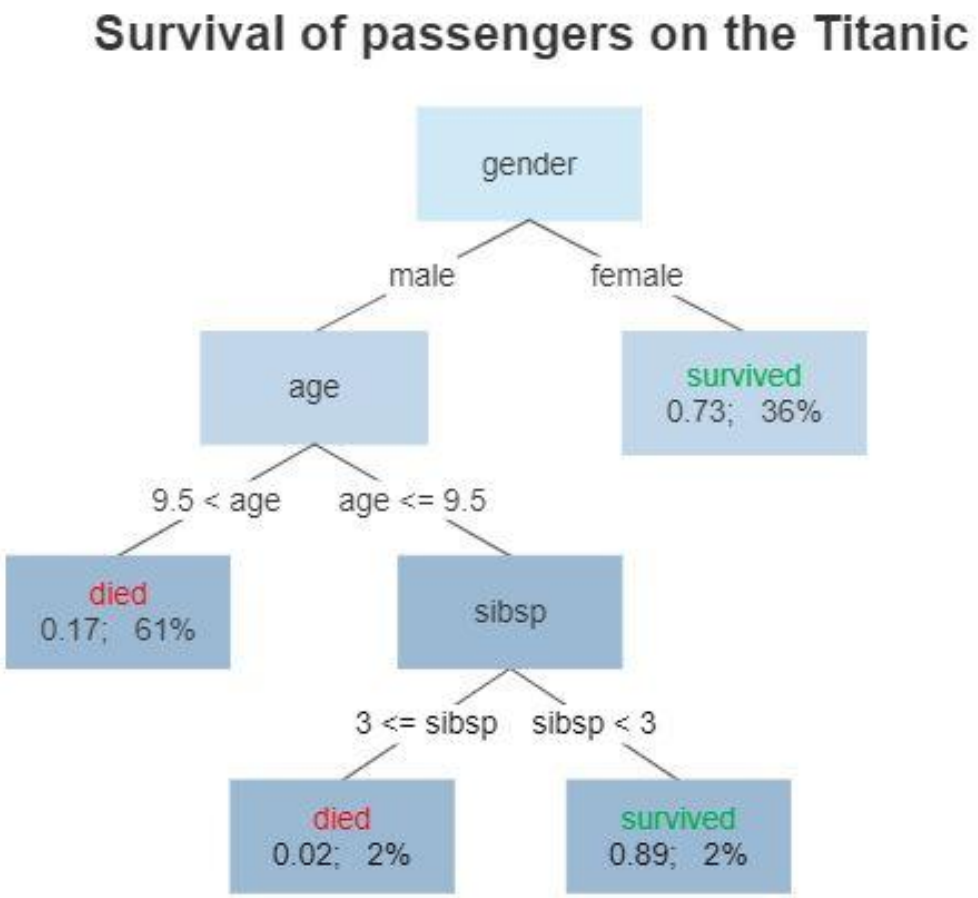
x

K-Digital Training 스마트 팩토리 3기

의사결정 나무
(Decision trees)

의사결정 나무란?

- 주어진 입력값들의 조합에 대한 의사결정규칙(rule)에 따라 출력값을 예측하는 모형
- 트리구조의 그래프로 표현
- 예측력은 다른 지도학습 기법들에 비해 대체로 떨어지나 해석이 수월
- 분류나무와 회귀나무가 있음
- 의사결정 나무는 전통적인 머신러닝기법이며 딥러닝이 아님



용어

- 뿌리노드(root node) : 최상단 시작되는 node
- 부모노드(parent node) : 주어진 마디의 상위 마디
- 자식노드(child node) : 부모로부터 분리되어 나간 2개 이상의 마디들
- 끝노드(leaf node) : 자식마디가 없는 마디
- 중간노드(internal node) : 부모와 자식이 모두 있는 마디
- 가지(branch) : 뿌리노드부터 끝노드까지 연결된 마디들
- 깊이(depth) : 뿌리노드부터 끝노드까지의 중간마디들의 수

