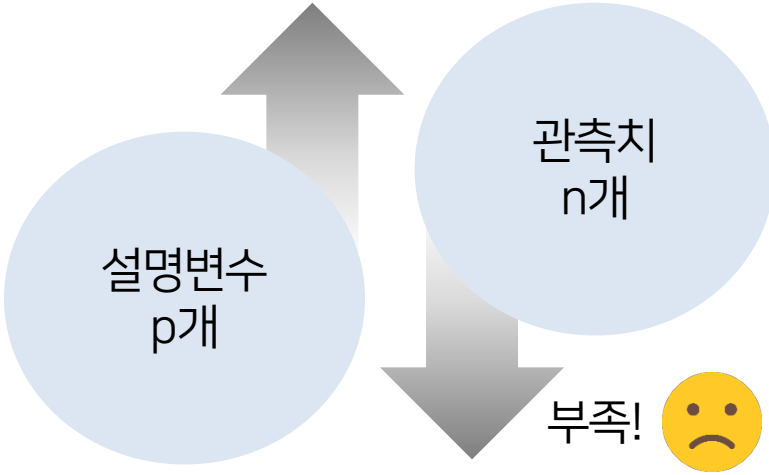
→ 3개의 샘플을 뽑았을 때
표본이 모집단에서 차지하는 비중 또한 급격히 감소

설명변수의 개수가 많아질수록 , input space(모집단)를
충분히 반영하기 위해 필요한 관측값의 개수가 기하급수적으로 증가!

차원의 저주

차원의 저주(Curse of Dimensionality)

"설명변수가 늘어나면 데이터에 대한 정보가 늘어나는 거 아닌가?"



모델이 **해당 데이터셋에만**
과적합되는 문제가 발생한다!

차원의 저주

차원의 저주(Curse of Dimensionality)

"설명변수가 늘어나면
데이터에 대한 정보가 늘어나는 거 아닌가?"

2

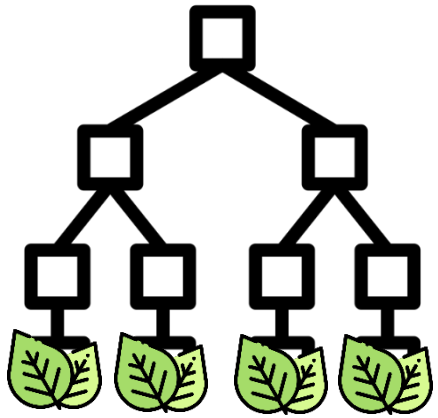
Tree 기반 모델

Models Based on Decision Tree

의사결정나무(Decision Tree)란?

What is Decision Tree?

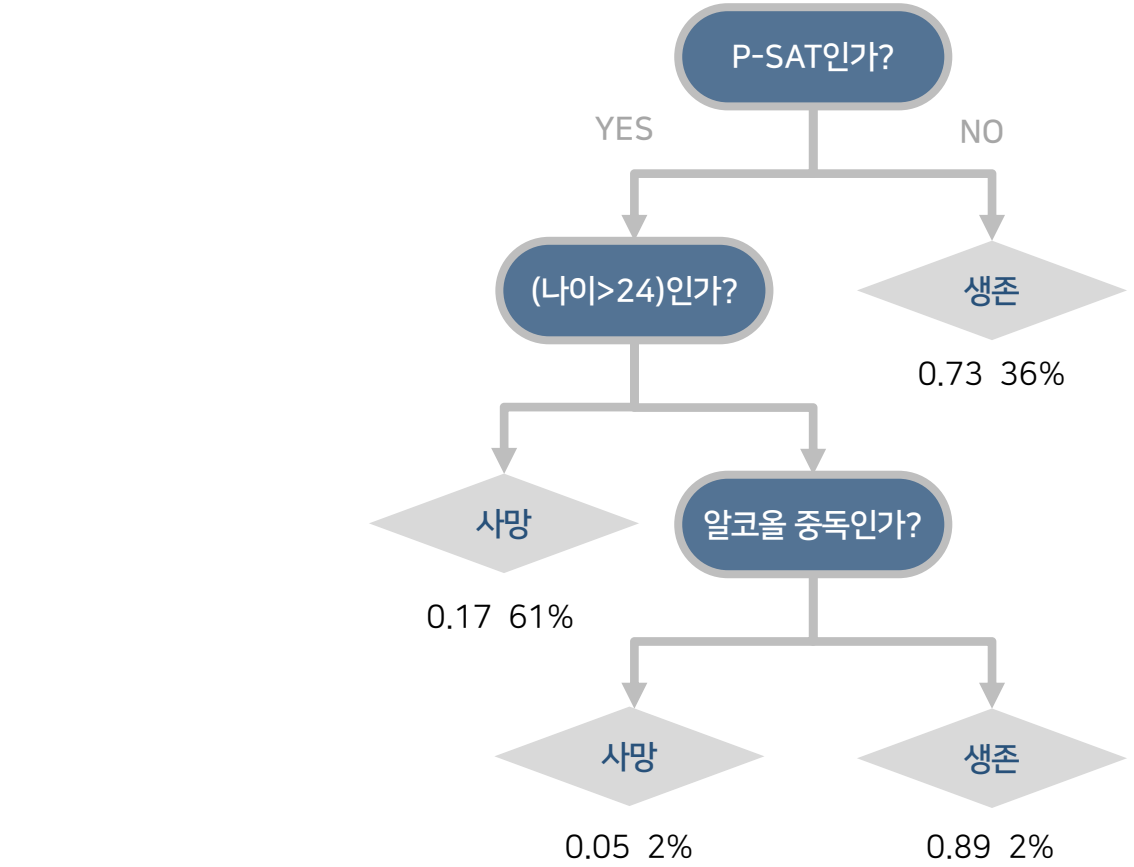
[Tree diagram]



