# 머신러닝(Machine Learning)

# 목 차



# 1-1 Random Forest

(가) 2001년에 레오 브라이만(Leo Breiman)이 제안한 머신러닝 알고리즘

(나) 학습 데이터에 따라 생성되는 모델이 달라지기 때문에 일반화하는데 어려움이 있다.

(다) 랜덤포레스트에서는 결정트리를 학습시킬 때 변수 선택의 임의성, 배깅(예제 선택의 임의성을 통해 이러한 문제점을 극복하였다.

# 1-1 Random Forest

(라) 이런 임의성으로 인해 조금씩 다른 특성을 갖는 트리들로 구성되기 때문에 각 트리들이 서로 **상관성이 없어지게 되며**, **일반화 성능이 향상**된다.

(마) 만약 학습 데이터가 노이즈가 포함된 데이터라 하더라도 노이즈의 영향을 덜 받는 모델을 만들어 주게 된다.

# 1-2 Random Forest 구성

## Bagging을 이용한 구성과 Random subspace 방법

**(가) Bagging 을 이용한 나무 구성**

**(나) Random subspace를 이용한 나무 구성**

# 1-2 Random Forest 구성

## Bagging을 이용한 Tree 구성(부트 스트래핑 – Bootstrapping)

**(가) 100개의 observation(관측치)의 데이터 셋이 있다. 숲을 구성하기 위해 100개의 나무가 필요하다.**
**그렇다면 우리는 각각 1개의 나무당 100개의 샘플이 필요하다.**

**즉, 10000개의 샘플이 필요하다.**
**⇒100개의 기본 데이터 셋에서 무작위 복원 추출을 이용하여 나무의 개수만큼 데이터 셋을 만든다.**
**이 경우 고유한 샘플은 약 63.2% 정도라고 한다.**

# 1-2 Random Forest 구성

## Random subspace 방법 – Feature Bagging

**왜?** 한 개의 특징 또는 극소수의 특징들이 결과에 대한 강한 예측을 지닌다면, 훈련 과정 중에 여러 트리노드에서 이러한 특징들이 중복되어 선택되고 결과적으로 소수의 특징들이 데이터를 분류하는데 여러 트리에서 중요한 특징이 될 것이다.

변수의 개수(m)를 선택하는 방법.
총개수 M,
(1) sqrt(M)
(2) floor(lnM + 1)

# 1-3 Random Forest 결과값

## 최종 결과 산출 방법

**회귀 트리의 경우**
연속적인 값이 나왔다면 이를 평균을 낸다.

**분류 트리의 경우**
이산적인 값이 많이 나왔다. 가장 많이 나온 친구들을 뽑는다.

=> 위의 이 과정을 우리는 Aggregating라고 한다.

# 1-4 Random Forest 성능 평가

OOB(out-of-bag) error라는 수치를 이용하여 성능 평가 수행

**일반적으로 고유한 샘플 비율이 약 63.2%**

**그렇다면 뽑히지 않은 데이터 비율은 약 36.8%이다.**

**이 예제들을 따로 랜덤포레스트로 학습 시켜서 결과적으로 값이 OOB Error이다.**

# 1-4 Random Forest 성능 평가

## OOB(out-of-bag) error라는 수치를 이용하여 성능 평가 수행

(가) OOB가 중요하다.

Breiman(1996b)의 Bagging 분류기들의 오차 측정에 대한 연구에서 OOB 예측방법이 훈련 데이터셋의 크기와 같은 테스트셋을 써서 검증을 한 것만큼 정확하다는 것 때문에 OOB 측정을 통해 test셋을 구성할 필요가 없어짐.