절차

1. 성장(tree growing) : 최대 크기의 나무 모형 형성
   - 각 마디에서 적절한 최적의 분리규칙을 찾아서 성장
   - 적절한 정지규칙을 만족하면 중단
2. 가지치기(pruning) : 최대 크기 나무모형에서 불필요한 가지 제거
3. 최적 나무 모형 선택 : 최적 모형 선택
   - 검증오차가 가장 작은 의사결정나무 평가
4. 해석 및 예측 : 구축된 나무모형 해석, 예측

데이터와 모형

- 전체 입력 공간의 분할 : M개의 영역 $R_1,\cdots,R_M$
- 출력값(y) 적합
  - 연속형 출력변수인 경우, 분할된 영역(R_m) 별로 그 영역의 평균으로 적합(예측)
  - 출력변수가 범주형인 경우 분할된 영역에 속한 출력값들 중 가장 많은 수의 범주값(최빈값) 또는 각 범주에 대한 확률로 적합(예측)함

분할 기준

- 분류모형
  - 노드 t에서 타겟값(y_i)이 j인 확률 추정치(비율)을 $p_j(t)$라고 할 때
  - 지니(gini)지수
    - 0~1 사이의 값,
    - 0 : 모든 데이터가 동일한 범주,
    - 1 : 데이터가 모든 범주에 골고루 분포

$$\text{gini}(t) = 1 - \sum_{j=1}^{J} p_j^2(t).$$

  - 엔트로피(entropy) 지수

$$\text{entropy}(t) = -\sum_{j=1}^{J} p_j(t) \log_2 p_j(t).$$

분할 기준

- 회귀모형
  - MSE(오차제곱합)

