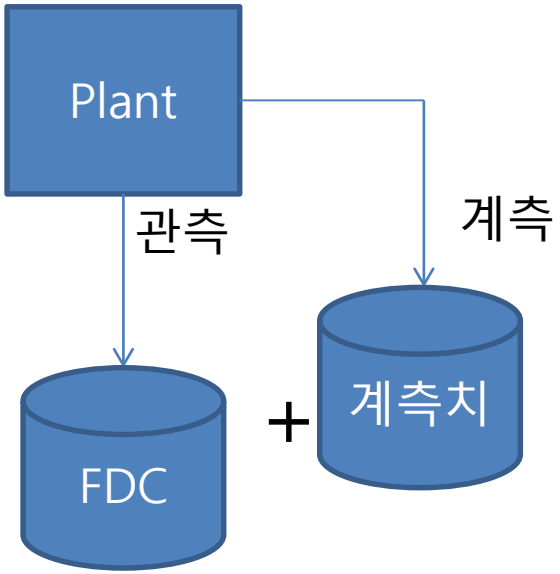
비선형 분류 모델링

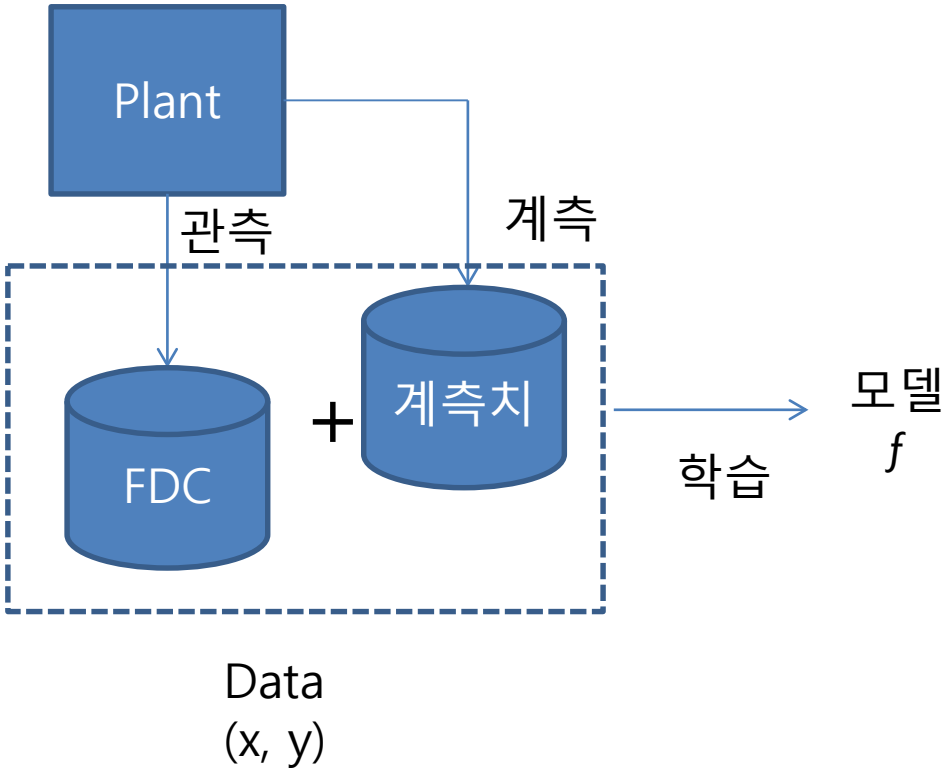
의사결정나무

Decision Tree

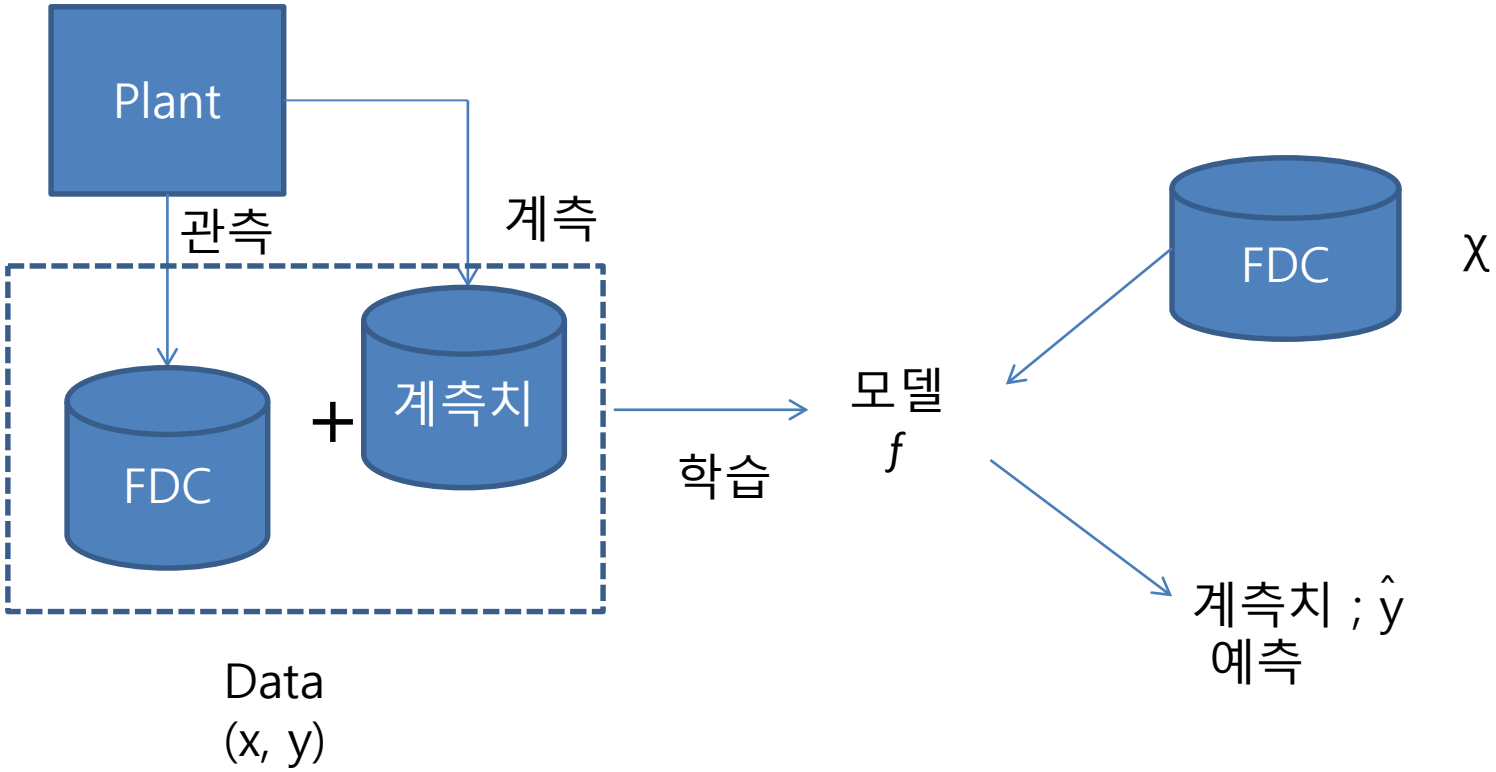
교사 학습 패러다임



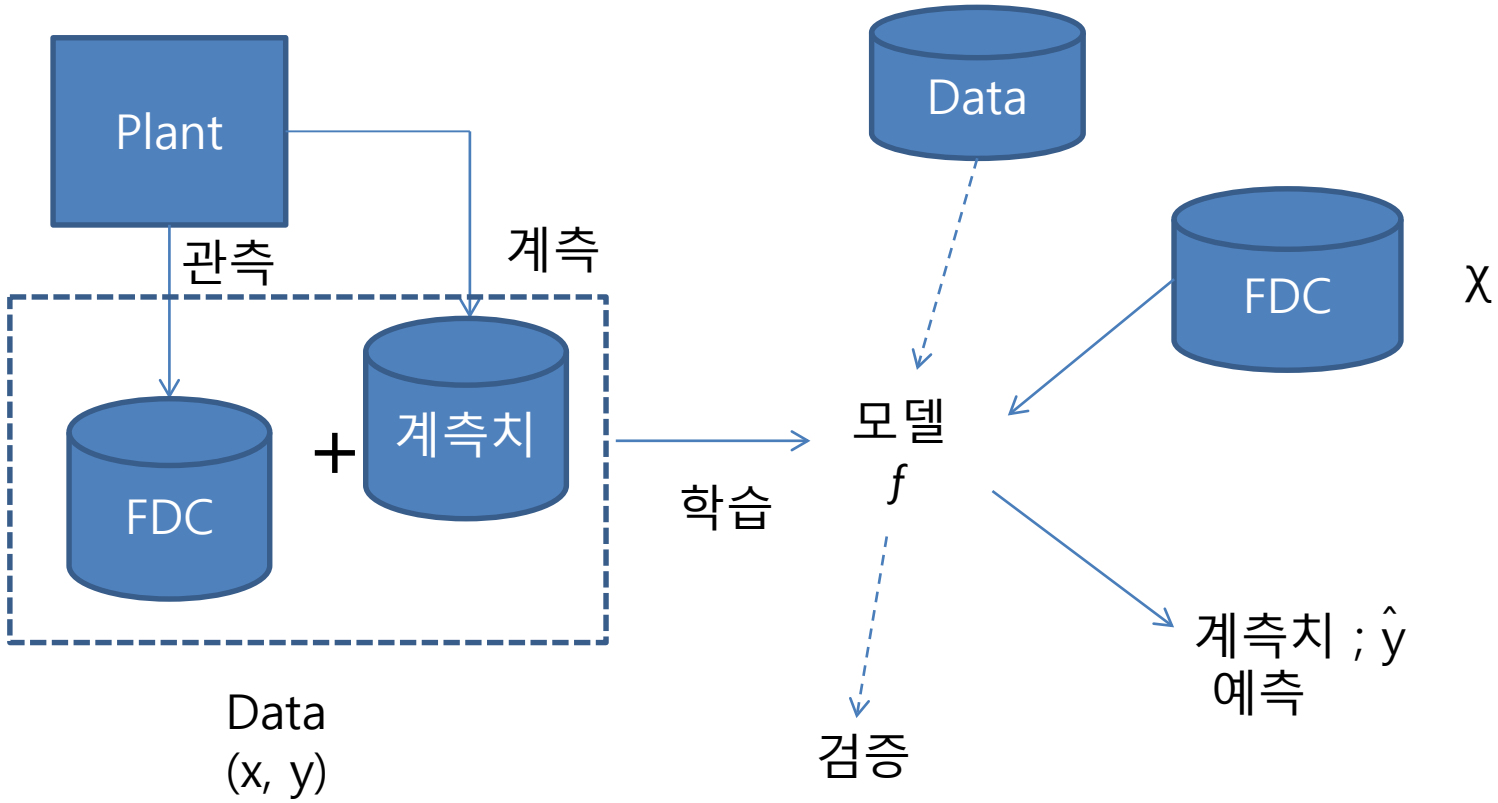
교사 학습 패러다임



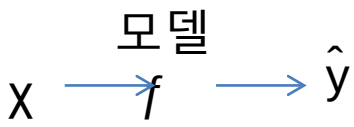
교사 학습 패러다임



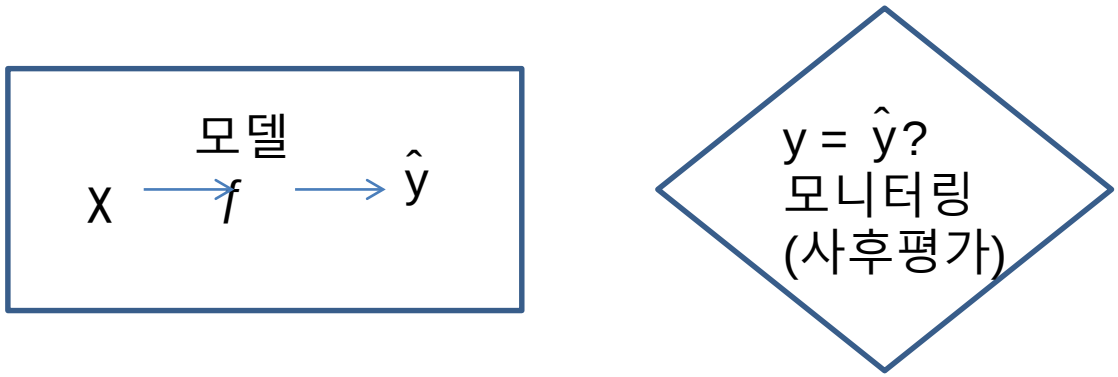
교사 학습 패러다임



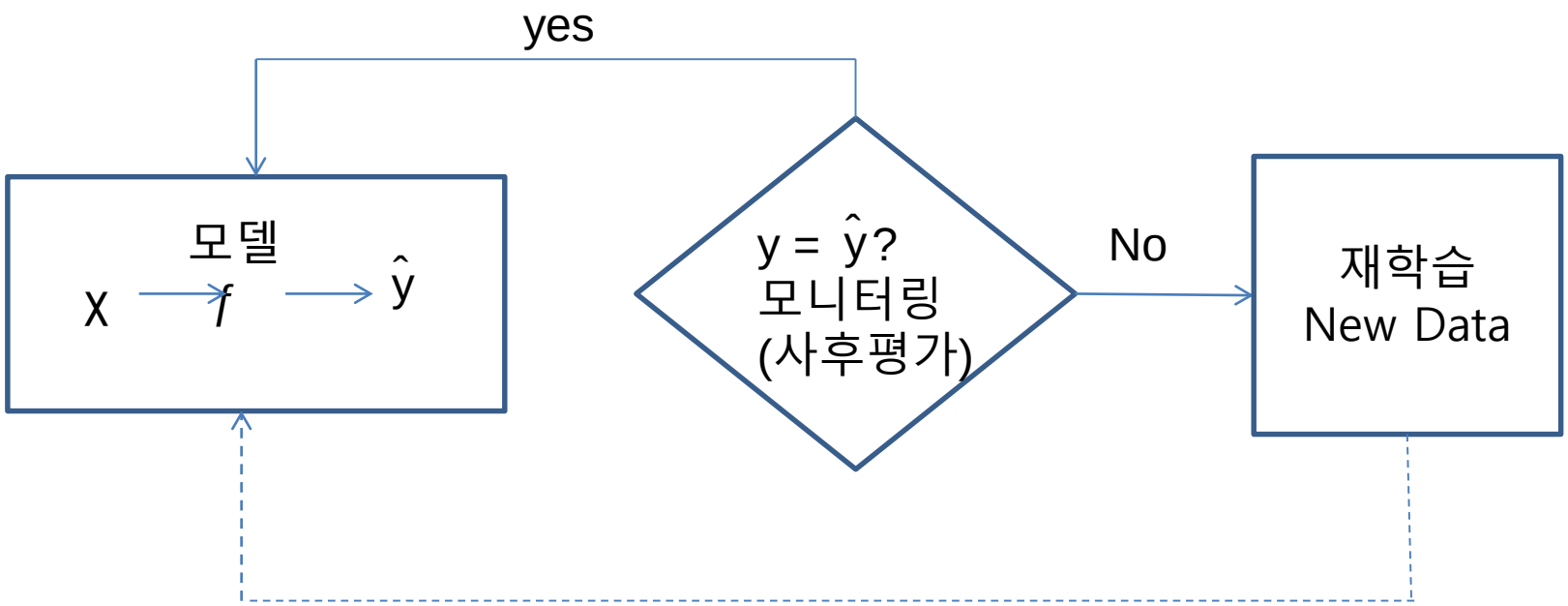
모델의 현업 배치, 모니터링 및 재학습



모델의 현업 배치, 모니터링 및 재학습



모델의 현업 배치, 모니터링 및 재학습



분류

- 일종의 회귀분석
- y 값이 연속이 아닌 범주
  - 0/1, 불량/정상, 이탈/잔류
- y 값을 “확률” 또는 “스코어”로 바꾸면?