# 1-5 Random Forest Variable

랜덤 포레스트를 구성하는데 중요한 매개변수

(가) 포레스트의 크기

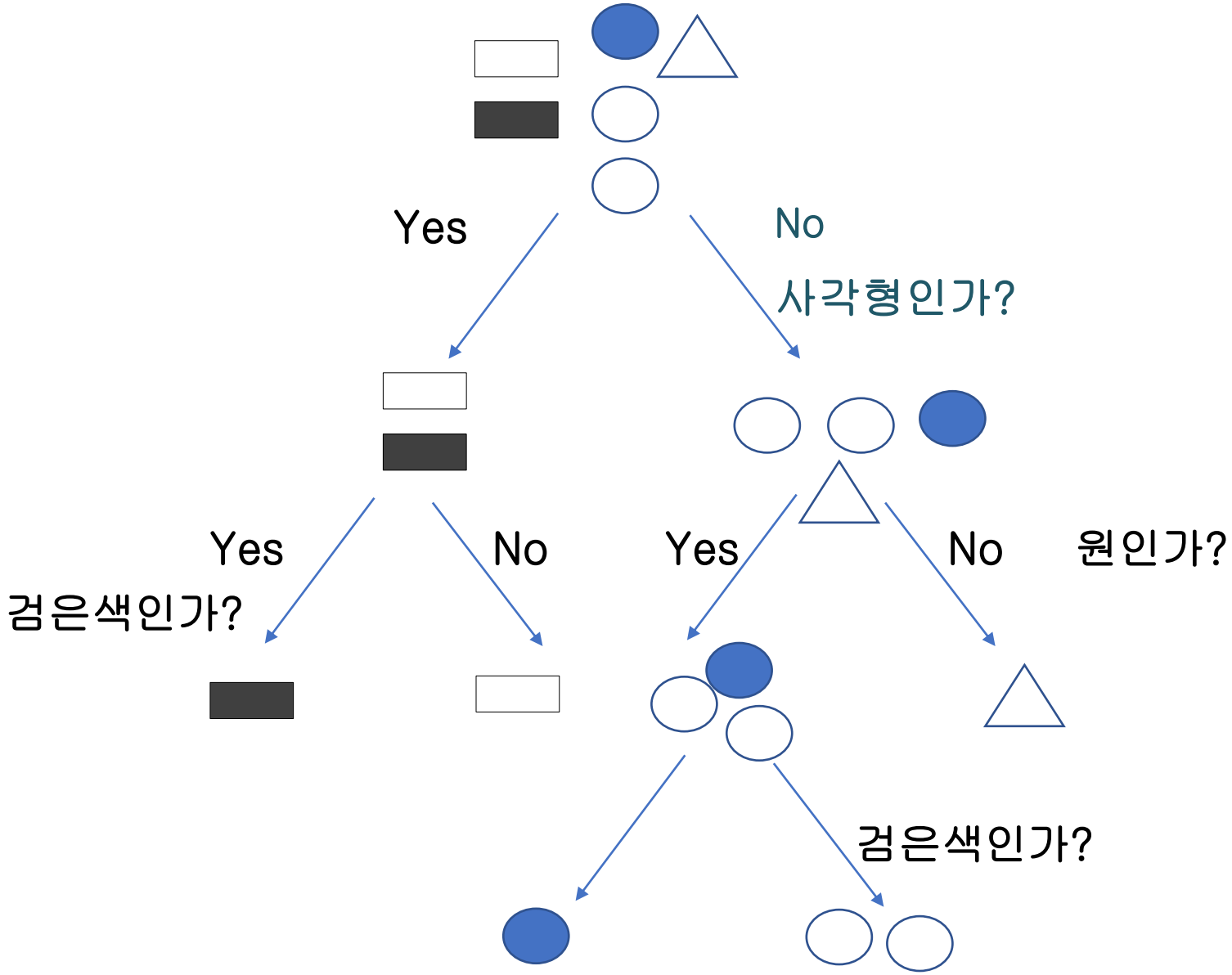
(나) 최대 허용 깊이

(다) 임의성의 정도와 종류

(라) 노드 분할 함수의 선택