의사결정 규칙을 나무구조로 도표화하여 분류와 회귀 문제 해결

: 예측값으로 평균값 또는 최빈값 사용

트리의 구조

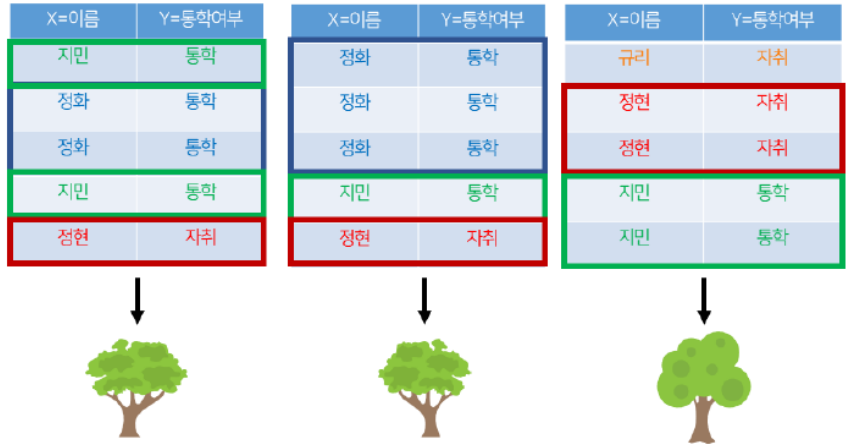


타이타닉에 탑승한 P-SAT의 생존율 예측 트리

Bagging의 한계

분산이 큰 **서로 다른 트리를 앙상블**시키는 방법이기 때문에
의사결정나무의 **분산을 감소**시키는 효과가 있는 것처럼 비춰짐

But, 1주차에서 살펴봤던 것처럼 각각의 부트스트랩 샘플들이 서로 너무 비슷함

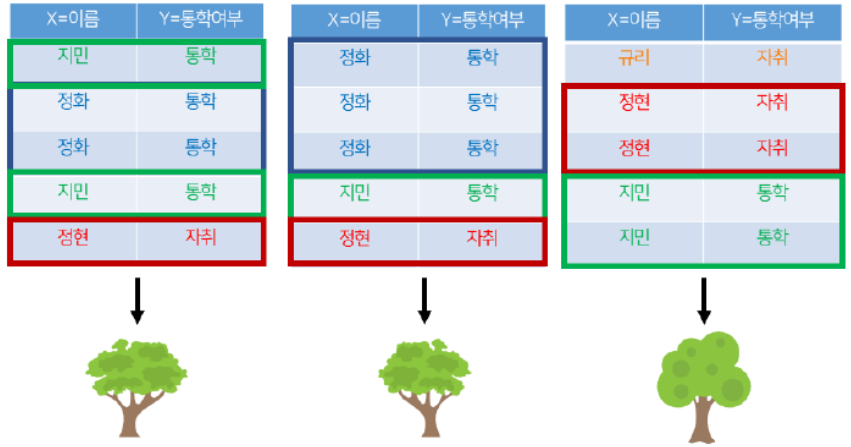


Bagging의 한계

1) 부트스트랩의 특성상 복원추출 시행

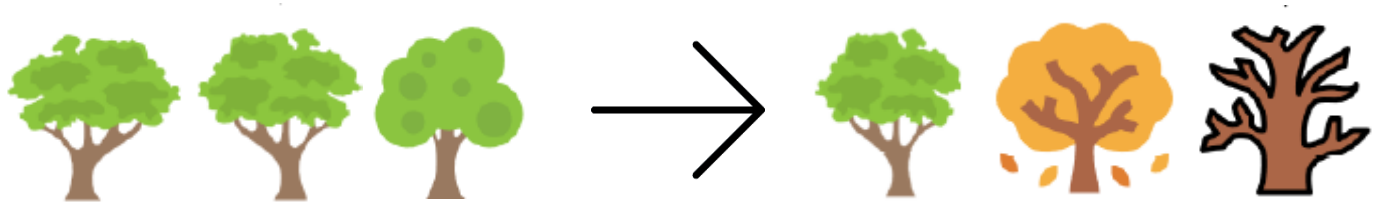