Prediction = Classification

- 종속변수(y)를 독립 변수 (x) 들의 함수 (f) 로 적합,
- 즉 데이터 {(x,y)} 로 부터 y = f (x) 의 f 를 찾는다

- **회귀분석**, 신경회로망, 사례기반추론, **의사결정나무**

- 예: y 무엇을 예측할 수 있는가?
  - 소비자가 마케팅 켐페인에 반응할 확률 => 반응 여부
  - 휴대폰 고객이 향후 6개월 내에 이탈할 확률 => 이탈 여부
  - 와인의 품질 => 품질 등급
  - 반도체 웨이퍼의 수율 => 수율 등급
  - 선박 건조 기간 => 기간 등급

Prediction

- Y 가 결정이 된 후에는…
- 무엇으로 y 를 예측하려고 하는가? 즉, x ???
- X 는 독립 변수 또는 “예측 변수 predictive variable”

- X 선택 기준
  - Y 와의 정확한 함수 관계를 알고 있다.
  - Y 와의 정확한 함수 관계는 모르지만, 영향을 준다는 걸 100% 확실
  - Y 와의 정확한 함수 관계는 모르지만, 영향을 줄 수 있는 가능성이 있다.

Predictive Analytics

- A new fancy name of “Supervised” data mining, or Regression and Classification

- “Find a mapping/function f such that y = f(x) given data set D = {(x,y)}”
- Regression when y is continuous
- Classification when y is categorical/binary

Predictive Analytics

- Regression
  - Multiple Linear Regression
  - k-NN
  - Decision Tree Regression
  - Neural Networks
- Classification
  - Logistic Linear Regression, Discriminant Analysis
  - k-NN, Naïve Bayese
  - Decision Tree Classifier
  - Neural Networks
  - SVM

