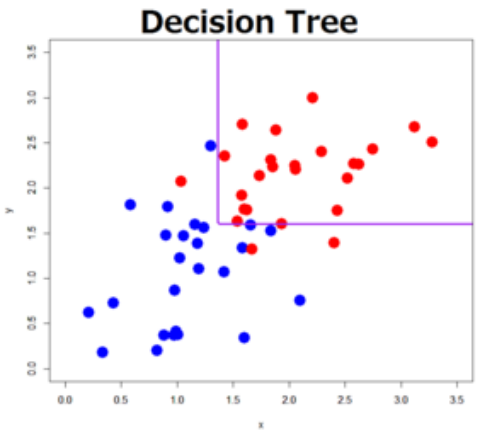
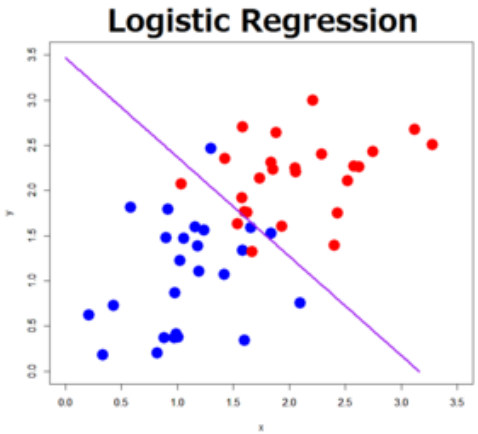
Decision Tree: Classification and Regression Tree

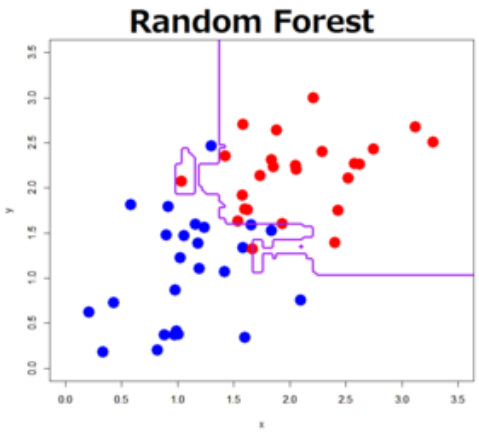
강필성
고려대학교 산업경영공학부
Blfysoft & WIGO AI LAB

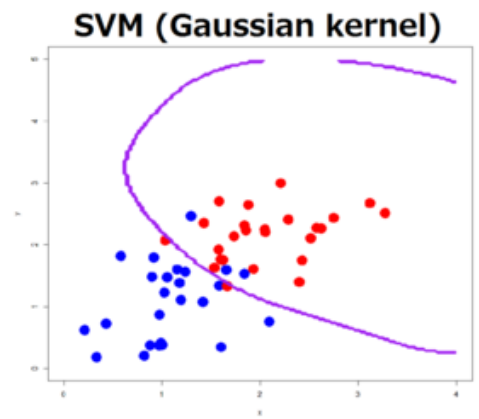
여러 가지 머신러닝 알고리즘이 필요한 이유?

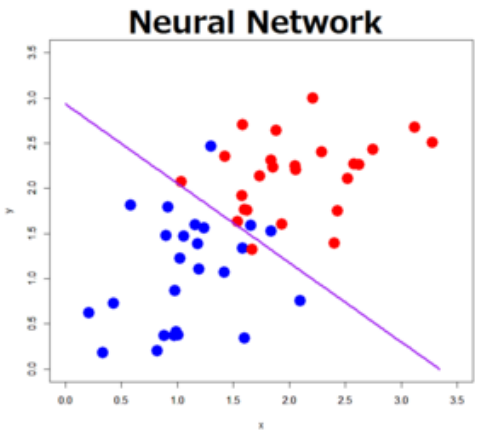
- 특정 알고리즘이 모든 상황에서 다른 알고리즘보다 우월하다는 결론을 내릴 수 없음







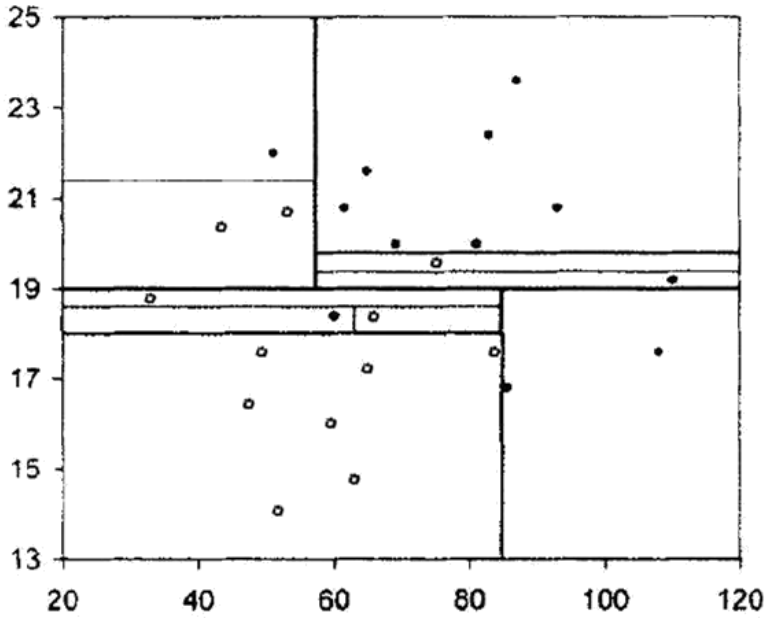


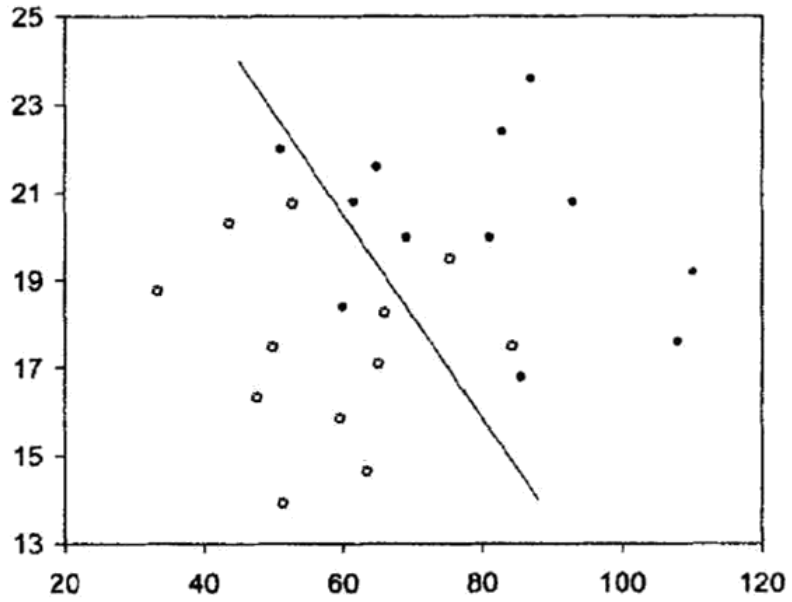


여러 가지 머신러닝 알고리즘이 필요한 이유?

- 머신러닝 알고리즘은 여러 가지가 존재
  - ✓동일한 결과를 얻기 위한 다양한 길이 존재하기 때문

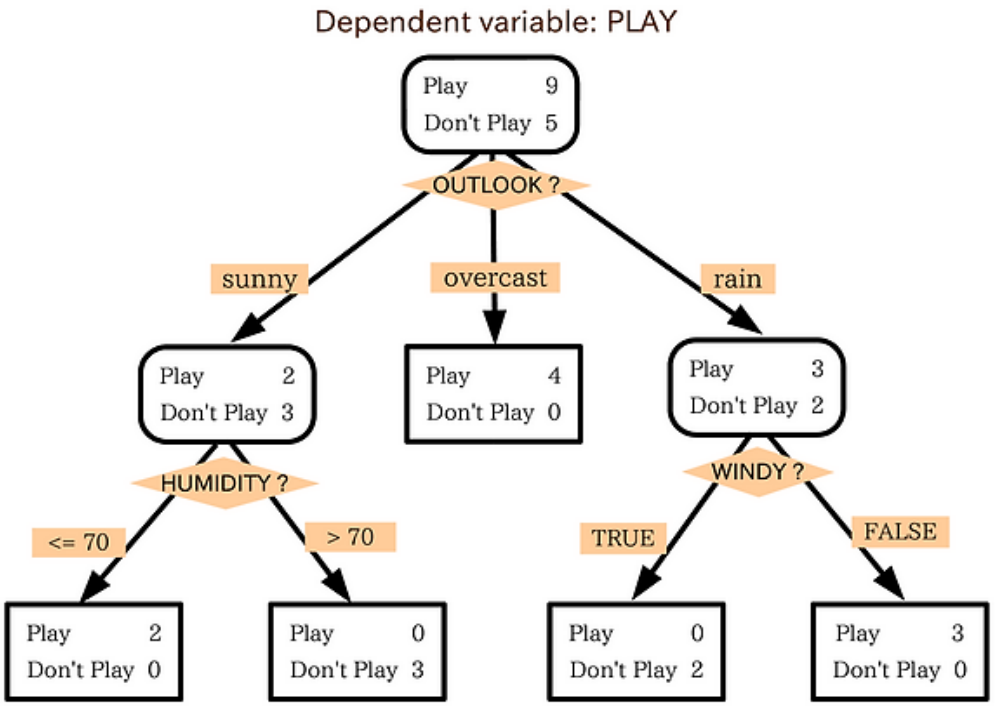
"Separate the riding mower buyers(●) from non-buyers(○)"





Decision Tree

- **목적**
  - ✓한 번에 하나씩의 설명변수를 사용하여 정확한 예측이 가능한 규칙들의 집합을 생성
  - ✓최종 결과물은 나무를 뒤집어 놓은 형태인 규칙들의 집합



[규칙 예시]

만일 내일 날씨가 맑고
습도가 70% 이하이면
아이는 밖에 나가서 놀
것이다.

or

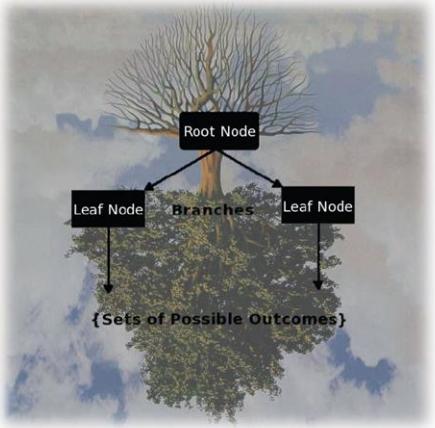
만일 내일 비가 오고
바람이 불면 아이는 밖에
나가 놀지 않을 것이다.

Decision Tree

- 왜 의사결정나무인가?
  - ✓결과를 사람이 이해할 수 있는 규칙의 형태로 제공함
  - ✓데이터의 사전 전처리를 최소화함 (정규화/결측치 처리 등을 하지 않아도 됨)
  - ✓수치형 변수와 범주형 변수를 함께 다룰 수 있음

- 핵심 아이디어: Key Ideas
  - ✓재귀적 분기: Recursive Partitioning
    - 입력 변수의 영역을 두 개로 구분 → 구분하기 전보다 구분된 뒤에 각 영역의 순도 (purity, homogeneity)가 증가하도록
  - ✓가지치기: Pruning the Tree
    - 과적합을 방지하기 위하여 너무 자세하게 구분된 영역을 통합

Classification and Regression Tree: CART

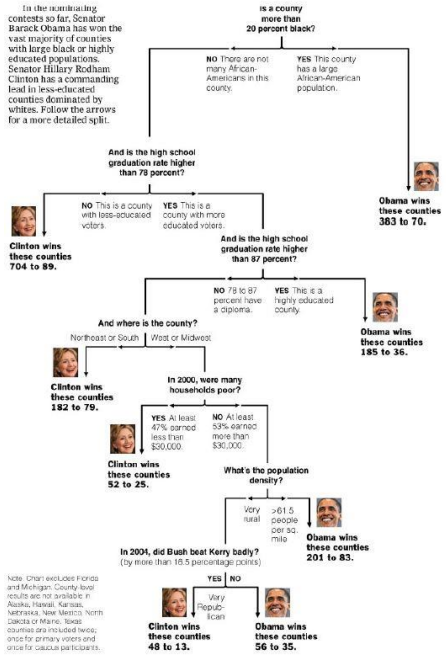


Classification and Regression Tree (CART)

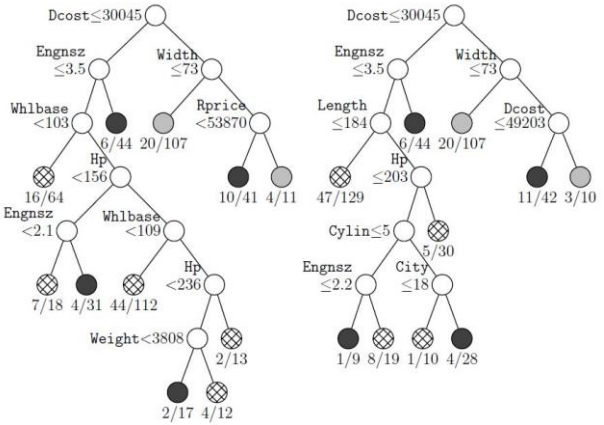
- 개별 변수의 영역을 반복적으로 분할함으로써 전체 영역에서의 규칙을 생성하는 지도학습 기법 (Breiman, 1984)
- If-then 형식으로 표현되는 규칙(rules)을 생성함으로써, 결과에 대한 예측과 함께 그 이유를 설명할 수 있는 장점이 있음
- 수치형 변수와 범주형 변수에 대한 동시 처리 가능