$$\mathrm{MSE}(t) = \frac{1}{n_t} \sum_{i=1}^{n_t} (y_i(t) - \bar{y}_t)^2.$$

분할 기준

- 불순도(impurity)의 감소량이 최대가 되는 최적 분리 기준에 의해 입력 공간을 분할

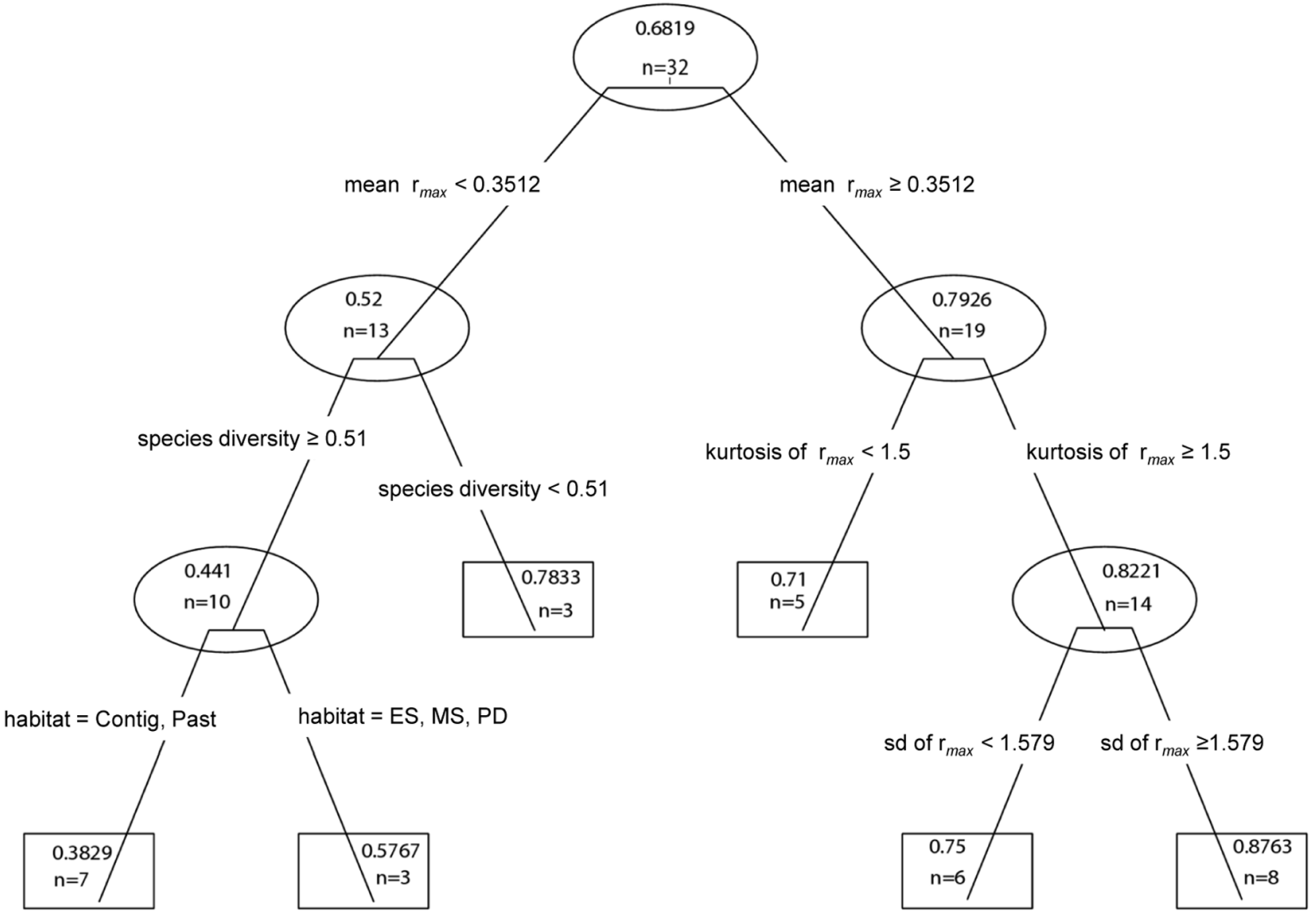
$$\Delta i(t) = \frac{N_t}{N}\left(i(t) - \frac{N_{t_R}}{N_t} i(t_R) - \frac{N_{t_L}}{N_t} i(t_L)\right)$$

$i(t)$, $i(t_L)$, $i(t_R)$: 부모노드, 왼쪽, 오른쪽 자식노드의 불순도

N : 전체 학습 데이터의 수

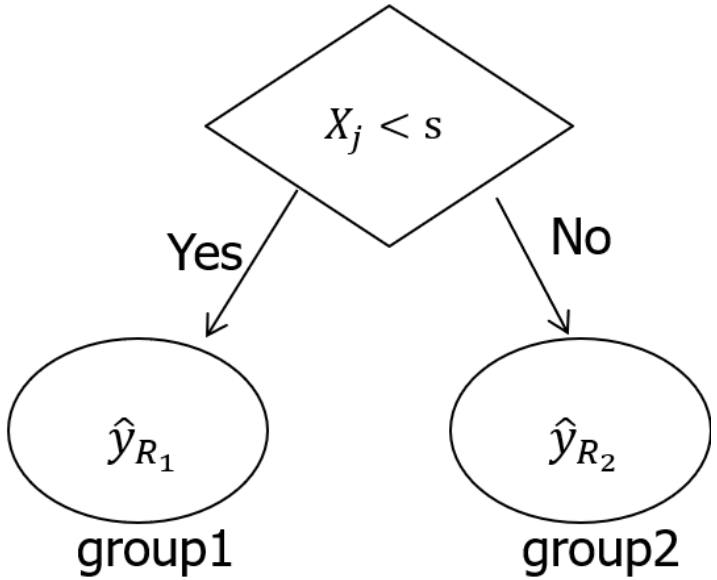
N_t, N_{tL}, N_{tR} : 부모노드, 왼쪽, 오른쪽 자식노드의 데이터 수