의사결정나무 Decision Tree

종로 데이트코스 심리 테스트

YES → NO ⇢

로미오와 줄리엣처럼 운명적인 사랑을 해보고 싶다

가끔은 탁 트인 곳에 머물고 싶다

낮보다는 야간 데이트를 즐긴다

연인과 낭만적인 분위기를 즐기고 싶다

가끔은 나도 드라마 같은 사랑을 꿈꾼다

무드와 느낌을 매우 중요시한다

내 애인에 대해 아직 모르는 게 너무 많다

애인과 만난 지 1년이 넘었다

나는 사진 찍는 것을 좋아한다

데이트 때 영화를 즐겨 보는 편이다

조용한 곳에서 둘만의 시간을 갖고 싶다

식상한 데이트는 이제 지겹다

작업파

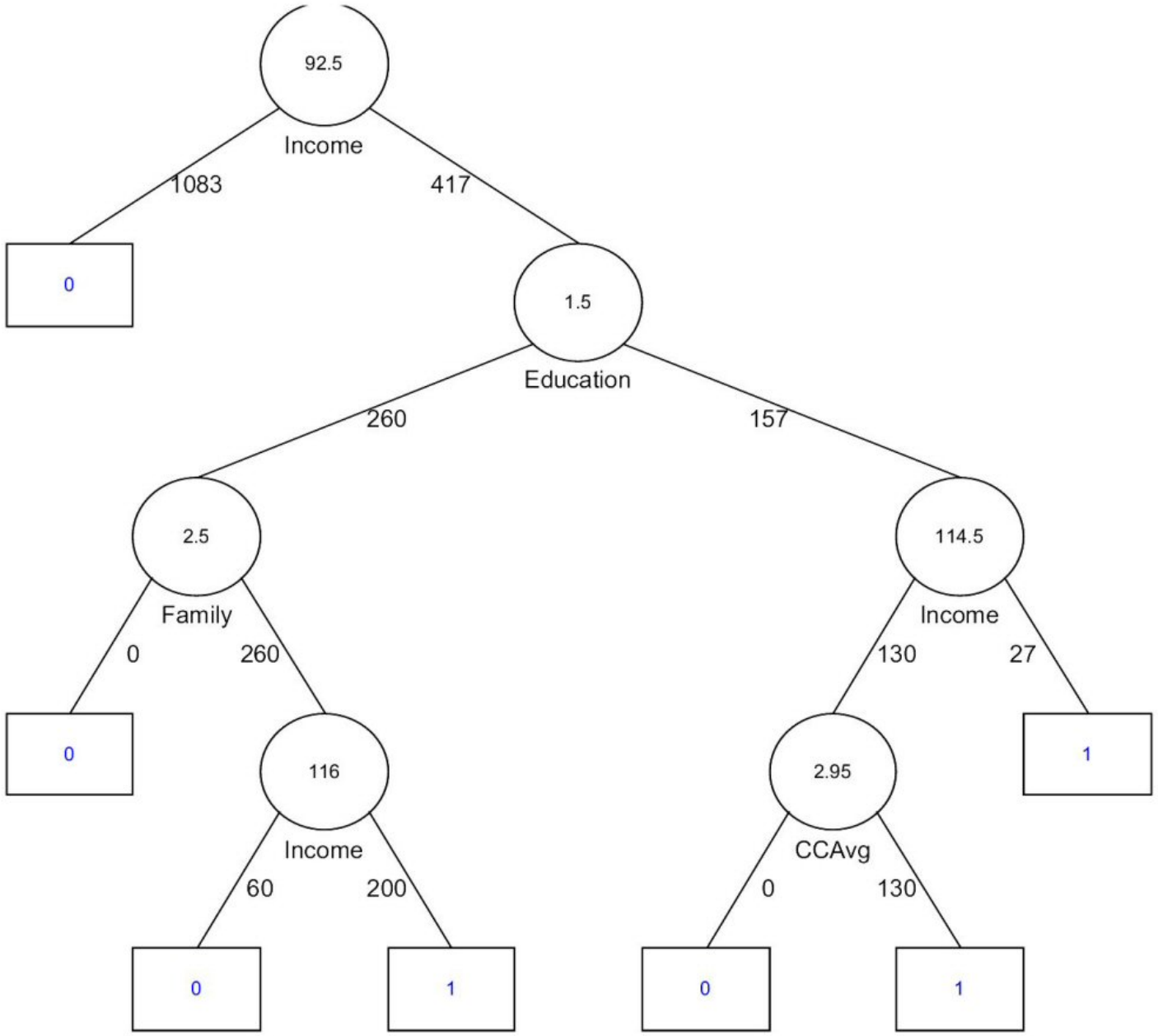
종로3가~종각~청계천~인사동~삼청동

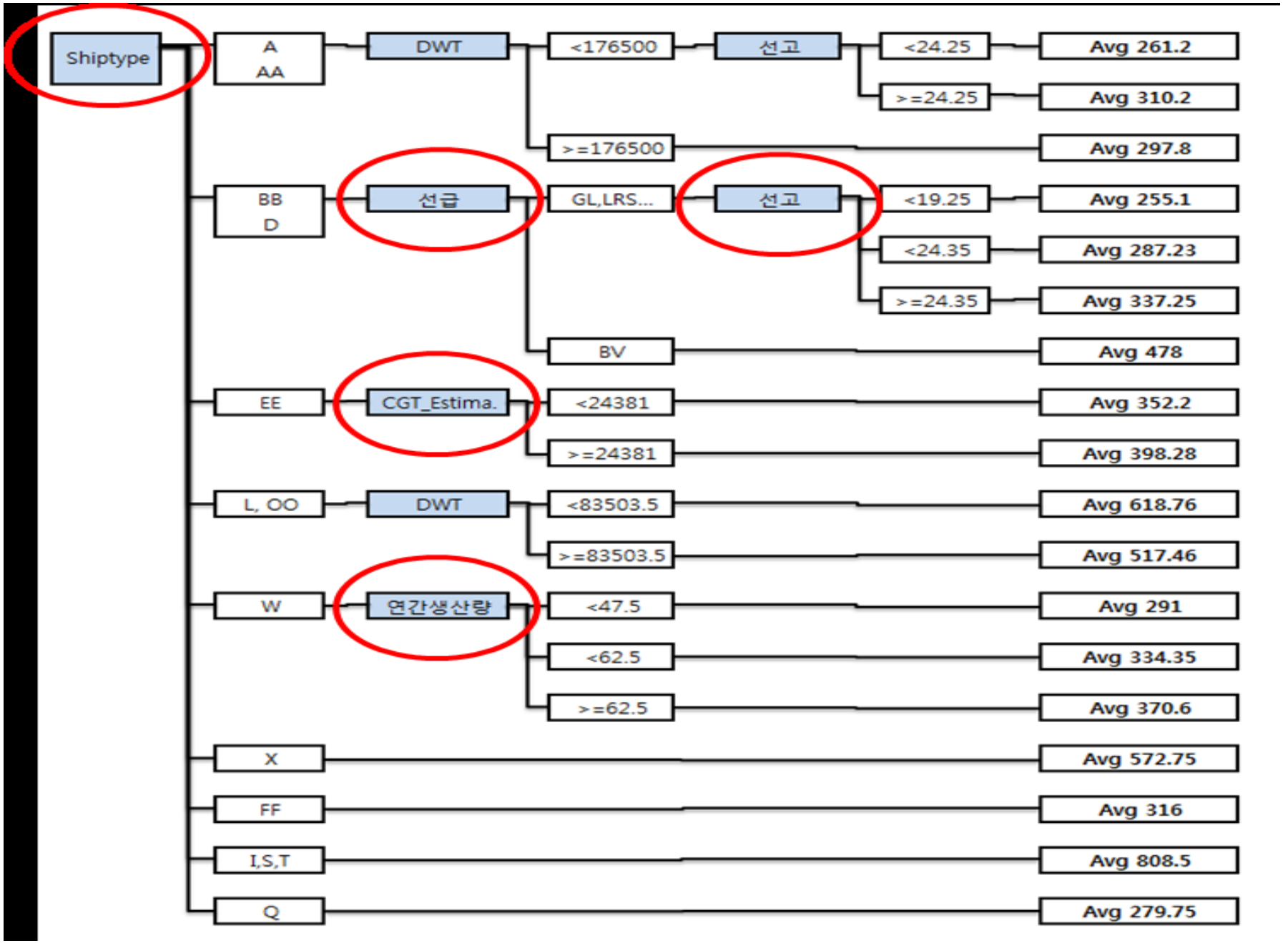
로맨틱파

창의문~환기미술관~부암동 능금나무 옛길~커피프린스1호점 촬영지~백사실 계곡~삼청공원~인사동

정열파

대학로~낙산공원~흥인지문~청계천~종각 나이트클럽

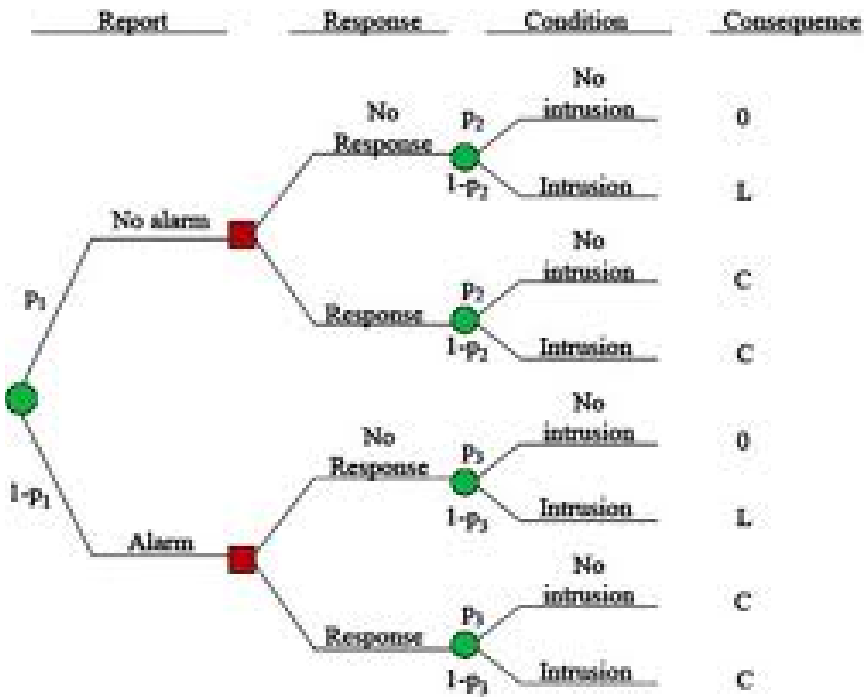

92.5
Income
1083
417
0
1.5
Education
260
157
2.5
Family
0
260
114.5
Income
130
27
0
116
Income
60
200
2.95
CCAvg
0
130
1
0
1
0
1


Shiptype
A
AA
DWT
<176500
선고
<24.25
Avg 261.2
>=24.25
Avg 310.2
>=176500
Avg 297.8
BB
D
선급
GL,LRS...
선고
<19.25
Avg 255.1
<24.35
Avg 287.23
>=24.35
Avg 337.25
BV
Avg 478
EE
CGT_Estima.
<24381
Avg 352.2
>=24381
Avg 398.28
L, OO
DWT
<83503.5
Avg 618.76
>=83503.5
Avg 517.46
W
연간생산량
<47.5
Avg 291
<62.5
Avg 334.35
>=62.5
Avg 370.6
X
Avg 572.75
FF
Avg 316
I,S,T
Avg 808.5
Q
Avg 279.75

앞 세 가지 tree 의 차이는?

- 데이트 코스; 전문가의 '지식'을 바탕으로 구축
- 구매 여부: 데이터로부터 자동으로 구축