재귀적 분기
(Recursive Partitioning)

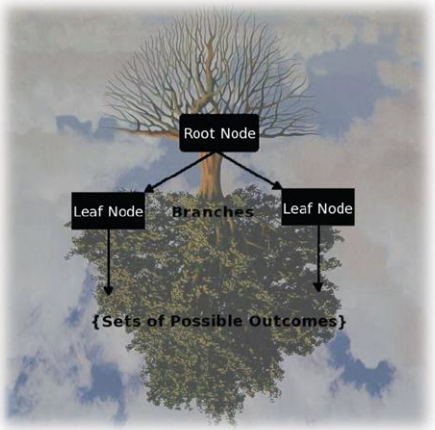
가지치기
(Pruning)



- 특정 영역(부모 노드)에 속하는 개체들을 하나의 기준 변수 값의 범위에 따라 분기
- 분기에 의해 새로 생성된 자식 노드의 동질성이 최대화 되도록 분기점 선택
- 불순도를 측정하는 기준으로는 범주형 변수에 대해서는 지니계수, 수치형 변수에 대해서는 분산을 이용

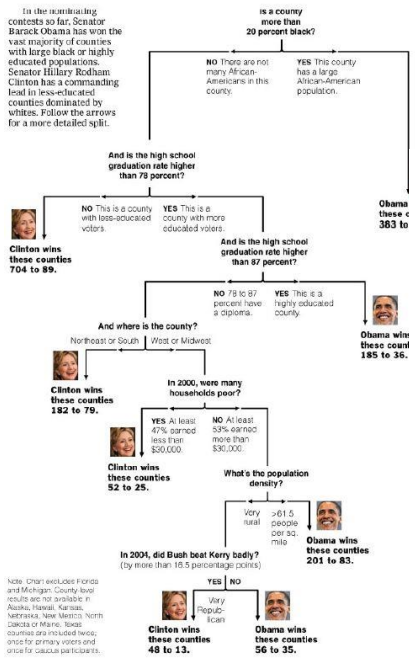


Classification and Regression Tree: CART

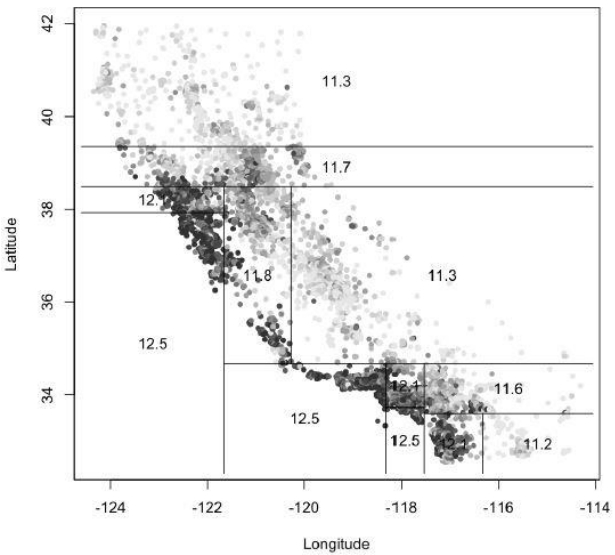


Classification and Regression Tree (CART)

- 개별 변수의 영역을 반복적으로 분할함으로써 전체 영역에서의 규칙을 생성하는 지도학습 기법 (Breiman, 1984)
- If-then 형식으로 표현되는 규칙(rules)을 생성함으로써, 결과에 대한 예측과 함께 그 이유를 설명할 수 있는 장점이 있음
- 수치형 변수와 범주형 변수에 대한 동시 처리 가능



재귀적 분기
(Recursive Partitioning)



가지치기
(Pruning)

- 과적합(Over-fitting)을 방지하기 위하여 하위 노드들을 상위 노드로 결합
- Pre-pruning: Tree를 생성하는 과정에서 최소 분기 기준을 이용하는 사전적 가지치기
- Post-pruning: Full-tree 생성 후, 검증 데이터의 오분류율과 Tree의 복잡도(말단 노드의 수) 등을 고려하는 사후적 가지치기

CART: 재귀적 분기

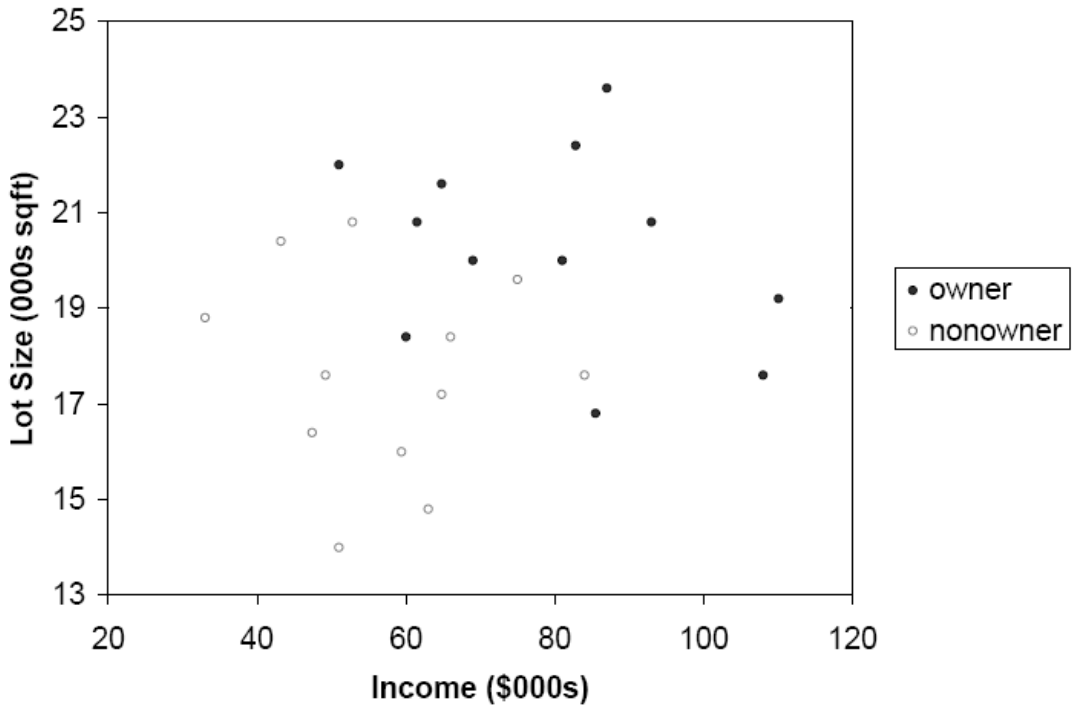
- 예시: 잔디깎기 기계 구입 예측
  - ✓ 목적: 24개의 가정에 대해 잔디깎기 기계 구입 여부를 분류
  - ✓ 설명변수: 수입(Income), 집의 크기(Lot size)

Income	Lot size	Ownership	Income	Lot size	Ownership
60.0	18.4	Owner	75.0	19.6	Non-owner
85.5	16.8	Owner	52.8	20.8	Non-owner
64.8	21.6	Owner	64.8	17.2	Non-owner
61.5	20.8	Owner	43.2	20.4	Non-owner
87.0	23.6	Owner	84.0	17.6	Non-owner
110.1	19.2	Owner	49.2	17.6	Non-owner
108.0	17.6	Owner	59.4	16.0	Non-owner
82.8	22.4	Owner	66.0	18.4	Non-owner
69.0	20.0	Owner	47.4	16.4	Non-owner
93.0	20.8	Owner	33.0	18.8	Non-owner
51.0	22.0	Owner	51.0	14.0	Non-owner
81.0	20.0	Owner	63.0	14.8	Non-owner

CART: 재귀적 분기

한 변수를 기준으로 하여 정렬

- 정렬 기준 변수: lot size



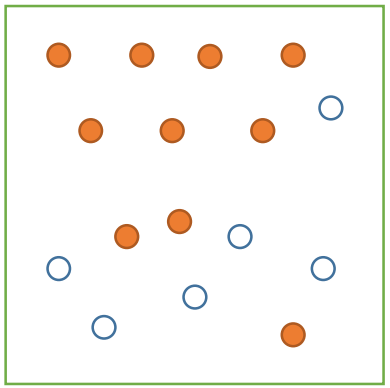
Income	Lot size	Ownership
51.0	14.0	Non-owner
63.0	14.8	Non-owner
59.4	16.0	Non-owner
47.4	16.4	Non-owner
85.5	16.8	Owner
64.8	17.2	Non-owner
108.0	17.6	Owner
84.0	17.6	Non-owner
49.2	17.6	Non-owner
60.0	18.4	Owner
66.0	18.4	Non-owner
33.0	18.8	Non-owner
110.1	19.2	Owner
75.0	19.6	Non-owner
69.0	20.0	Owner
81.0	20.0	Owner
43.2	20.4	Non-owner
61.5	20.8	Owner
93.0	20.8	Owner
52.8	20.8	Non-owner
64.8	21.6	Owner
51.0	22.0	Owner
82.8	22.4	Owner
87.0	23.6	Owner

불순도 지표: 지니 계수(Gini Index)

- m개의 레코드가 속하는 A 영역에 대한 지니 계수

$$I(A) = 1 - \sum_{k=1}^{m} p_k^2$$

✓ p_k = A영역에 속한 레코드 중 k 범주에 속하는 레코드의 수



$$I(A) = 1 - \sum_{k=1}^{m} p_k^2$$

$$= 1 - \left(\frac{6}{16}\right)^2 - \left(\frac{10}{16}\right)^2$$

$$\approx 0.47$$

✓ 모든 레코드가 동일한 범주에 속할 경우 I(A) = 0

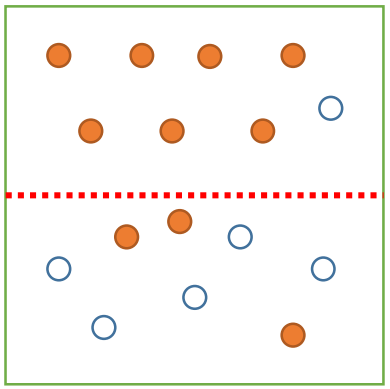
✓ 두 개의 범주에 속하는 개체의 수가 동일할 경우 I(A)=0.5

불순도 지표: 지니 계수(Gini Index)

- 두 개 이상의 영역에 대한 지니 계수

$$I(A) = \sum_{i=1}^{d} \left(R_i \left(1 - \sum_{k=1}^{m} p_{ik}^2 \right) \right)$$

✓ R_i = 분할 전 레코드 중 분할 후 i영역에 속하는 레코드의 비율



$$I(A)$$

$$= 0.5 \times \left(1 - \left(\frac{7}{8} \right)^2 - \left(\frac{1}{8} \right)^2 \right) + 0.5 \times \left(1 - \left(\frac{3}{8} \right)^2 - \left(\frac{5}{8} \right)^2 \right)$$

$$= 0.34$$

✓분기 후의 정보 획득(Information gain): 0.47-0.34=0.13

CART: 재귀적 분기

2

순차적으로 가능한 분기점에 대한 정보 획득 계산

- 첫 번째 후보 분기점 = 14.4 (0.5*(14.0+14.8))
- Lot size > 14.4 와 Lot size < 14.4 인 두 영역으로 구분
- 불순도(impurity) 계산: 지니 계수(Gini index)
  - ✓ 분기 전:

$$1-\left(\frac{12}{24}\right)^2-\left(\frac{12}{24}\right)^2=0.5$$

  - ✓ 분기 후:

$$\frac{1}{24}\left(1-\left(\frac{1}{1}\right)^2\right)+\frac{23}{24}\left(1-\left(\frac{12}{23}\right)^2-\left(\frac{11}{23}\right)^2\right)\approx 0.48$$

  - ✓ 정보 획득(Information gain): 0.50-0.48=0.02

Income	Lot size	Ownership
51.0	14.0	Non-owner
63.0	14.8	Non-owner
59.4	16.0	Non-owner
47.4	16.4	Non-owner
85.5	16.8	Owner
64.8	17.2	Non-owner
108.0	17.6	Owner
84.0	17.6	Non-owner
49.2	17.6	Non-owner
60.0	18.4	Owner
66.0	18.4	Non-owner
33.0	18.8	Non-owner
110.1	19.2	Owner
75.0	19.6	Non-owner
69.0	20.0	Owner
81.0	20.0	Owner
43.2	20.4	Non-owner
61.5	20.8	Owner
93.0	20.8	Owner
52.8	20.8	Non-owner
64.8	21.6	Owner
51.0	22.0	Owner
82.8	22.4	Owner
87.0	23.6	Owner