회귀모형 예제

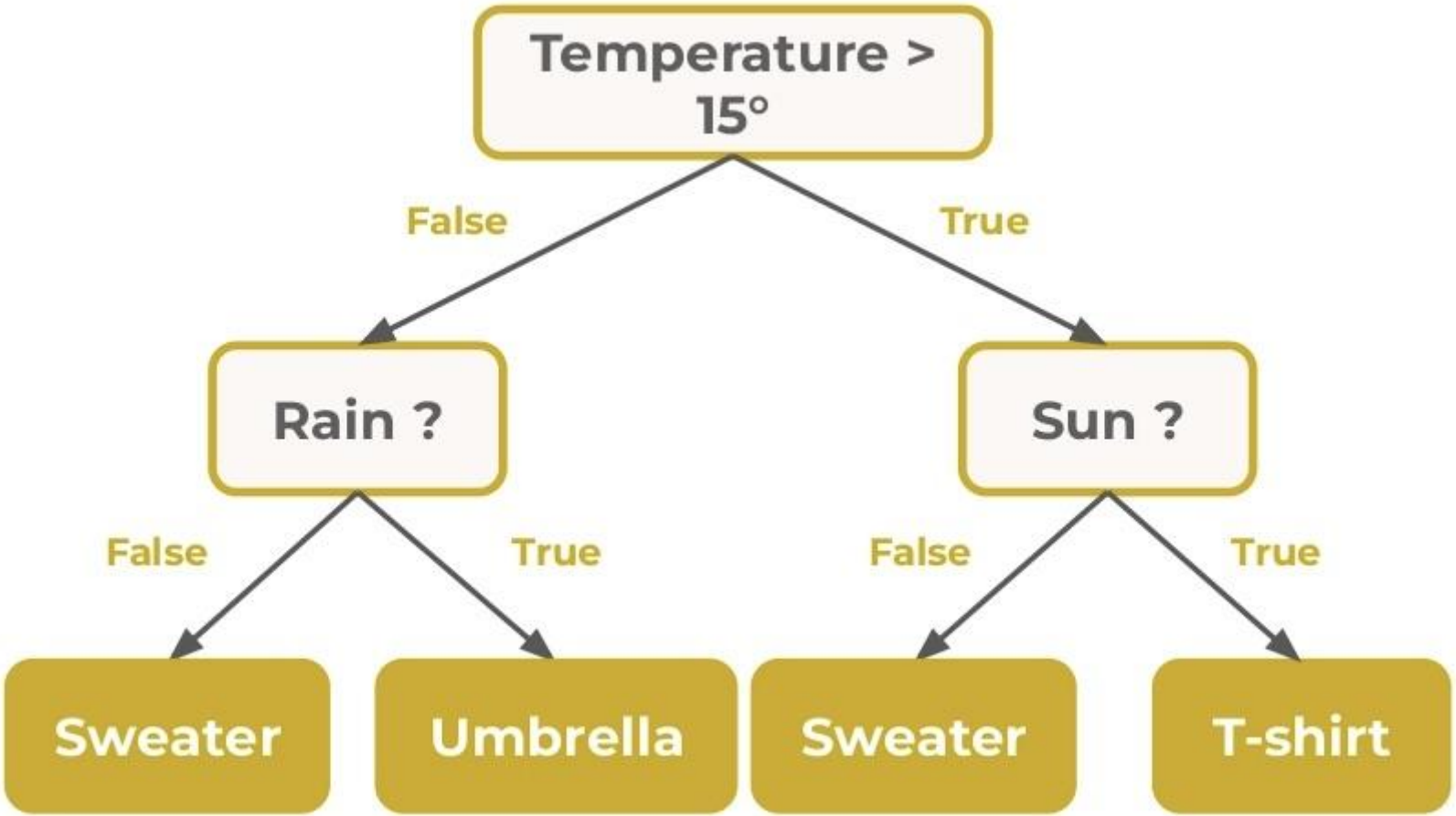

0.6819
n=32
mean r_{max} < 0.3512
mean r_{max} ≥ 0.3512
0.52
n=13
0.7926
n=19
species diversity ≥ 0.51
species diversity < 0.51
kurtosis of r_{max} < 1.5
kurtosis of r_{max} ≥ 1.5
0.441
n=10
0.7833
n=3
0.71
n=5
0.8221
n=14
habitat = Contig, Past
habitat = ES, MS, PD
sd of r_{max} < 1.579
sd of r_{max} ≥1.579
0.3829
n=7
0.5767
n=3
0.75
n=6
0.8763
n=8