- 생산 기간: 데이터로부터 자동으로 구축

의사결정 나무 (의사결정 이론)



Trees and Rules

Goal: Classify or predict an outcome based on a set of predictors

The output is a set of **rules**

Example:

- Goal: classify a record as “will accept credit card offer” or “will not accept”
- Rule might be “IF (Income > 92.5) AND (Education < 1.5) AND (Family <= 2.5) THEN Class = 0 (nonacceptor)
- Also called CART, Decision Trees, or just Trees
- Rules are represented by tree diagrams

핵심 아이디어

재귀적 분할: 반복적으로 레코드를 두 개의 파트로 나눔. 따라서 최대의 동질성을 얻음

가지치기: 과적합을 피하기 위해 지엽적 가치를 침으로써 나무를 간결화

재귀적 분할

재귀적 분할 단계

- 예측변수 중 하나, x_i 선택
- x_i 의 값, 말하자면 s_i를 선택, 학습 데이터를 두 개의 부분으로 나눔(반드시 같은 필요는 없음)
- 그 결과로 나온 부분들이 각각 얼마나 "순수"한가 또는 동질적인가 축정

  "순수" = 대개 하나의 클래스 레코드들을 포함
- 알고리즘은 최초 분할에서 순수성을 최대화하기 위해 x_i와 s_i의 다양한 값들을 시도
- "최대 순수성" 분할을 얻은 후에, 2번째 분할 과정 반복 등등

예: 승차식 잔디깎기

- 목표: 승차식 잔디깎기를 소유하거나 소유하지 않은 24개의 가정 분류
- 예측변수 = 수입, 주택대지 크기

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

어떻게 분할하나

- 하나의 변수, 말하자면 주택대지 크기에 따라 레코드 정렬
- 연속적 값 사이의 중심점 찾기
  E.g. 첫 중심점은14.4 (14.0과 14.8 사이의 가운데)
- 레코드를 lotsize > 14.4과 lotsize < 14.4으로 분할
- 분할 평가 후 다음 것 시도, 즉 15.4 (14.8과 16.0 사이의 가운데)