CART: 재귀적 분기

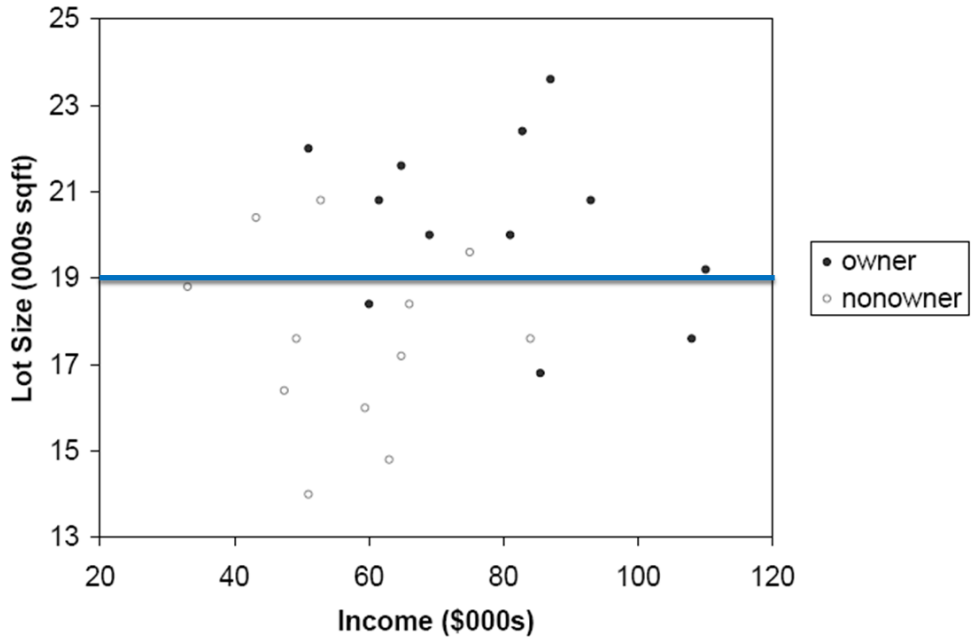
- 범주형 변수에 대해서는 분기가 가능한 모든 경우의 수를 조사
  - ✓ 예시: 3개의 범주 A, B, and C가 존재하는 경우
    - {A} vs. {B,C}
    - {B} vs. {A,C}
    - {C} vs. {A,B}
  - ✓ 위 경우의 수에 대한 정보 획득을 각각 계산

CART: 재귀적 분기

최적의 분기점 선택

- 정보획득을 최대화하는 분기점을 선택하여 분기

3

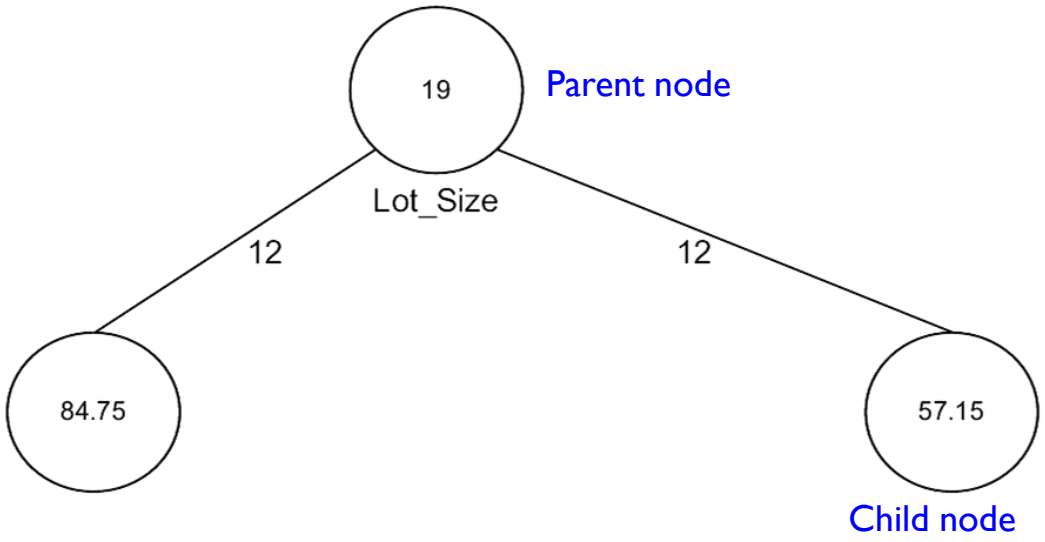


Income	Lot size	Ownership
51.0	14.0	Non-owner
63.0	14.8	Non-owner
59.4	16.0	Non-owner
47.4	16.4	Non-owner
85.5	16.8	Owner
64.8	17.2	Non-owner
108.0	17.6	Owner
84.0	17.6	Non-owner
49.2	17.6	Non-owner
60.0	18.4	Owner
66.0	18.4	Non-owner
33.0	18.8	Non-owner
110.1	19.2	Owner
75.0	19.6	Non-owner
69.0	20.0	Owner
81.0	20.0	Owner
43.2	20.4	Non-owner
61.5	20.8	Owner
93.0	20.8	Owner
52.8	20.8	Non-owner
64.8	21.6	Owner
51.0	22.0	Owner
82.8	22.4	Owner
87.0	23.6	Owner

CART: 재귀적 분기

3

의사결정나무 구조



- 분기점은 나무의 노드가 되며 원 안의 숫자는 분기 기준이 되는 변수의 값임
- 사각형은 말단 노드(leaf node, 더 이상 분기가 수행되지 않는 노드)가 됨
- 선 아래 숫자는 분기로 인해 나뉜 레코드의 수를 의미

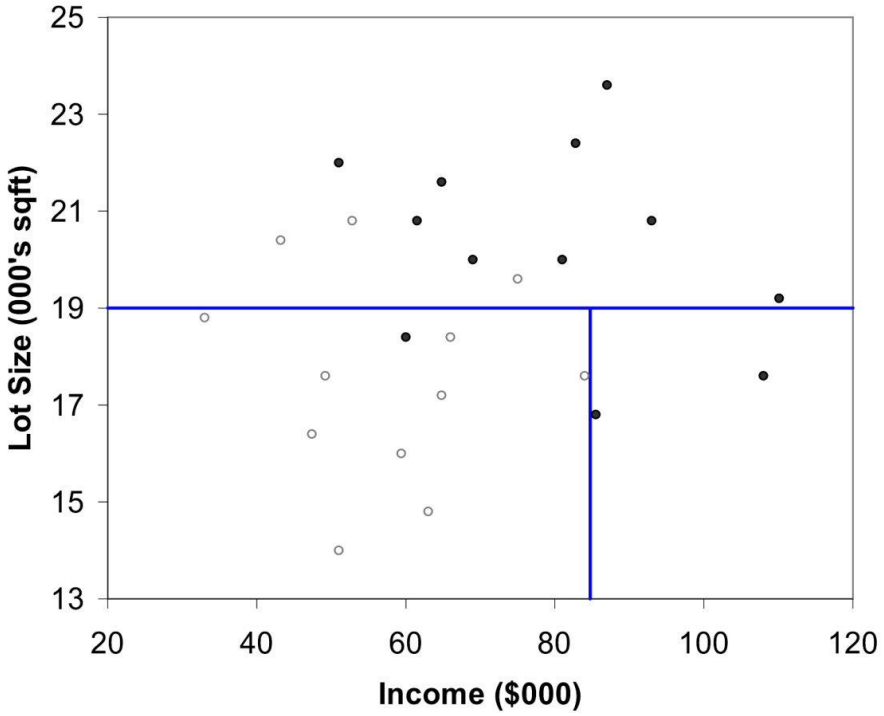
Income	Lot size	Ownership
51.0	14.0	Non-owner
63.0	14.8	Non-owner
59.4	16.0	Non-owner
47.4	16.4	Non-owner
85.5	16.8	Owner
64.8	17.2	Non-owner
108.0	17.6	Owner
84.0	17.6	Non-owner
49.2	17.6	Non-owner
60.0	18.4	Owner
66.0	18.4	Non-owner
33.0	18.8	Non-owner
110.1	19.2	Owner
75.0	19.6	Non-owner
69.0	20.0	Owner
81.0	20.0	Owner
43.2	20.4	Non-owner
61.5	20.8	Owner
93.0	20.8	Owner
52.8	20.8	Non-owner
64.8	21.6	Owner
51.0	22.0	Owner
82.8	22.4	Owner
87.0	23.6	Owner

CART: 재귀적 분기

4

모든 노드의 순도가 100%가 될 때까지 반복적으로 분기를 수행

- 정보획득이 0이 되는 시점까지 수행
- 두 번째 분기점: income = 84.75



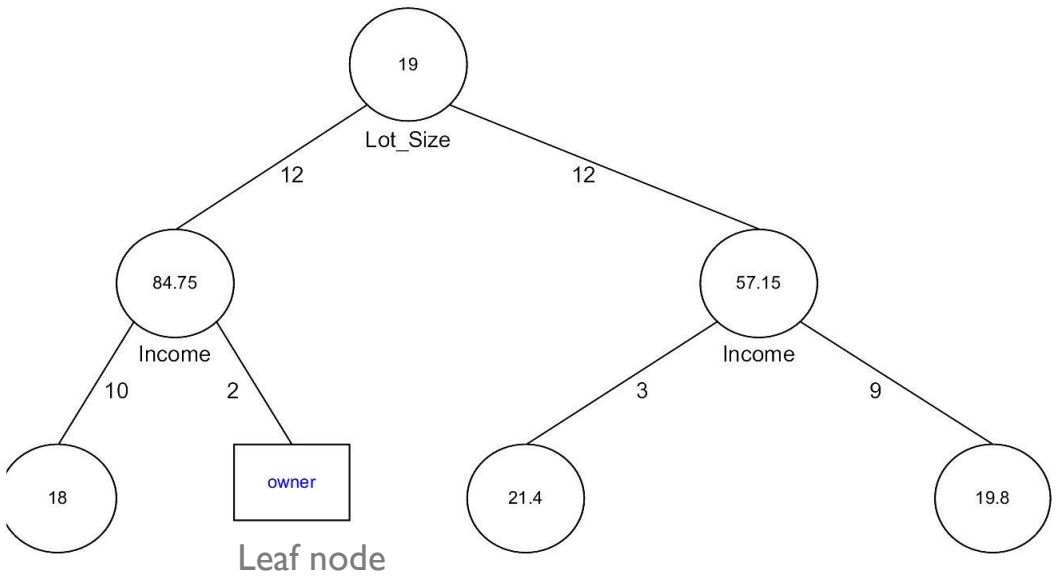
Income	Lot size	Ownership
33.0	18.8	Non-owner
47.4	16.4	Non-owner
49.2	17.6	Non-owner
51.0	14.0	Non-owner
59.4	16.0	Non-owner
60.0	18.4	Owner
63.0	14.8	Non-owner
64.8	17.2	Non-owner
66.0	18.4	Non-owner
84.0	17.6	Non-owner
85.5	16.8	Owner
108.0	17.6	Owner

CART: 재귀적 분기

4

모든 노드의 순도가 100%가 될 때까지 반복적으로 분기를 수행

- 정보획득이 0이 되는 시점까지 수행
- 두 번째 분기점: income = 84.75

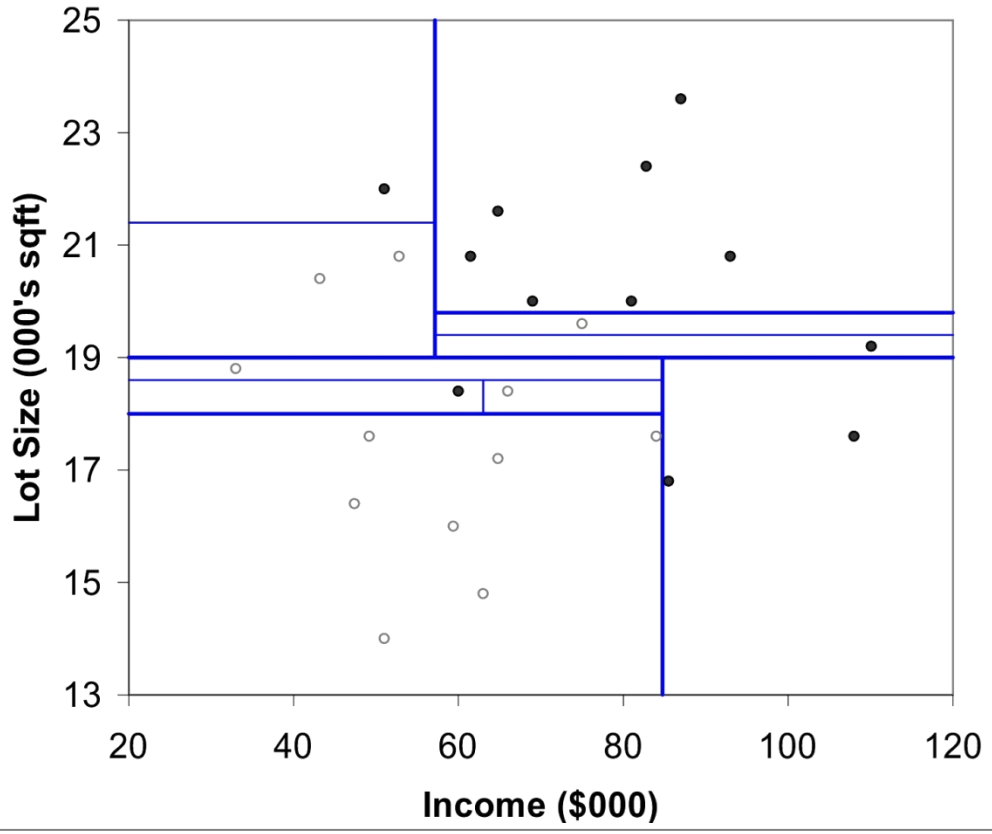


Income	Lot size	Ownership
33.0	18.8	Non-owner
47.4	16.4	Non-owner
49.2	17.6	Non-owner
51.0	14.0	Non-owner
59.4	16.0	Non-owner
60.0	18.4	Owner
63.0	14.8	Non-owner
64.8	17.2	Non-owner
66.0	18.4	Non-owner
84.0	17.6	Non-owner
85.5	16.8	Owner
108.0	17.6	Owner