회귀모형

- 각각의 그룹끼리는 모두 동일한 값 예측
- 각 그룹의 오차값을 최소화 하는 방향으로 나눔
- min(group1 오차 + group2 오차)
- greedy 한 방법으로 모든 값을 테스트
- s를 정하는 방식은 연속형/범주형 에 따라 다름
  - 연속형
    - 내림/오름 차순으로 정렬후에, 이웃하는 두개의 값의 평균을 우선적으로 테스트
  - 범주형
    - 중간값을 그대로 사용



분류모형 예제


Temperature > 15°
False
True
Rain ?
Sun ?
False
True
False
True
Sweater
Umbrella
Sweater
T-shirt

분류모형

- 지니 지수 예제
  - 공이 10개가 있고 6개가 파란색, 4개가 노란색일때
    - 파란색일 확률 = 6/10
    - 노란색일 확률 = 4/10
    - $Gini(S) = 1 - (0.6^2 + 0.4^2) = 0.48$

분류모형

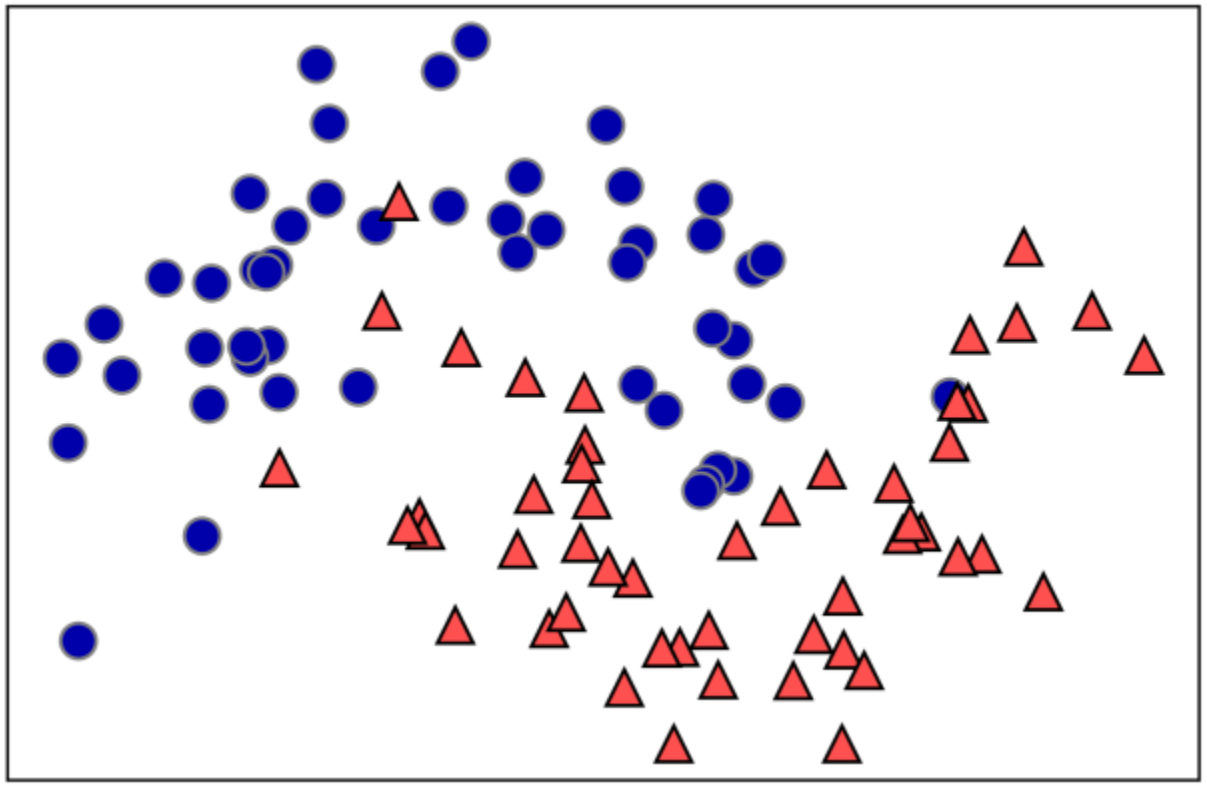
- 공의 종류로 분류했다고 하면
  - 야구공 : 파란색 4개, 노란색 1개
  - 축구공 : 파란색 2개, 노란색 3개
  - Gini(S_1) = 1- ((4/5)^2 + (1/5)^2) = 0.32
  - Gini(S_2) = 1- ((2/5)^2 + (3/5)^2) = 0.48
  - IG(S,X) = Gini(S) – (5/10*(Gini(S_1) + 5/10*(Gini(S_2))
    = 0.48 – ((0.5 * 0.32) + (0.5 * 0.48)) = 0.08
  - 데이터 세트를 분할하기 위한 정보 이득(지니지수의 감소)은 0.08