2) 각각의 부트스트랩 샘플에 적합된 트리 모양이 비슷함

3) 적합된 트리들이 **양의 상관 관계**를 갖게 되는 결과



Bagging의 한계

So, 위와 같은 문제점을 해결하기 위해서는
트리 간의 공분산을 줄여 Decorrelated Trees를 만들어야 한다.



이를 보완하기 위해 착안된 Random Forest

Random Forest

“앙상블 모델 Ensemble Model”

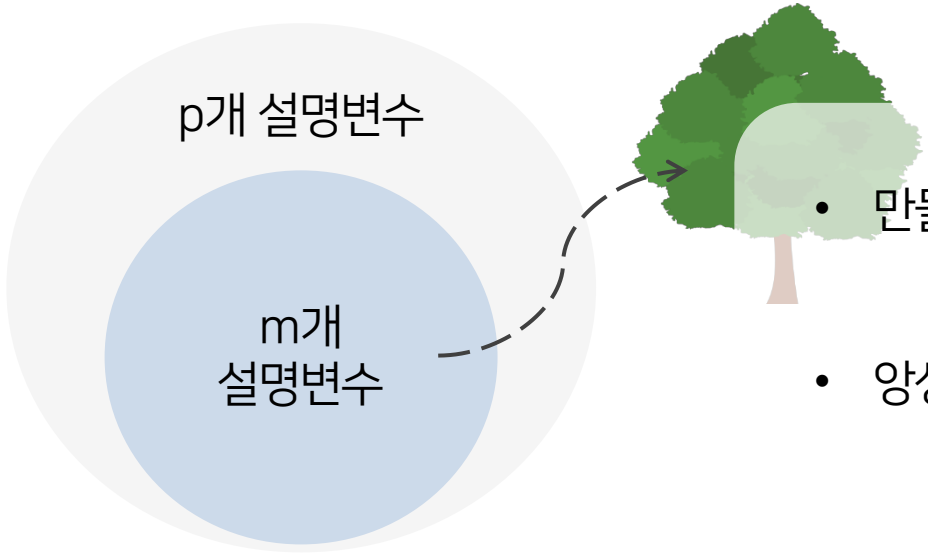
Bagging (배깅)

Random Forest (랜덤 포레스트)

Boosting Model (부스팅)

Random Forest

"샘플이 너무 유사해서 모델도 유사하다면,
모델 적합 시 쓰일 **X변수들을 랜덤으로 뽑아** 모델을 다르게 만들자!"