CART: 재귀적 분기

재귀적 분기 완료

- 모든 영역에는 하나의 범주에 속하는 레코드만 존재



4

CART: 예측

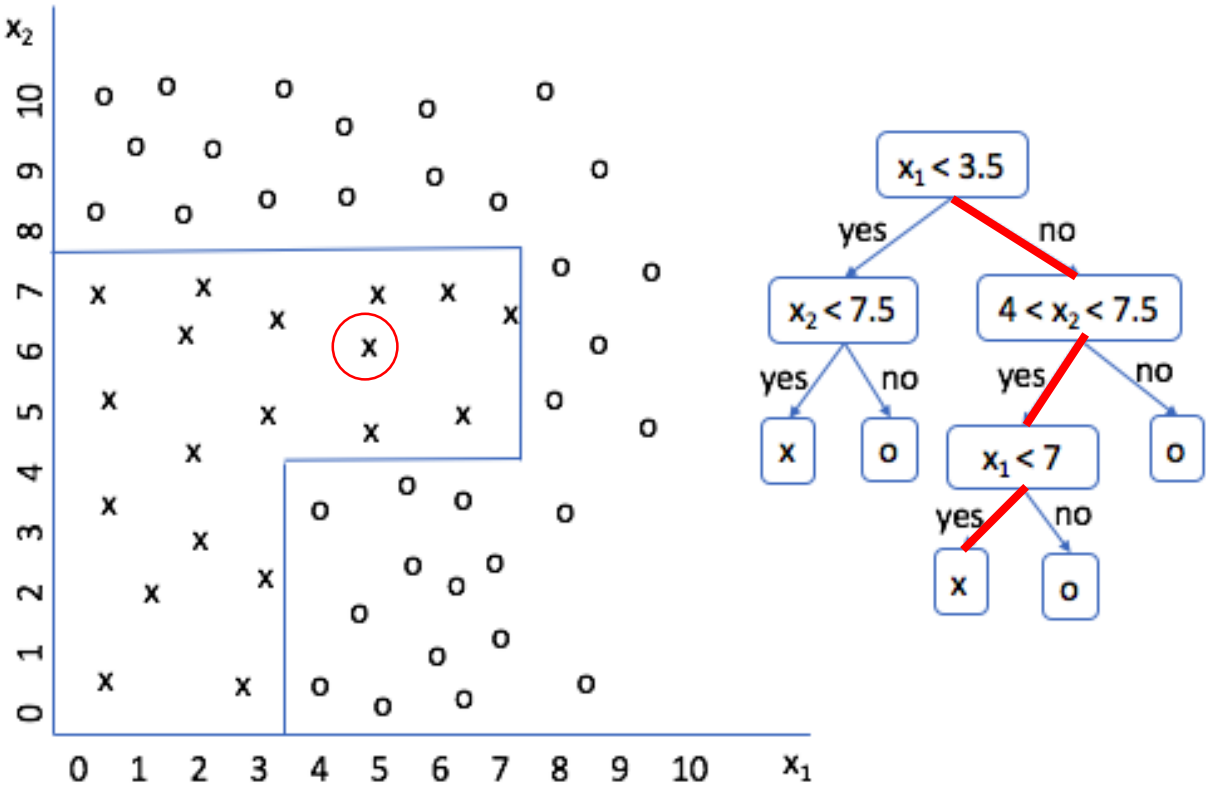
의사결정나무를 통한 예측

- 의사결정나무가 생성된 뒤 각 말단노드에 속하는 레코드들은 학습데이터에 속한 레코드들임
- 각 말단노드에 해당하는 범주는 해당 노드에 포함된 개체들이 속한 범주의 비율을 이용하여 판단
- 분류 기준값(Cut-off)을 0.5로 사용하게 되면 다수결(majority voting) 방식을 이용하여 범주를 예측하는 것임
- 범주 A에 대한 Cut-off를 0.75로 설정 = 각 말단노드에 속하는 레코드의 비율이 0.75 이상일 경우에만 범주 A로 예측

4

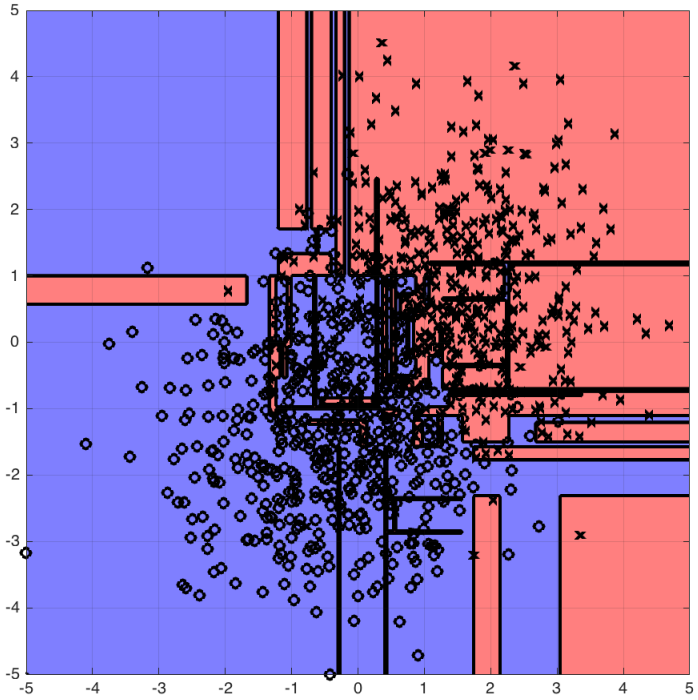
CART: 예측

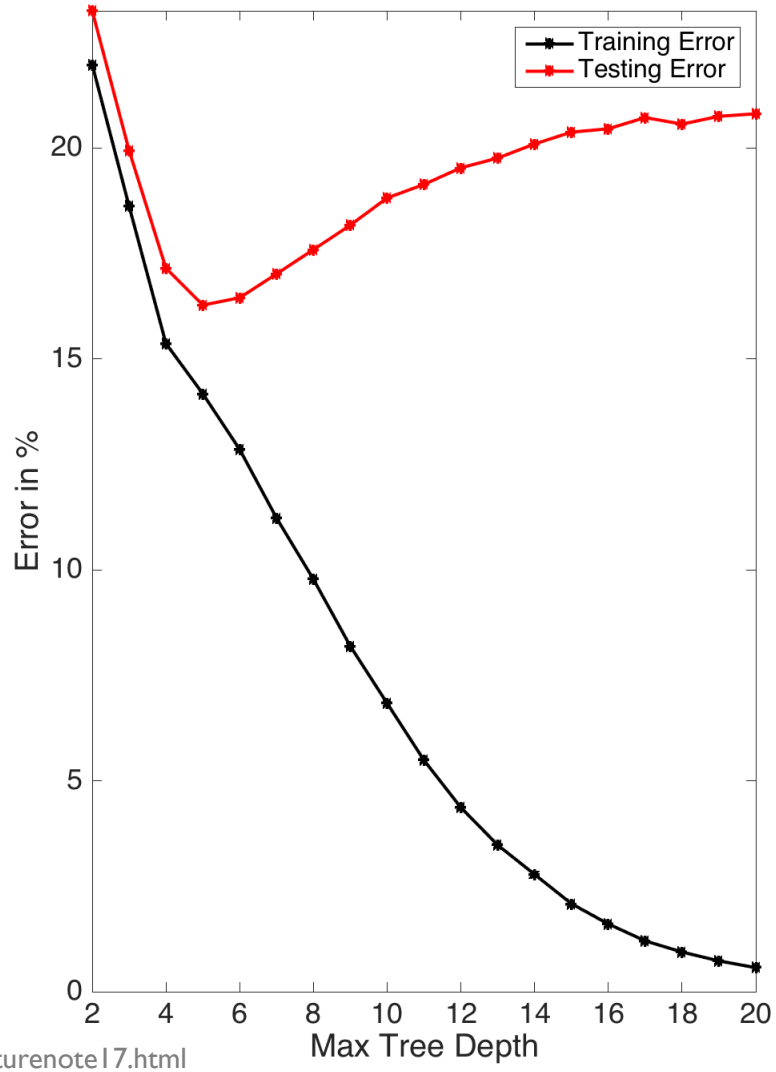
- 의사결정나무를 통한 분류 예측 예시



의사결정나무: 과적합 문제

- 재귀적 분기는 모든 말단노드의 순도가 100%일때 종료됨

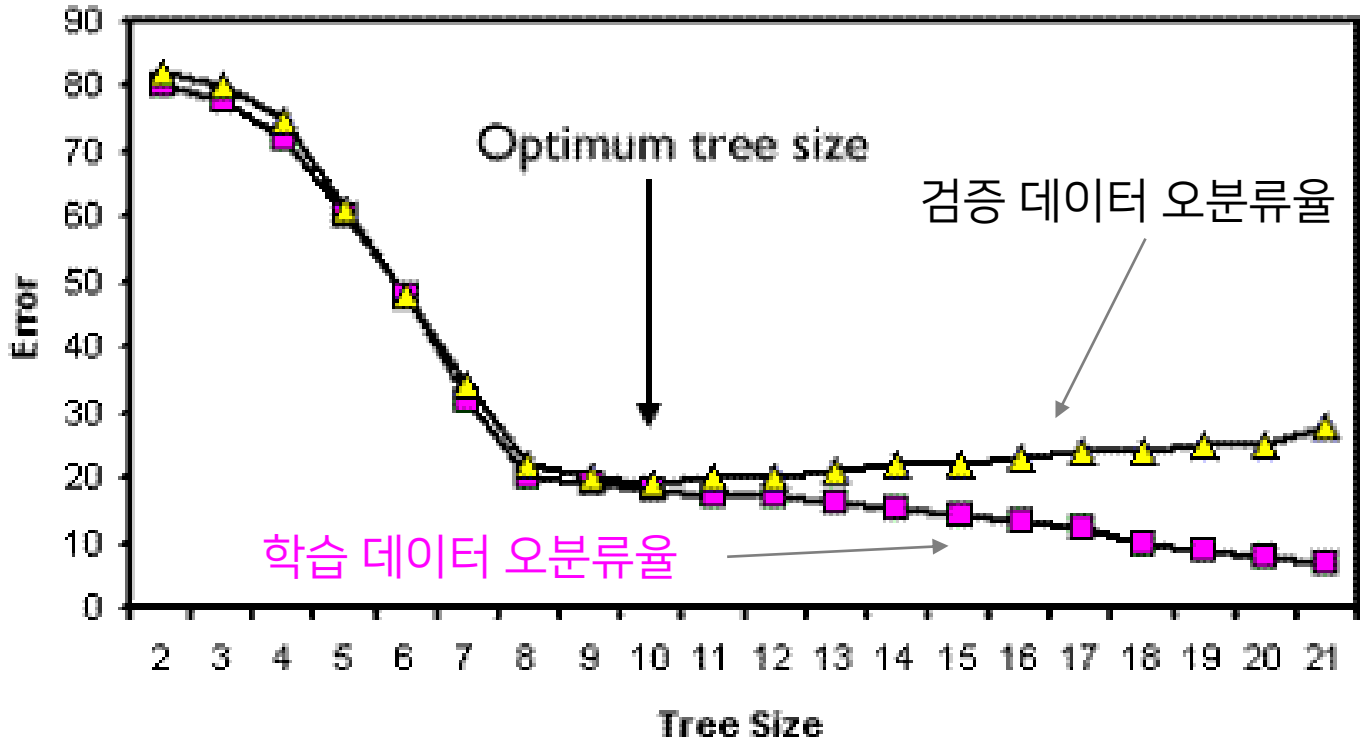




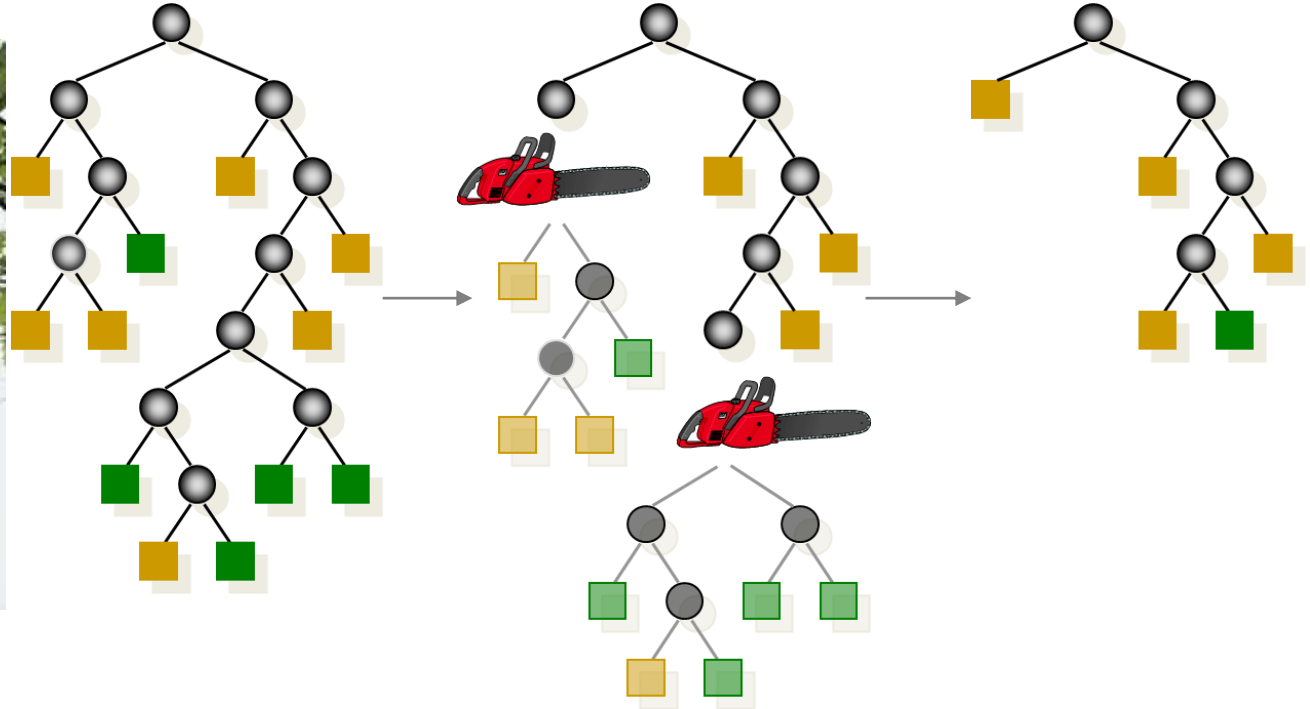
http://www.cs.cornell.edu/courses/cs4780/2017sp/lectures/lecturenote17.html