분류모형

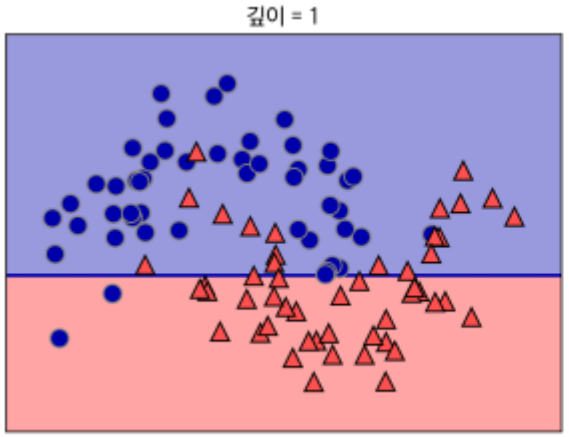
- 공의 주인으로 분류했다고 하면
  - 철이 : 파란색 6개
  - 영희 : 노란색 4개
  - Gini(S_1) = 1 - ((6/6)^2 + (0/6)^2) = 0
  - Gini(S_2) = 1 - ((0/4)^2 + (4/4)^2) = 0
  - IG(S,X) = Gini(S) - (6/10 * (Gini(S_1) + 4/10 * (Gini(S_2))
    = 0.48 - ((0.6 * 0) + (0.4 * 0)) = 0.48
  - 데이터 세트를 분할하기 위한 정보 이득(지니지수의 감소)은 0.48
- 정보이득(Information Gain)이 높은 쪽을 선택