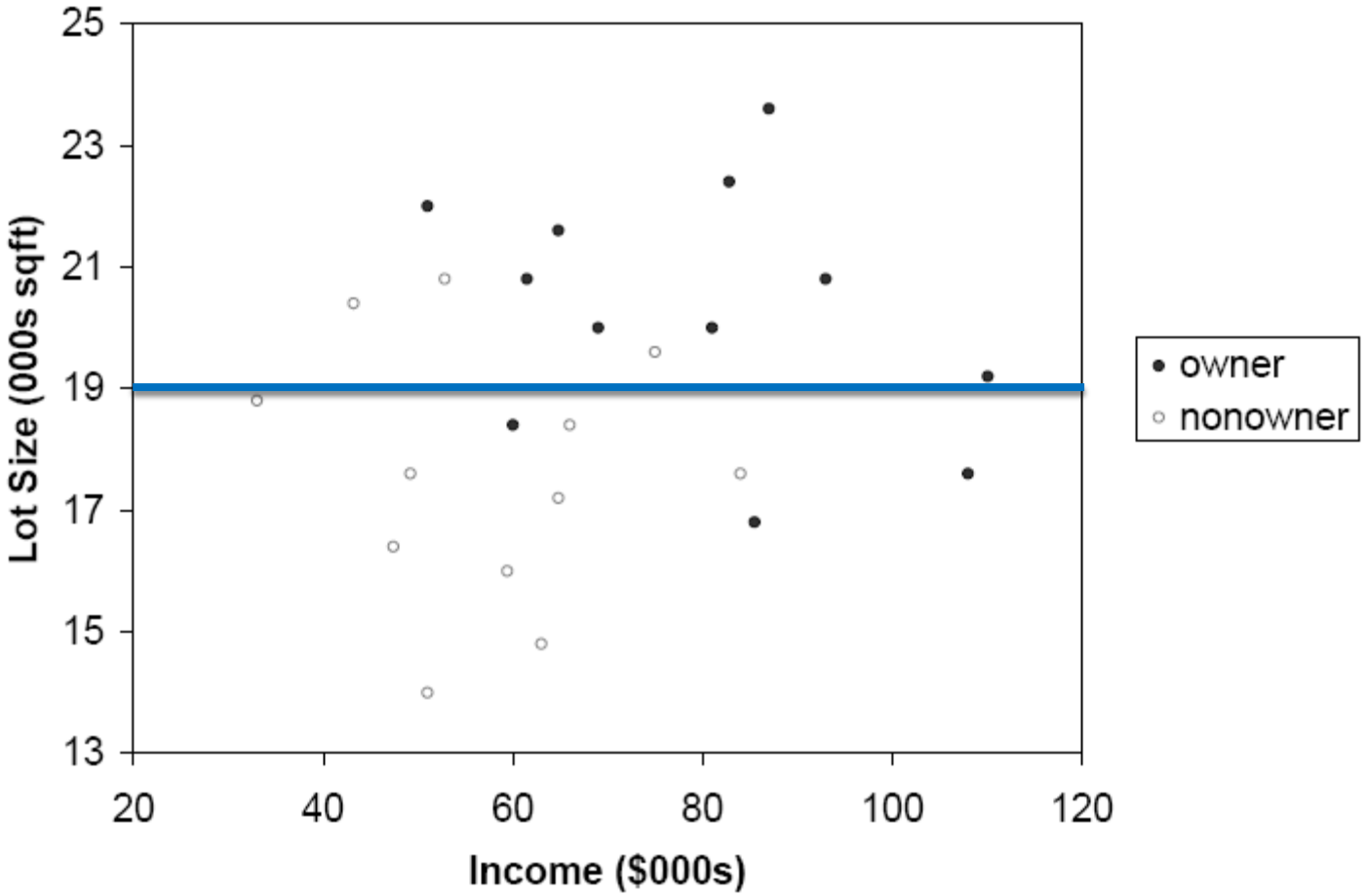
Note: 범주적 변수

- 범주가 분할될 수 있는 모든 가능한 방법 조사

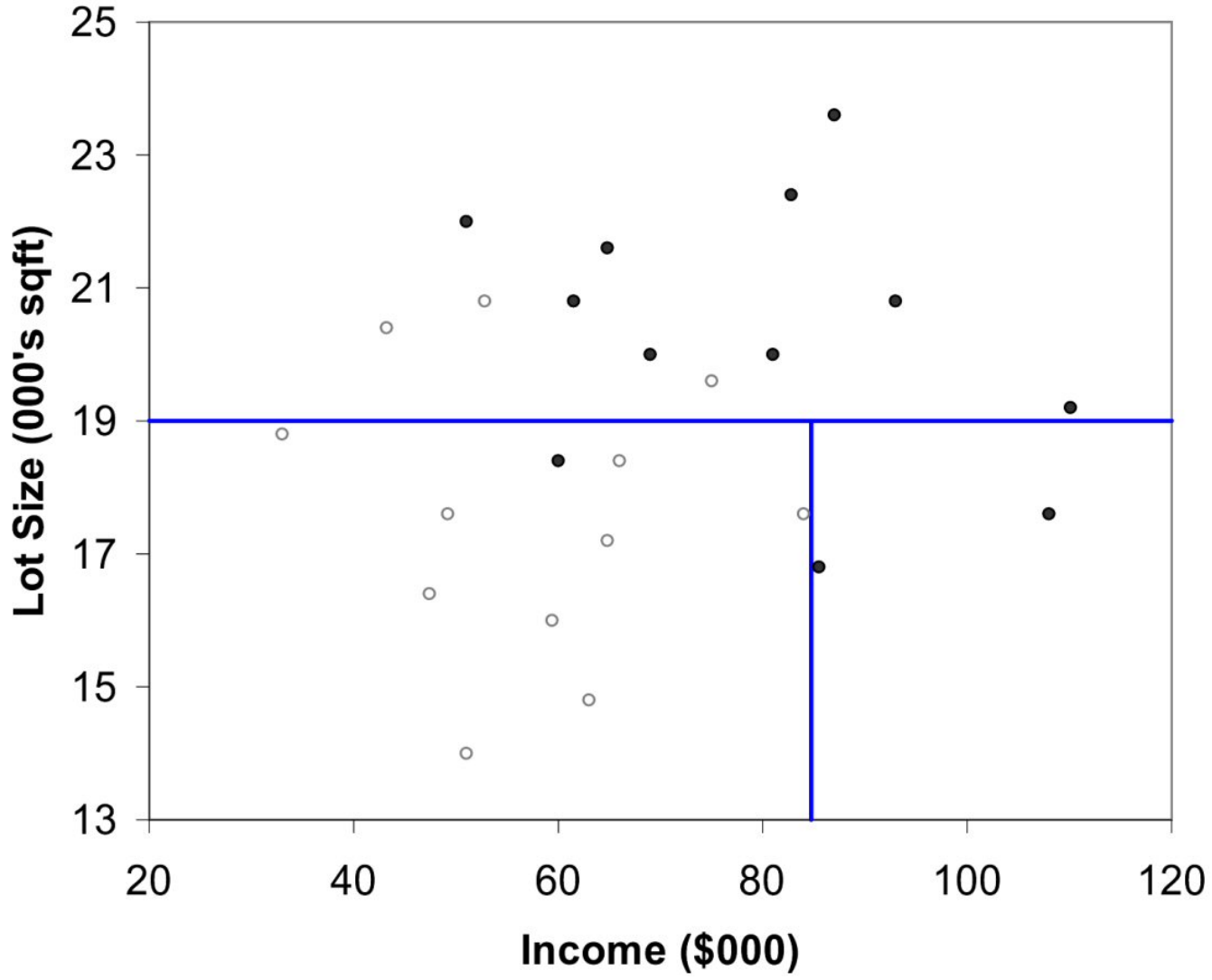
- E.g., 범주 A, B, C는 3가지 방식으로 분할될 수 있음
  {A} and {B, C}
  {B} and {A, C}
  {C} and {A, B}