의사결정나무: 과적합 문제

- 재귀적 분기는 모든 말단노드의 순도가 100%일때 종료됨
  - ✓ 일반적으로 이러한 Full tree는 **과적합의 문제**를 내포하고 있으며, 이는 **새로운 데이터에 대한 예측 성능 저하의 위험**을 안고 있음
  - ✓ 의사결정나무의 노드 수가 증가할 때, 처음에는 새로운 데이터에 대한 오분류율이 감소하나, 일정 수준 이상이 되면 오분류율이 증가하는 현상 발생



의사결정나무: 가지치기 (Pruning)



- ✓ 의사결정나무는 Full Tree를 생성한 뒤 적절한 수준에서 말단 노드를 결합하는 가지치기 수행
- ✓ 검증데이터에 대한 오분류율이 증가하는 시점에서 가지치기
- ✓ Full tree에 비해 구조가 단순한 의사결정나무가 생성됨
- ✓ 비용 복잡도(cost complexity) 를 사용하여 최적의 의사결정나무 구조 선택

의사결정나무: 가지치기 (Pruning)

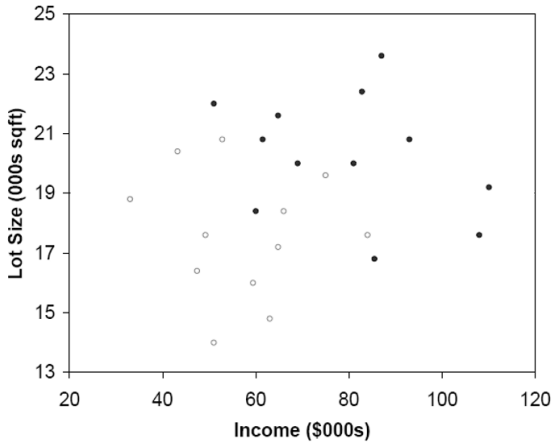
- 비용 복잡도 (Cost complexity)

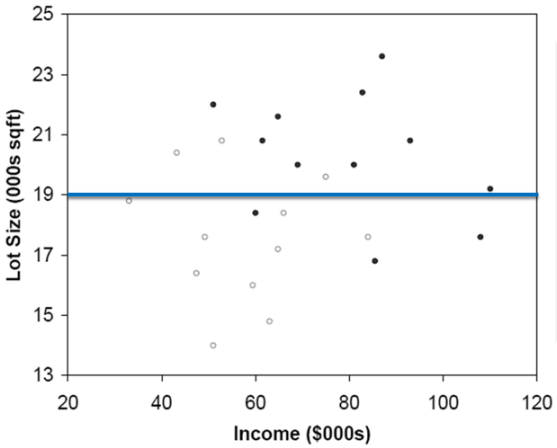
$$CC(T) = Err(T) + \alpha \times L(T)$$

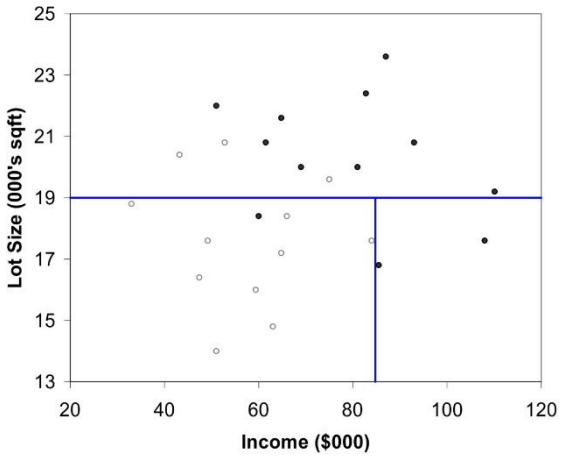
- ✓ CC(T) = 의사결정나무의 비용 복잡도 (낮을수록 우수한 의사결정나무)
- ✓ ERR(T) = 검증데이터에 대한 오분류율
- ✓ L(T) = 말단 노드의 수 (구조의 복잡도)
- ✓ Alpha = ERR(T)와 L(T)를 결합하는 가중치 (사용자에 의해 부여됨)

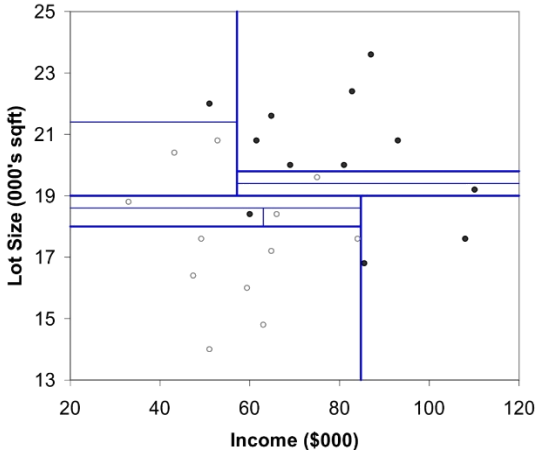
의사결정나무: 가지치기 (Pruning)

- 재귀적 분기를 통한 Full Tree 생성





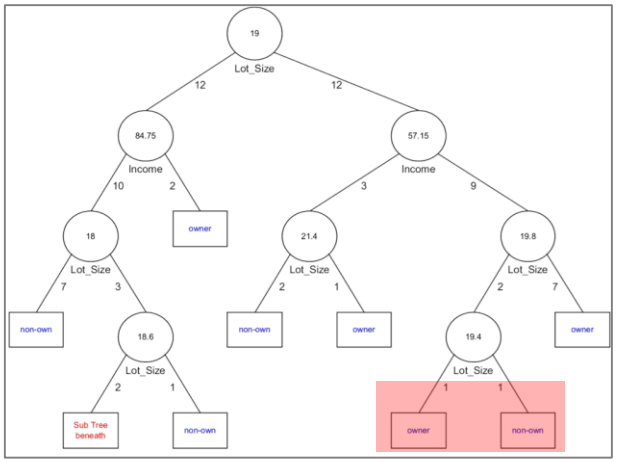




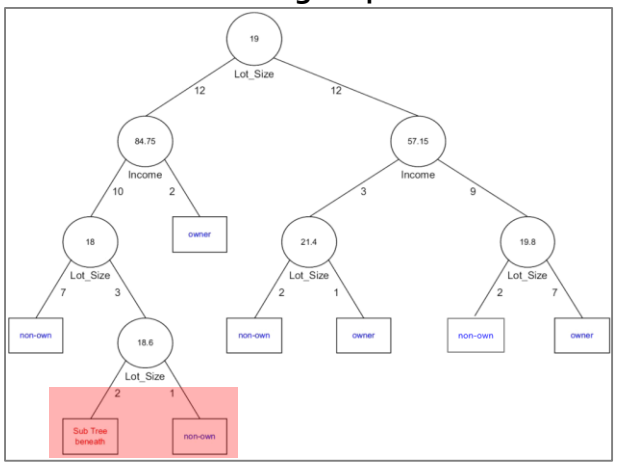
의사결정나무: 가지치기 (Pruning)