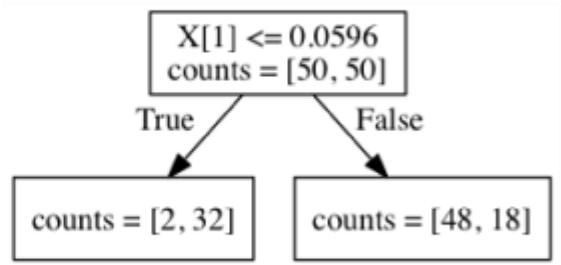
결정 트리 과정

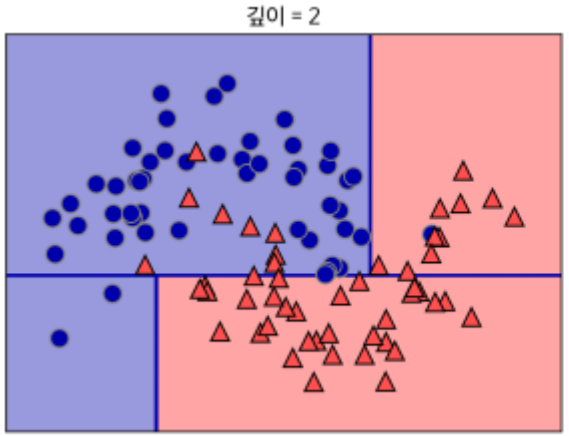
- 아래의 데이터를 나눈다고 할때

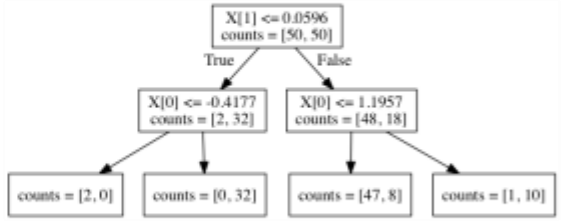


결정 트리 과정

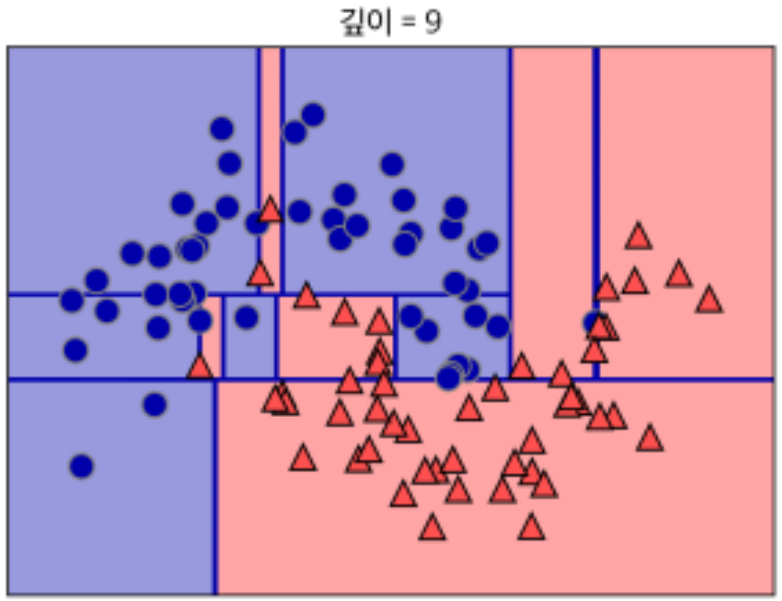

깊이 = 1


X[1] <= 0.0596
counts = [50, 50]
True
False
counts = [2, 32]
counts = [48, 18]


깊이 = 2


X[1] <= 0.0596
counts = [50, 50]
True
False
X[0] <= -0.4177
counts = [2, 32]
X[0] <= 1.1957
counts = [48, 18]
counts = [2, 0]
counts = [0, 32]
counts = [47, 8]
counts = [1, 10]

결정 트리 과정



가지치기(pruning)

- 너무 큰 나무모형 – 과대적합
- 너무 작은 나무모형 – 과소적합
- 최적의 나무모형 크기 결정 필요

가지치기(pruning)

- 각 나무모형에서 비용-복잡도(cost-complexity) 계산

$$C_\alpha(T_k) = \sum_{t=1}^{|T_k|} N_t i(T_k) + \alpha|T_k|$$

  - T_K : 최대크기 나무모형
  - $|T_k|$: 나무모형의 끝마디 개수

- 비용-복잡도 가지치기
  - 비용-복잡도가 가장 작은 나무모형을 선택

$$T^* = \arg \min_{T_0, T_1, \ldots, T_K} C_\alpha(T_k)$$

개별 트리 모델의 단점

- 계층적 구조로 인해 중간에 에러가 발생하면 다음 단계로 에러가 전파
- 학습 데이터의 미세한 변동에도 최종 결과에 영향이 크다
- 적은 개수의 노이즈에도 크게 영향
- 나무의 최종 노드 개수를 늘리면 과적합
- 해결 -> 랜덤 포레스트