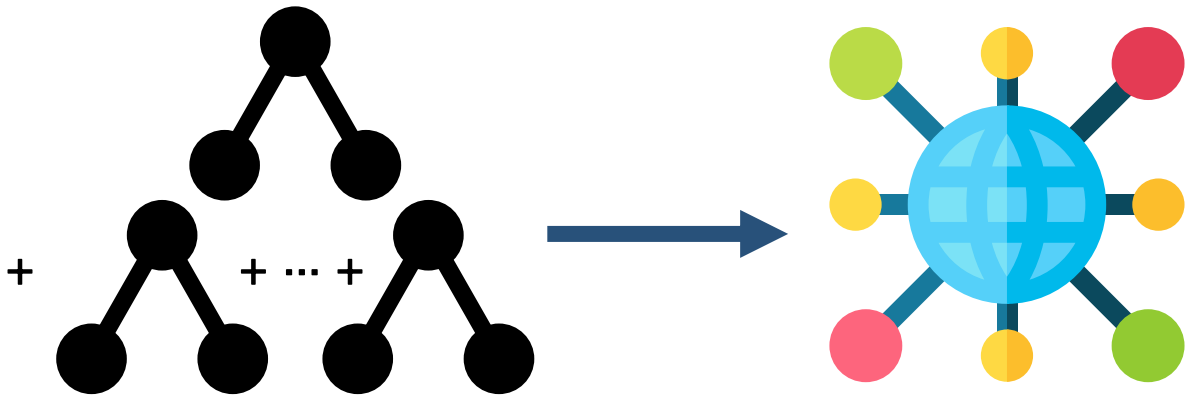


- 만들어진 트리 간 상관관계 감소
- 앙상블된 모델의 분산 크게 감소

부스팅(Boosting) 기법이란?

What is BOOSTING?

약한 학습기를 반복적으로 결합하여 강한 학습기를 만드는 방법



약한 학습기
(Weak Learner)

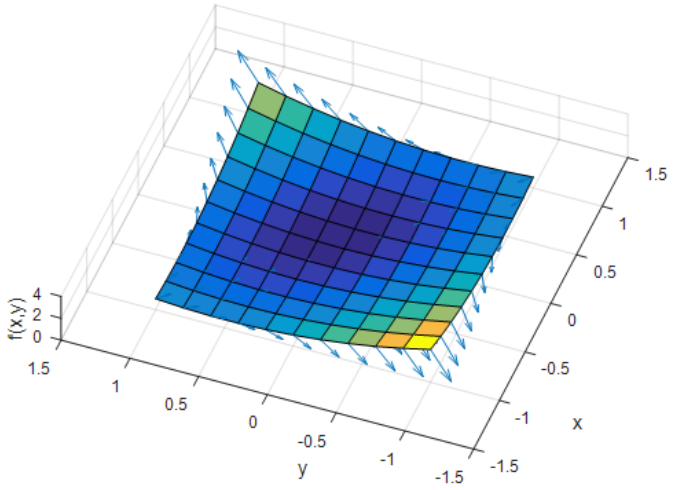
강한 학습기
(strong Learner)

Gradient Boosting Model(GBM)

What is Gradient?

Gradient : 접선의 기울기

∇f : 3차원 공간의 스칼라장 f와 어떤 점 p에 대해서 각 세 축에 대한 편미분

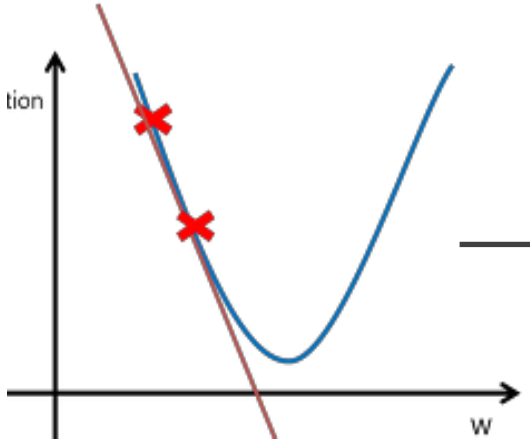


$$\nabla f = \frac{\partial f}{\partial x}\underline{e}_x + \frac{\partial f}{\partial y}\underline{e}_y + \frac{\partial f}{\partial z}\underline{e}_z$$

Gradient Boosting Model(GBM)

Gradient Descent? 함수 f 에서 step을 옮겨가며 최솟값을 찾는 방법

STEP 3.



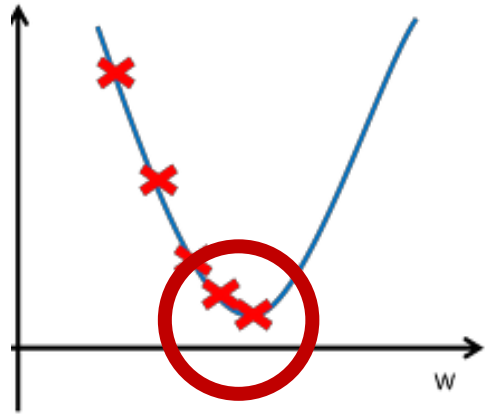
Learning rate를 곱한 값을 초기 w값에 더해줌

이 과정 반복

...

W값 업데이트

STEP 4.