- 가지치기를 통한 일반화 성능 확보

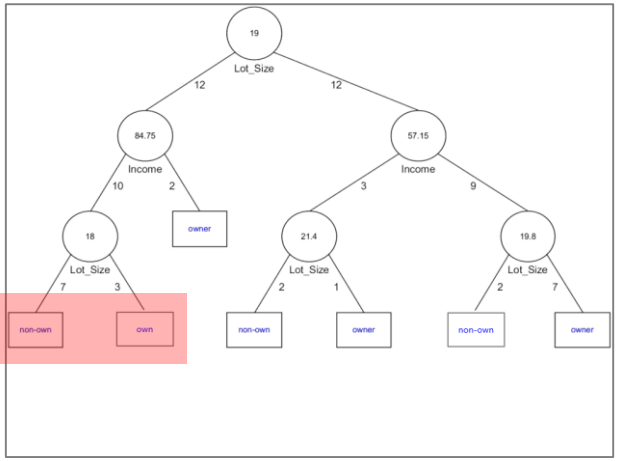
Full Tree

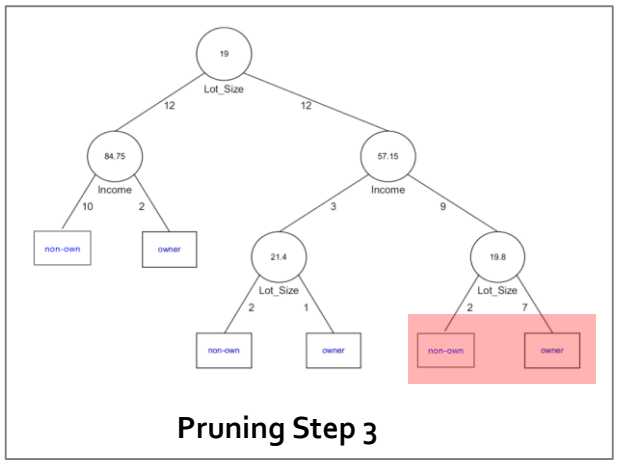


Pruning Step 1

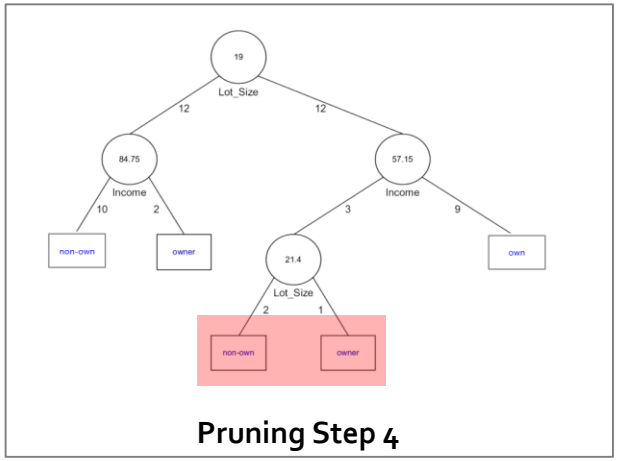


Pruning Step 2

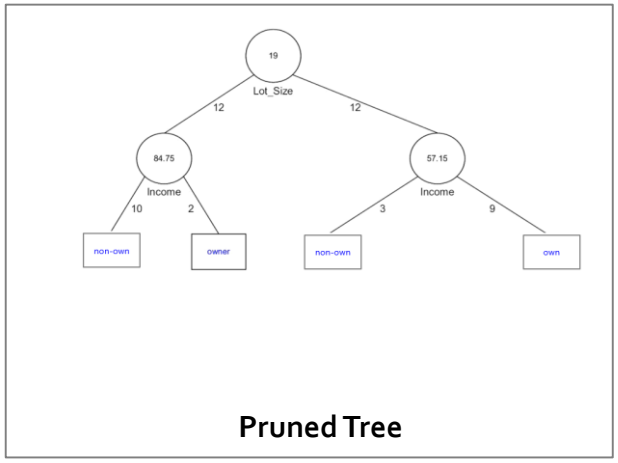




Pruning Step 3



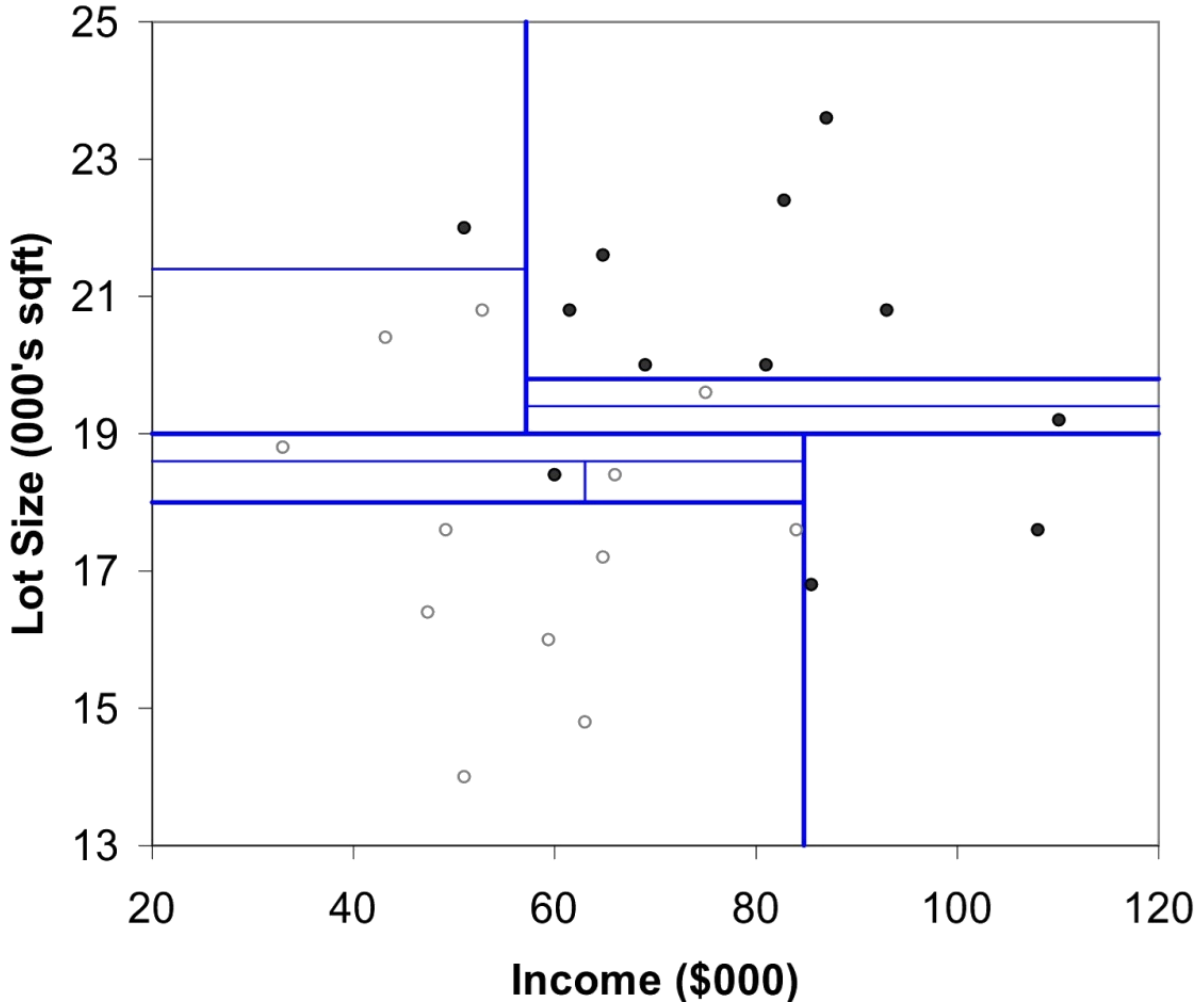
Pruning Step 4



Pruned Tree

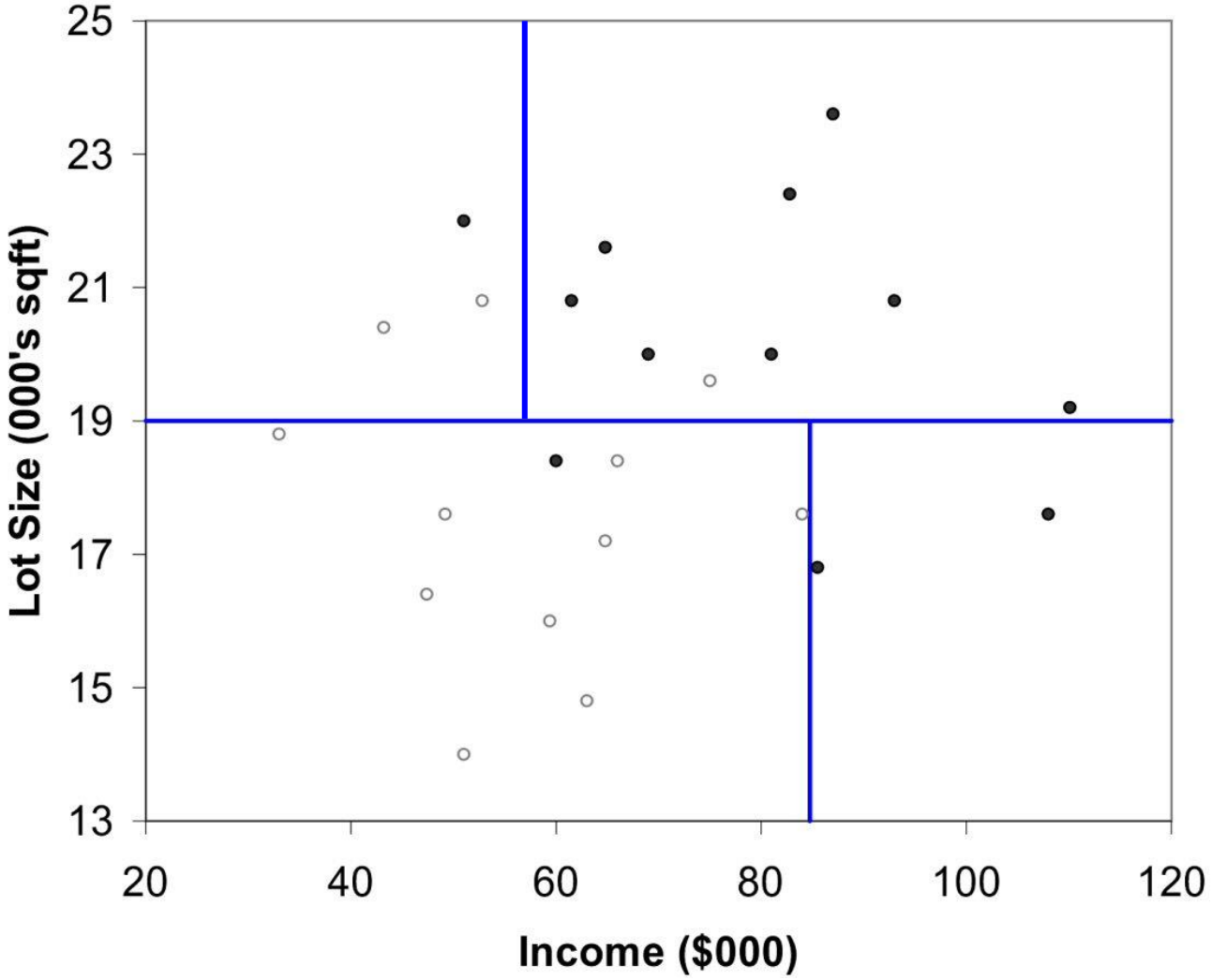
의사결정나무: 가지치기 (Pruning)

- 가지치기 수행 전 (재귀적 분기 완료 단계)



의사결정나무: 가지치기 (Pruning)

- 가지치기 수행 후



의사결정나무 예시

- Universal bank의 개인신용대출 예측
  - ✓ 고객의 인구통계학적 정보와 은행 이용 행태를 바탕으로 개인신용대출을 이용할 고객 판별

일련 번호	나이	경력	소득	가족 수	월별 신용카드 평균사용액	교육 수준	담보부 채권	개인 대출	증권 계좌	CD 계좌	온라인 뱅킹	신용 카드
1	25	1	49	4	1.60	UG	0	No	Yes	No	No	No
2	45	19	34	3	1.50	UG	0	No	Yes	No	No	No
3	39	15	11	1	1.00	UG	0	No	No	No	No	No
4	35	9	100	1	2.70	Grad	0	No	No	No	No	No
5	35	8	45	4	1.00	Grad	0	No	No	No	No	Yes
6	37	13	29	4	0.40	Grad	155	No	No	No	Yes	No
7	53	27	72	2	1.50	Grad	0	No	No	No	Yes	No
8	50	24	22	1	0.30	Prof	0	No	No	No	No	Yes
9	35	10	81	3	0.60	Grad	104	No	No	No	Yes	No
10	34	9	180	1	8.90	Prof	0	Yes	No	No	No	No
11	65	39	105	4	2.40	Prof	0	No	No	No	No	No
12	29	5	45	3	0.10	Grad	0	No	No	No	Yes	No
13	48	23	114	2	3.80	Prof	0	No	Yes	No	No	No
14	59	32	40	4	2.50	Grad	0	No	No	No	Yes	No
15	67	41	112	1	2.00	UG	0	No	Yes	No	No	No
16	60	30	22	1	1.50	Prof	0	No	No	No	Yes	Yes
17	38	14	130	4	4.70	Prof	134	Yes	No	No	No	No
18	42	18	81	4	2.40	UG	0	No	No	No	No	No
19	46	21	193	2	8.10	Prof	0	Yes	No	No	No	No
20	55	28	21	1	0.50	Grad	0	No	Yes	No	No	Yes

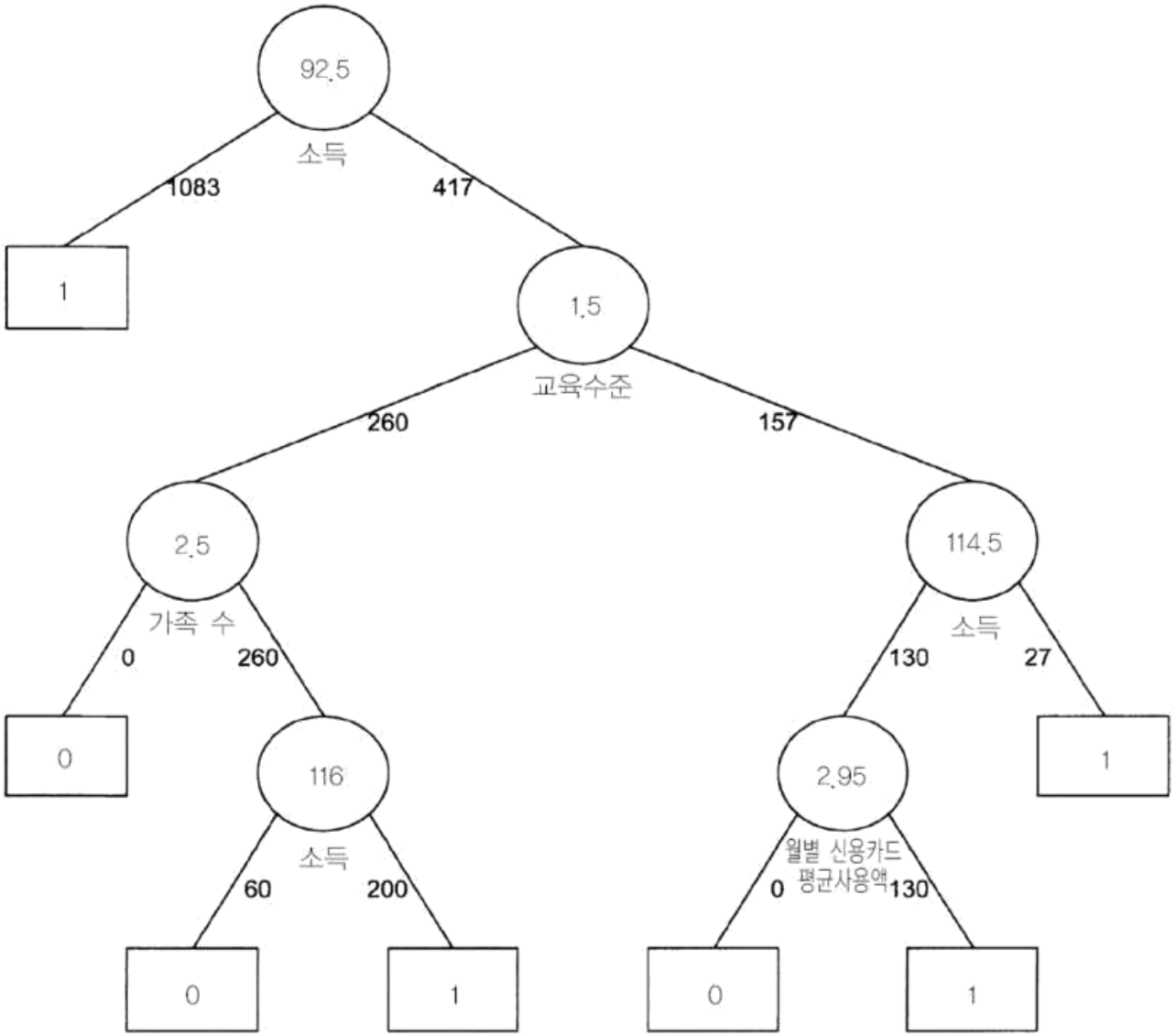
의사결정나무 예시

의사결정 마디	학습용 집합의 오차율	평가용 집합의 오차율
41	0	2.133333
40	0.04	2.2
39	0.08	2.2
38	0.12	2.2
37	0.16	2.066667
36	0.2	2.066667
35	0.2	2.066667
34	0.24	2.066667
...	...	...
13	1.16	1.6
12	1.2	1.6
11	1.2	1.466667
10	1.6	1.666667
9	2.2	1.666667
8	2.2	1.866667
7	2.24	1.866667
6	2.24	1.6
5	4.44	1.8
4	5.08	2.333333
3	5.24	3.466667
2	9.4	9.533333
1	9.4	9.533333
0	9.4	9.533333