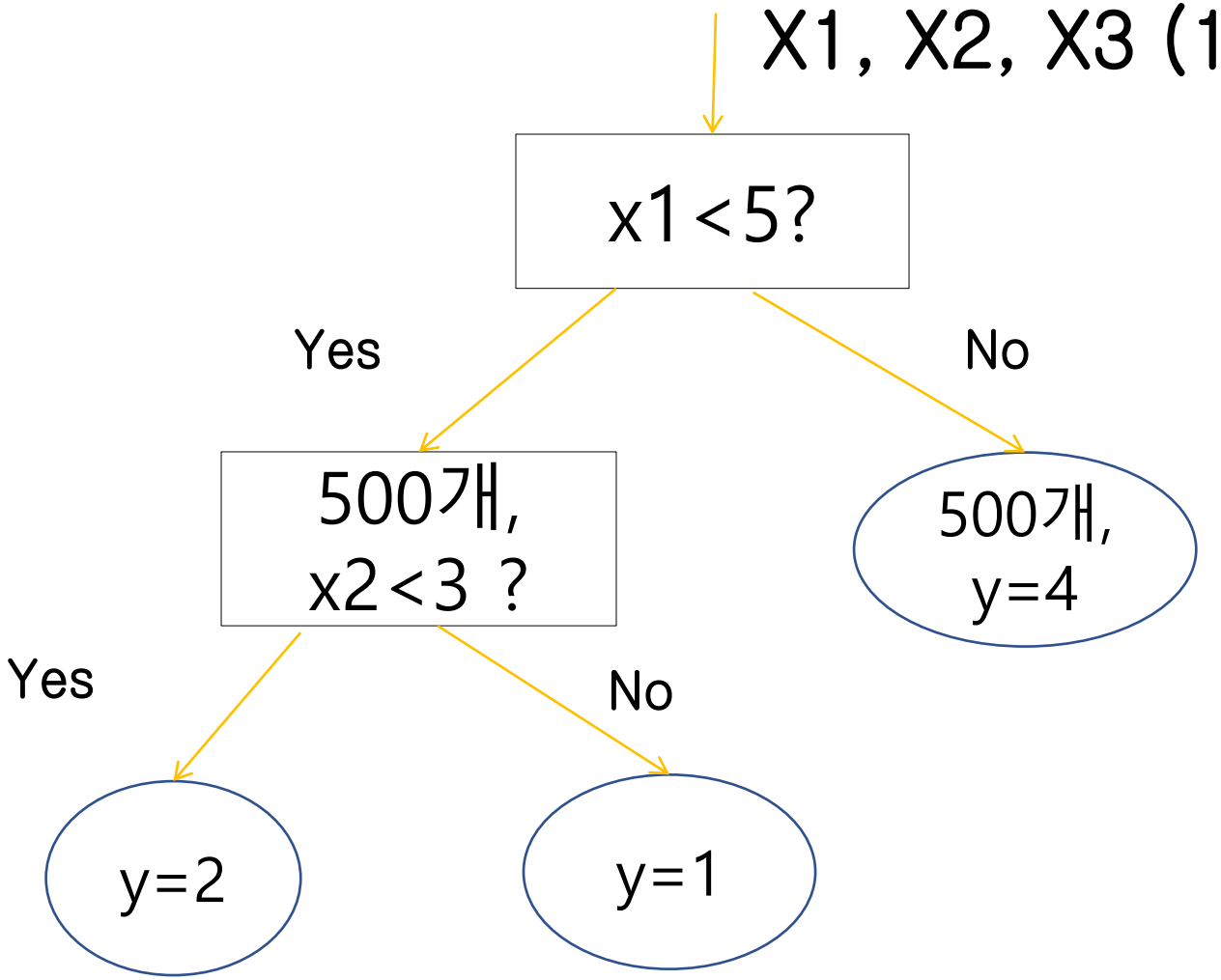
 

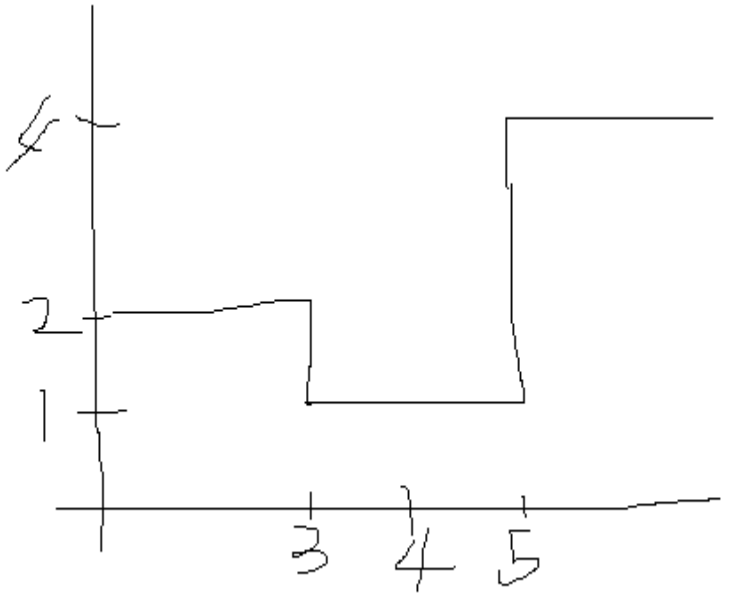
# 1-6 Decision Tree model



# 1-7 Binary Decision Tree model



# 1-8 배깅(Bagging)