- 많은 범주에서, 분할 수는 거대해짐
- XLMiner는 오직 이항 범주형 변수만 제공

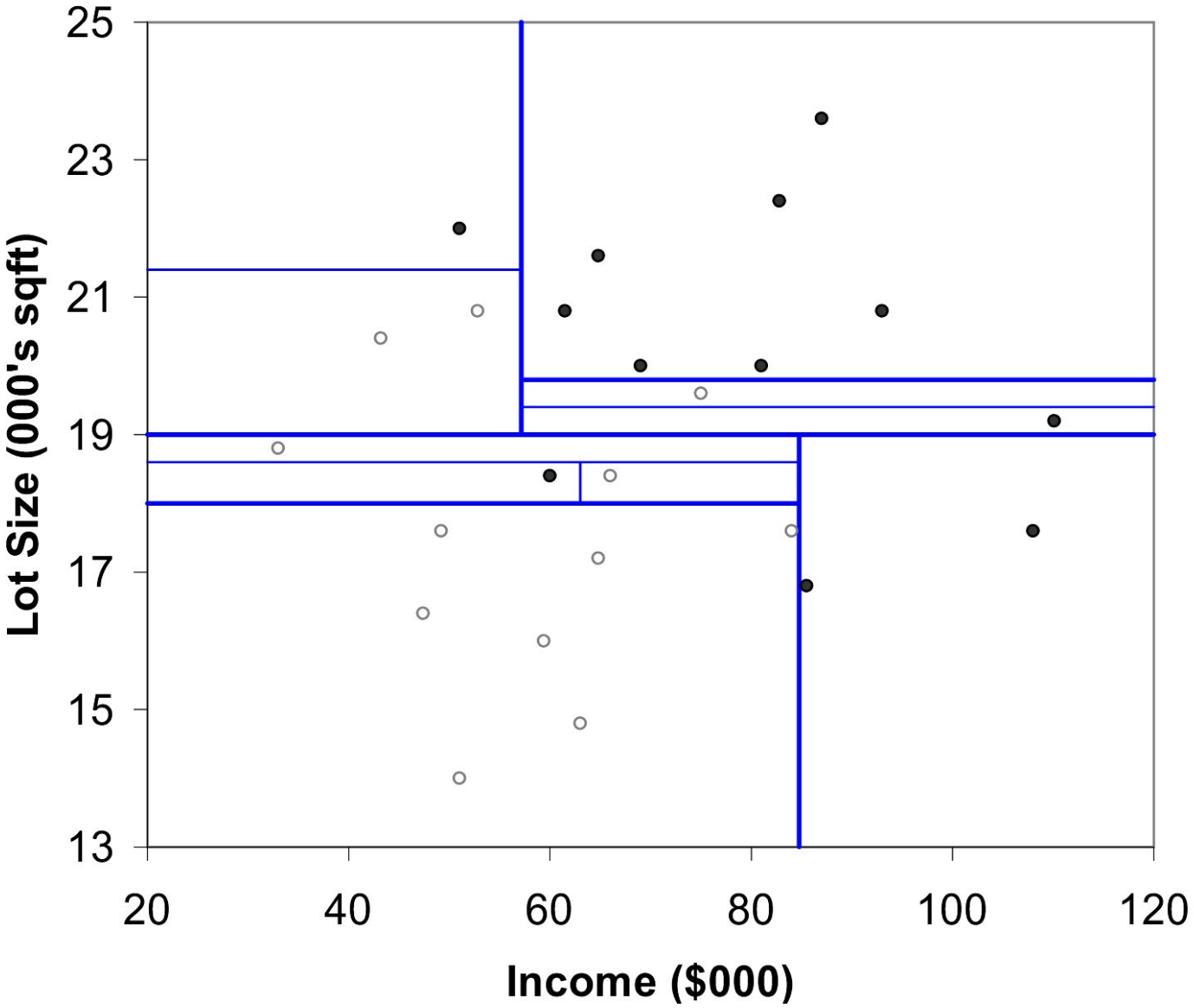
첫 번째 분할: Lot Size = 19,000



두 번째 분할: Income = $84,000



모든 분할 후


25
23
21
19
17
15
13
Lot Size (000's sqft)
20
40
60
80
100
120
Income ($000)

불순도 측정

데이터

- 집합 X 에 두 종류의 원소가 있다.
- 이 집합의 불순도는 얼마인가?
- 예: 바둑알 흰색과 검정색
  - X = {B, B, B, B, W, W, W, W} 불순도가 높은가?
  - Y = {B, B, W, W, W, W, W, W} 불순도가 높은가?
  - Z = {W, W, W, W, W, W, W, W} 불순도가 높은가?
  - Z'= {B, B, B, B, B, B, B, B} 불순도가 높은가?
  - X'= {B, W} 불순도가 높은가?
- 불순도를 측정하는 방법?