최소 오차 나무 | 표준오차 | 0.003103929

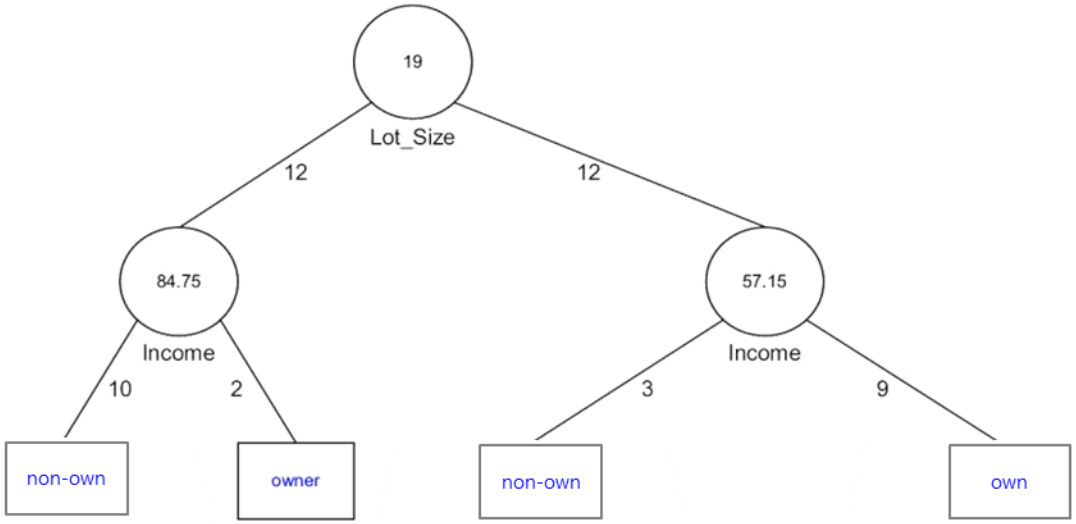
<-- 최적의 가지친 나무

의사결정나무 예시


92.5
소득
1083
417
1
1.5
교육수준
260
157
2.5
가족 수
0
260
114.5
소득
130
27
0
116
소득
60
200
2.95
월별 신용카드 평균사용액
0
130
1
0
1
0
1

의사결정나무 예시: 규칙 생성

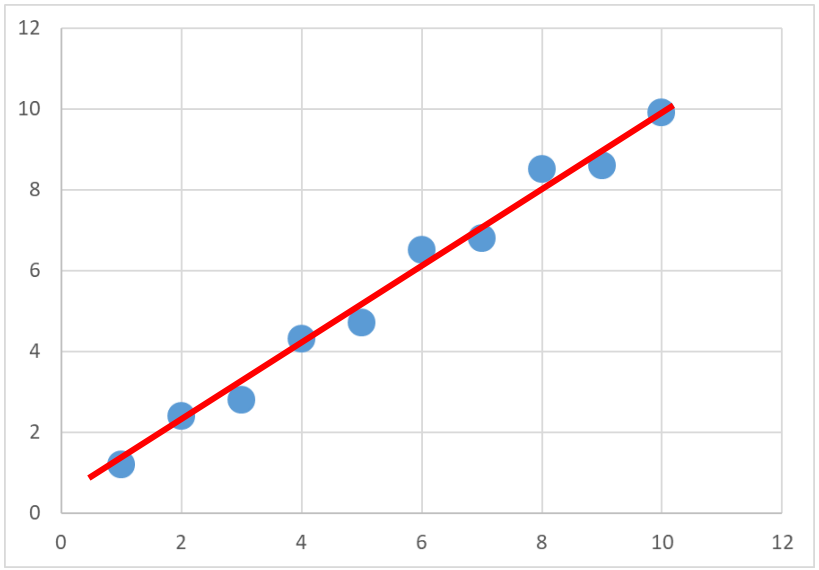
- 최적 의사결정으로부터 분류 규칙을 생성

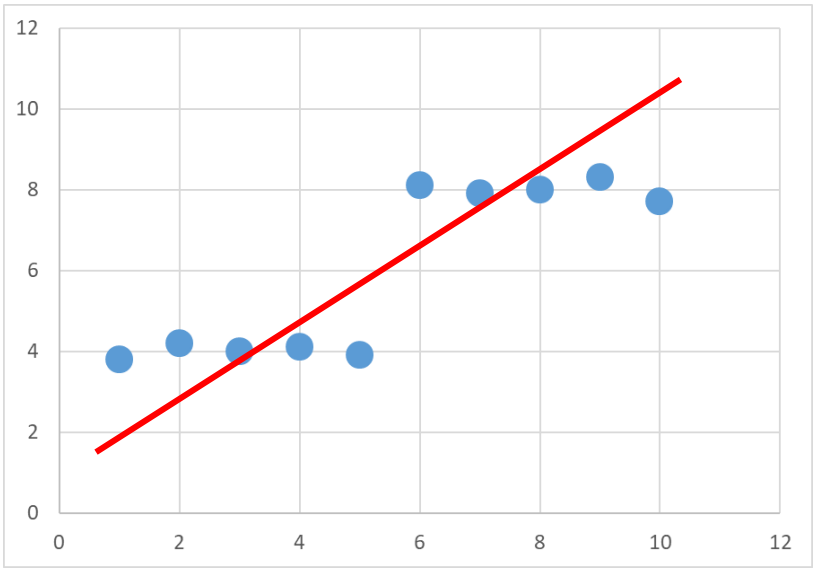


- IF(Lot size < 19) AND IF(Income < 84.75) THEN Owner = No
- IF(Lot size < 19) AND IF(Income > 84.75) THEN Owner = YES
- IF(Lot size > 19) AND IF(Income < 57.15) THEN Owner = NO
- IF(Lot size > 19) AND IF(Income > 57.15) THEN Owner = YES

의사결정나무: 회귀

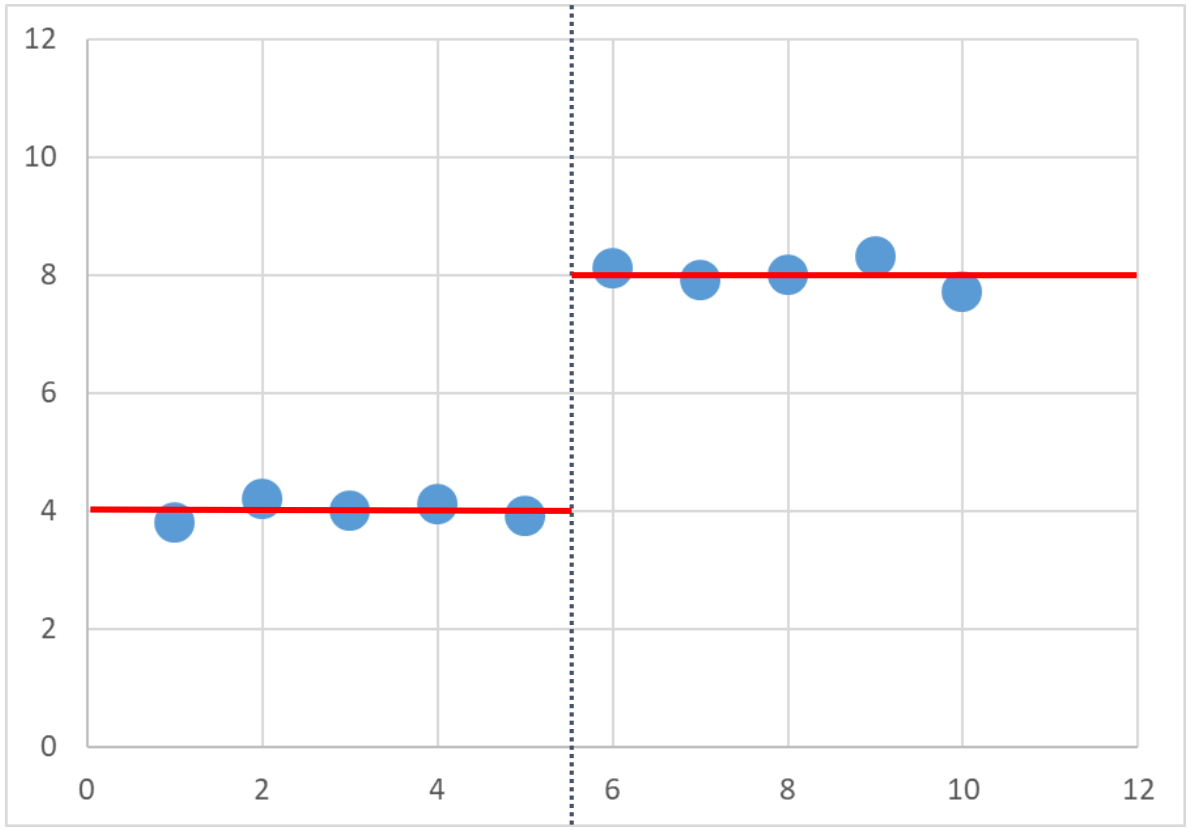
- 회귀 나무의 형태
  - ✓아래 그림에서 왼쪽과 같은 형태는 선형 회귀분석으로 추정하는 것이 적합함
  - ✓오른쪽 그림과 같은 형태는 선형 회귀분석으로 추정하는 것이 부적합함





의사결정나무: 회귀

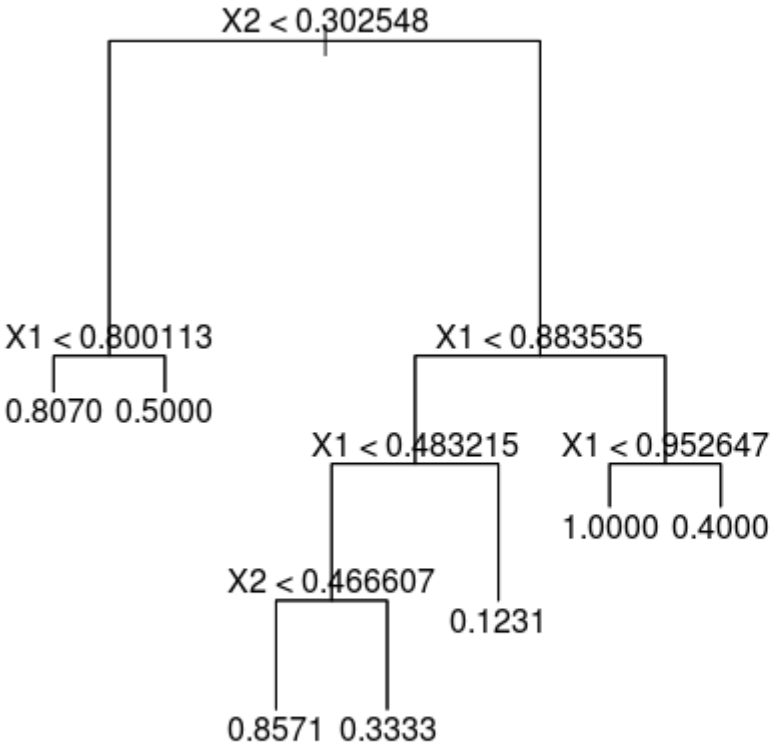
- 말단 노드의 예측값
  - ✓ 해당 노드에 속하는 모든 개체의 종속변수 값의 평균으로 예측
  - ✓ 아래 예시에서 split point = 5.5, if x < 5.5, y = 4, if x > 5.5, y = 8