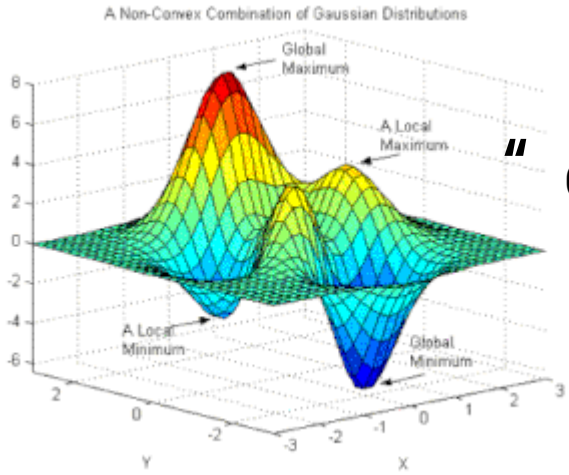


최솟값을 찾게 됨

Gradient Boosting Model(GBM)

Gradient Boosting Model

Gradient Descent + Boosting



" Gradient Descent 의 원리를 부스팅기법에 적용,

의사결정나무 기반의 부스팅모델 "

Gradient Boosting Model(GBM)

Gradient Boosting Model : Why residual?

왜 예측값을 적합시키지 않고, 잔차에 적합시킬까?

Q.

A.

손실함수를 squared error로 설정했을 때,

negative gradient를 구하면 잔차가 되기 때문!

$$\frac{\partial j(y_i, f(x_i))}{\partial f(x_i)} = \frac{\partial\left[\frac{1}{2}(y_i - f(x_i))^2\right]}{\partial f(x_i)} = f(x_i) - y_i$$

- Negative gradient를 사용하여
- 손실함수를 최소화하고자 하는 문제를 해결 가능함

Gradient Boosting Model(GBM)

Gradient Boosting Model : Learning rate

Q.

Learning rate란?

A.

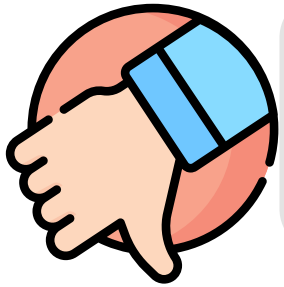
GBM의 Learning rate = 예측된 잔차가 기존의 예측값에 더해질 때, 잔차 앞에 곱해지는 작은 상수값

$$F_m(x) = F_{m-1}(x) + \gamma_m h_m(x)$$

Boosting기법의 장단점

장점 Advantage

머신러닝 기법 중 성능이 가장 좋음, 빅데이터에 사용가능



단점 Drawbacks

블랙박스 모델이라 해석 거의 불가,
파라미터에 민감하기에 튜닝이 필수적

LGBM

LightGBM

What is LGBM?

LightGBM

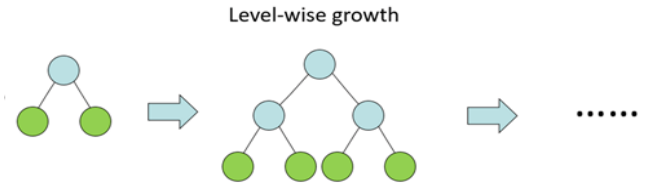
- 2016년 Microsoft에 의해 개발된 GBM계열 알고리즘
- 다른 모델들에 비해 좀더 빠르고 메모리를 적게 차지(Light),성능도 좋음.
- Kaggle 에서 가장 많이 우승한 알고리즘!

LGBM

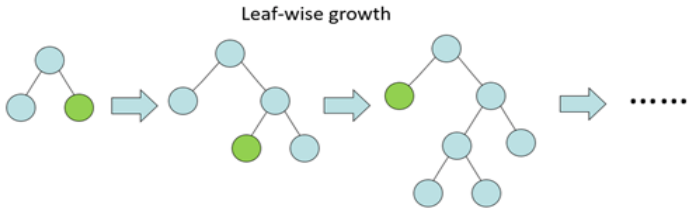
LightGBM

Why "Light"GBM?

XGBoost
Catboost



LGBM



1 수평적으로 분기(level-wise)하는 다른 모델과 달리 수직적으로 분기(leaf-wise)

→ 트리의 균형을 맞추는 데 필요한 연산 감소

2 히스토그램기반의 알고리즘, 히스토그램을 관측함으로써 근사치로 분할을 수행

→ 각 split에서 split point를 찾는 계산 감소.