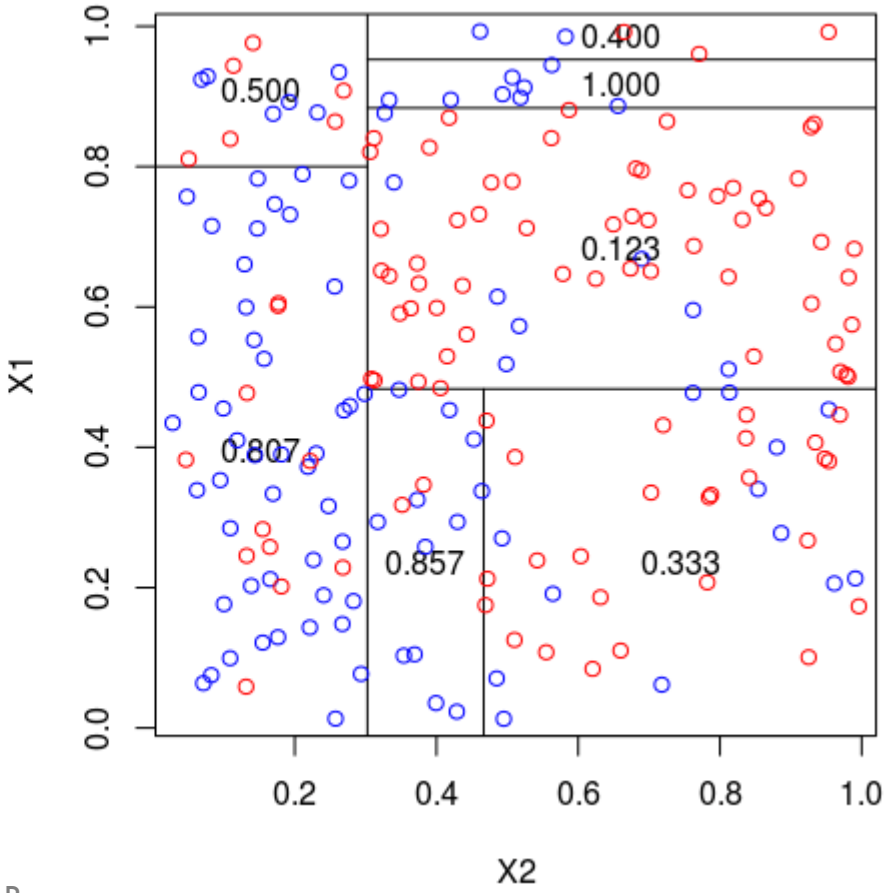


Split point

의사결정나무: 회귀

- 말단 노드의 예측값
  - ✓해당 노드에 속하는 모든 개체의 종속변수 값의 평균으로 예측

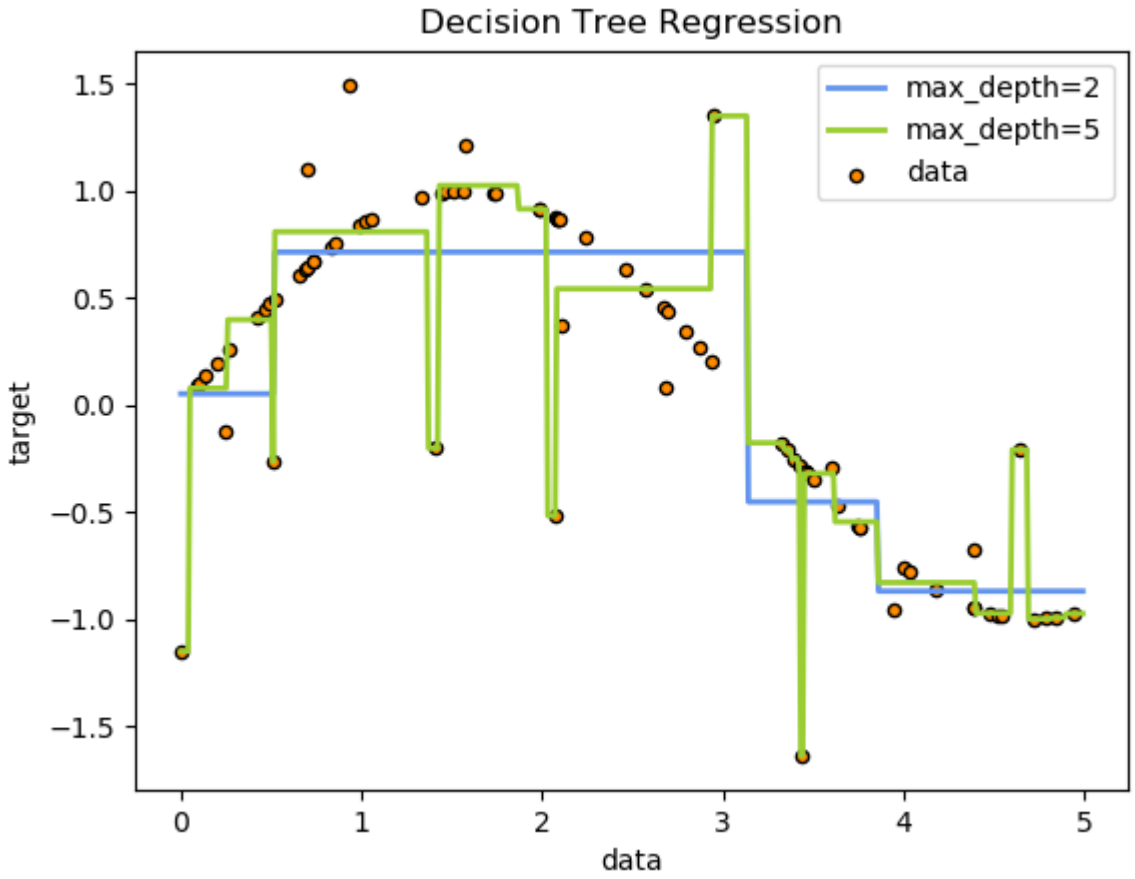




https://www.datacamp.com/community/tutorials/decision-trees-R

의사결정나무: 회귀

- 말단 노드의 예측값
  - ✓해당 노드에 속하는 모든 개체의 종속변수 값의 평균으로 예측

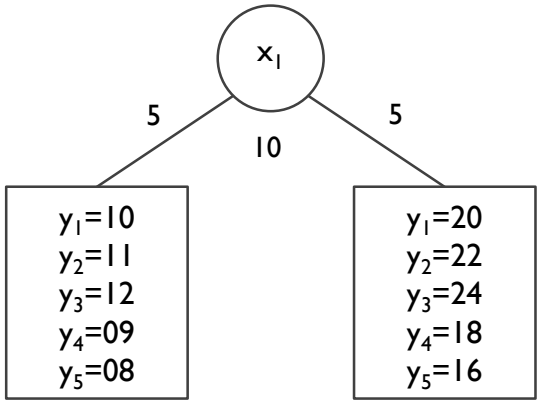


https://scikit-learn.org/stable/auto_examples/tree/plot_tree_regression.html#sphx-glr-auto-examples-tree-plot-tree-regression-py

의사결정나무: 회귀

회귀 모형에서 불순도를 측정하는 과정

- 불순도(Impurity) 측정
  - ✓ Sum of the squared error (SSE: $\sum_{i=1}^{n}(y_i - \hat{y})^2$)
  - ✓ SSE(Parent) = 300, SSE(Left) = 10, SSE(Right) = 40, Gain = 250



- 왼쪽 말단 노드의 예측값 = 10
- 오른쪽 말단 노드의 예측값 = 20

의사결정나무: 회귀

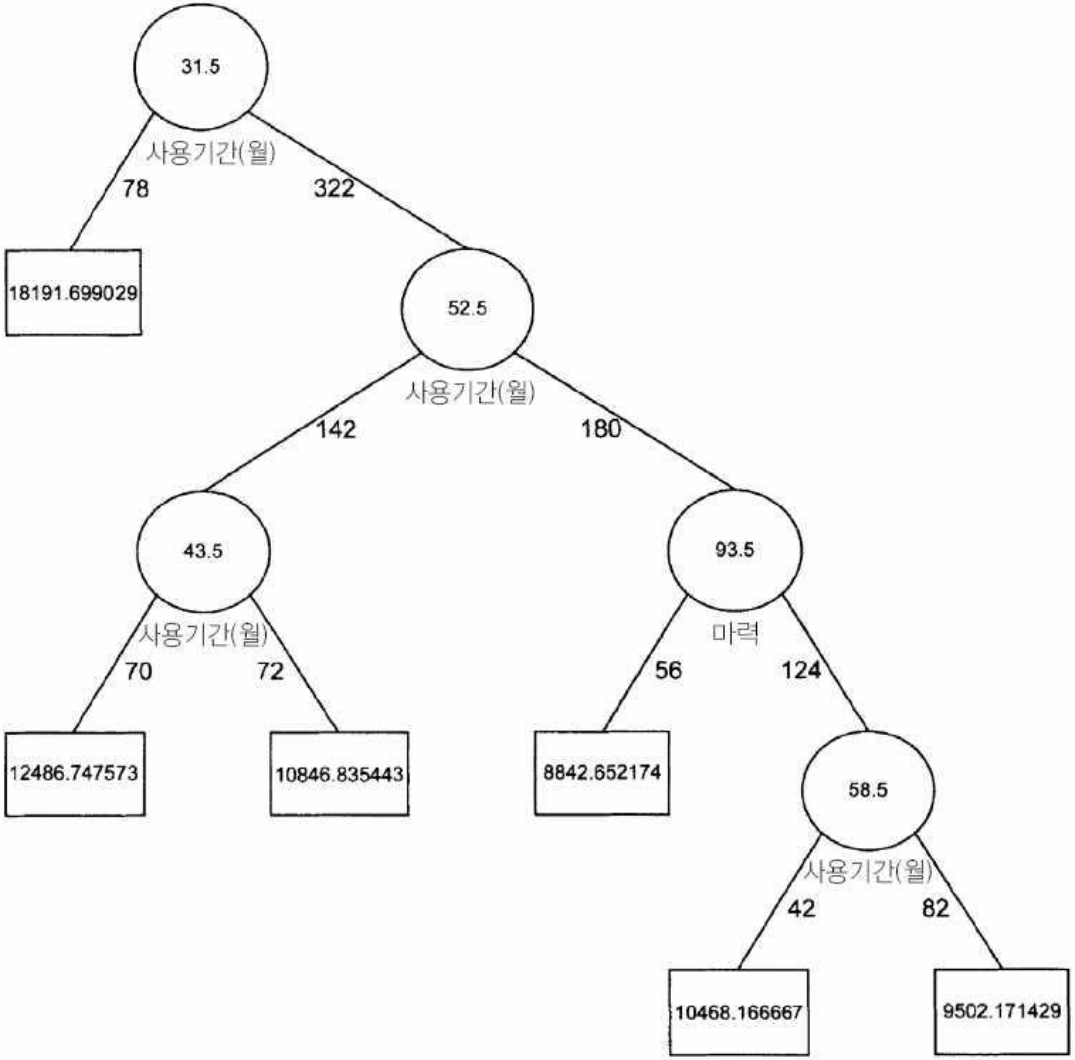
- 토요타 코롤라 중고차 가격 예측 문제

Dependent variable (target)

Independent variables (attributes, features)

Variable	Description
Price	Offer Price in EUROs
Age_08_04	Age in months as in August 2004
KM	Accumulated Kilometers on odometer
Fuel_Type	Fuel Type (Petrol, Diesel, CNG)
HP	Horse Power
Met_Color	Metallic Color? (Yes=1, No=0)
Automatic	Automatic ((Yes=1, No=0)
CC	Cylinder Volume in cubic centimeters
Doors	Number of doors
Quarterly_Tax	Quarterly road tax in EUROs
Weight	Weight in Kilograms

의사결정나무: 회귀

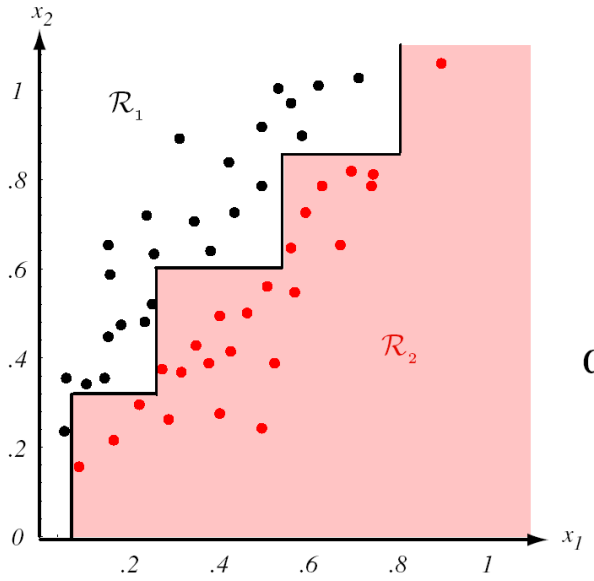

31.5
사용기간(월)
78
322
18191.699029
52.5
사용기간(월)
142
180
43.5
93.5
사용기간(월)
마력
70
72
56
124
12486.747573
10846.835443
8842.652174
58.5
사용기간(월)
42
82
10468.166667
9502.171429

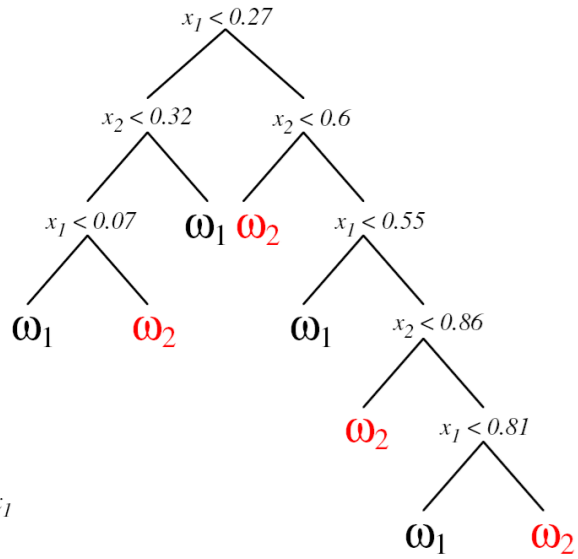
의사결정나무: 요약

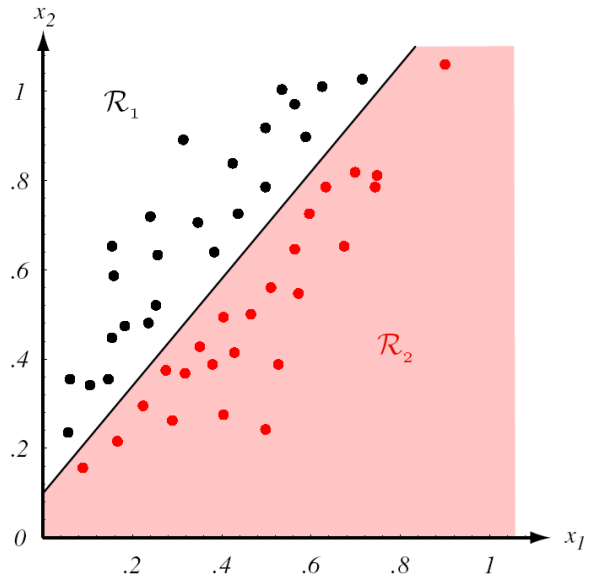
- **의사결정나무의 장점**
  - ✓ 의사결정나무는 예측에 대한 설명을 제공할 수 있음
  - ✓ 변수 선택 과정이 자동적으로 수행됨
  - ✓ 특별한 통계적 가정을 요구하지 않음
  - ✓ 결측치가 존재하는 상황에서도 모델 구축이 가능함

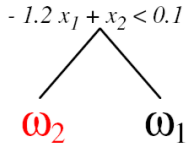
- **의사결정나무의 단점**
  - ✓ 축에 수직이나 수평인 분류 경계면만 이용하여 분류가 어려운 문제에서는 예측 성능이 저하될 수 있음
  - ✓ 한 번에 하나의 변수만 고려하므로 변수간 상호작용을 파악하기 어려움

의사결정나무: 요약


x_2
1
.8
.6
.4
.2
0
$\mathcal{R}_1$
$\mathcal{R}_2$
.2
.4
.6
.8
1
x_1


$x_1 < 0.27$
$x_2 < 0.32$
$x_2 < 0.6$
$x_1 < 0.07$
ω_1
ω_2
$x_1 < 0.55$
ω_1
ω_2
ω_1
$x_2 < 0.86$
ω_2
$x_1 < 0.81$
ω_1
ω_2


x_2
1
.8
.6
.4
.2
0
$\mathcal{R}_1$
$\mathcal{R}_2$
.2
.4
.6
.8
1
x_1


$-1.2\,x_1 + x_2 < 0.1$
ω_2
ω_1


ANY
questions?