지니 지수

*M*개의 레코드를 지닌 직사각형 A에 대한 지니 지수

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

p = 클래스 k에 속하는 직사각형 A에서의 케이스 비율

- I(A) = 0 모든 케이스가 같은 클래스에 속할 때
- 모든 클래스가 똑같이 표현될 때 최댓값 (= 0.50, 이항 케이스에서)

Note: XLMiner uses a variant called “delta splitting rule”

엔트로피

$$entropy\,(A) = -\sum_{k=1}^{m} p_k \log_2(p_k)$$

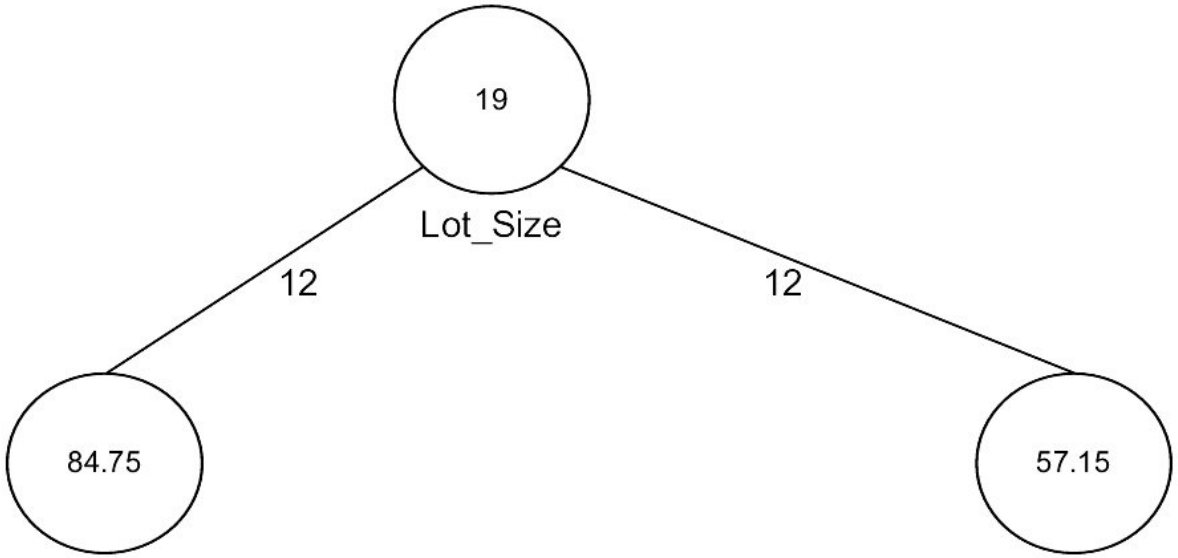
p = 클래스 k에 속하는 직사각형 A에서
(m으로부터 나오는) 케이스 비율

- 엔트로피는 0(가장 순수)과 $\log_2(m)$
(클래스가 똑같이 표현) 사이에 분포

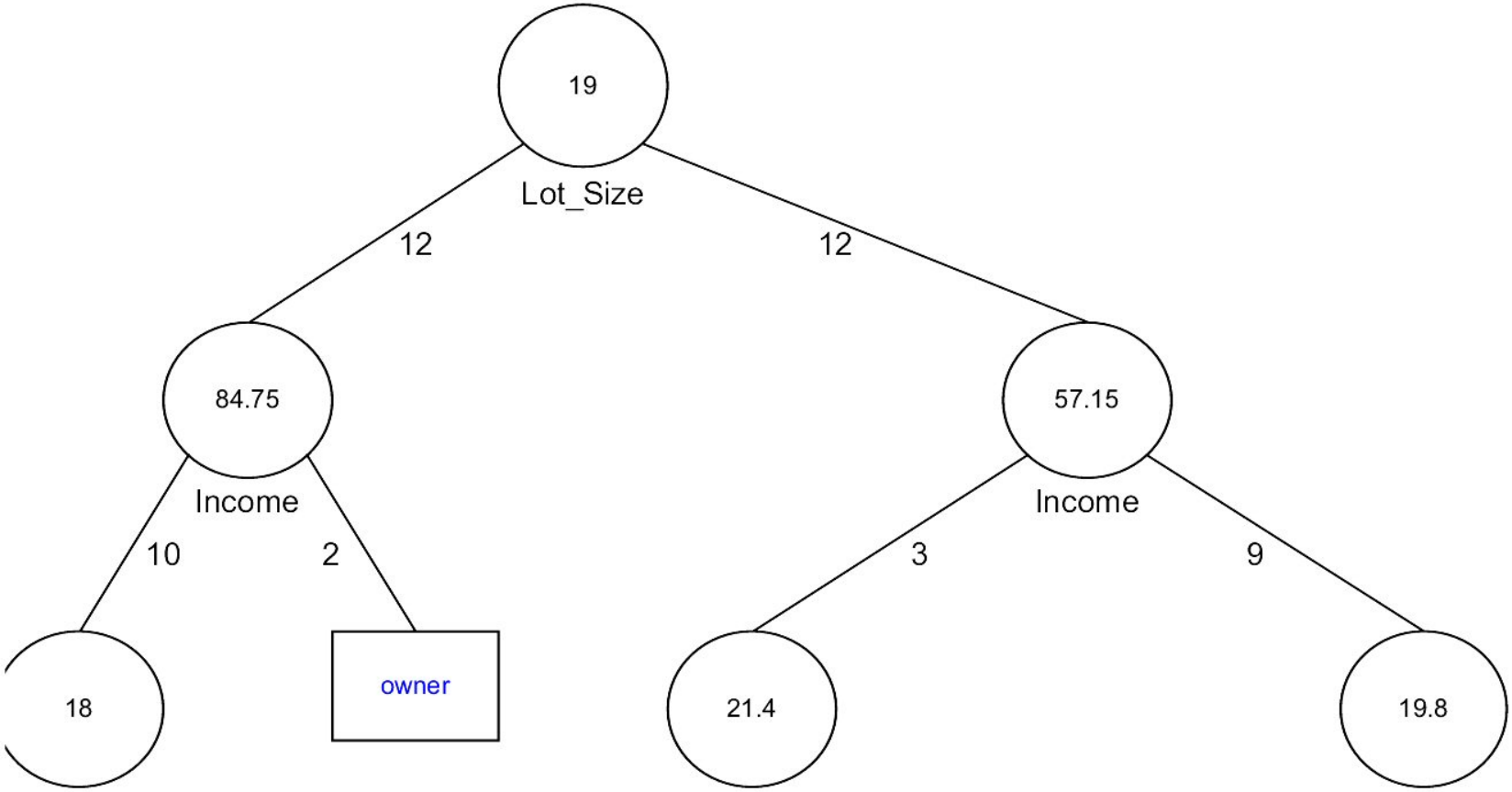
불순도와 재귀적 분할

- 전반적 불순도 측도의 얻는다 (개개 직사각형의 가중 평균).
- 각각의 연속적 단계에서, 이 측도를 전체 변수에서 모든 가능한 분할을 가로질러 비교한다.
- 불순도가 가장 축소하는 분할을 고른다.
- 고른 분할점은 나무의 노드가 된다.

첫 번째 분할– 나무



세 번째 분할 후 나무


19
Lot_Size
12
12
84.75
57.15
Income
Income
10
2
3
9
18
owner
21.4
19.8

나무 구조

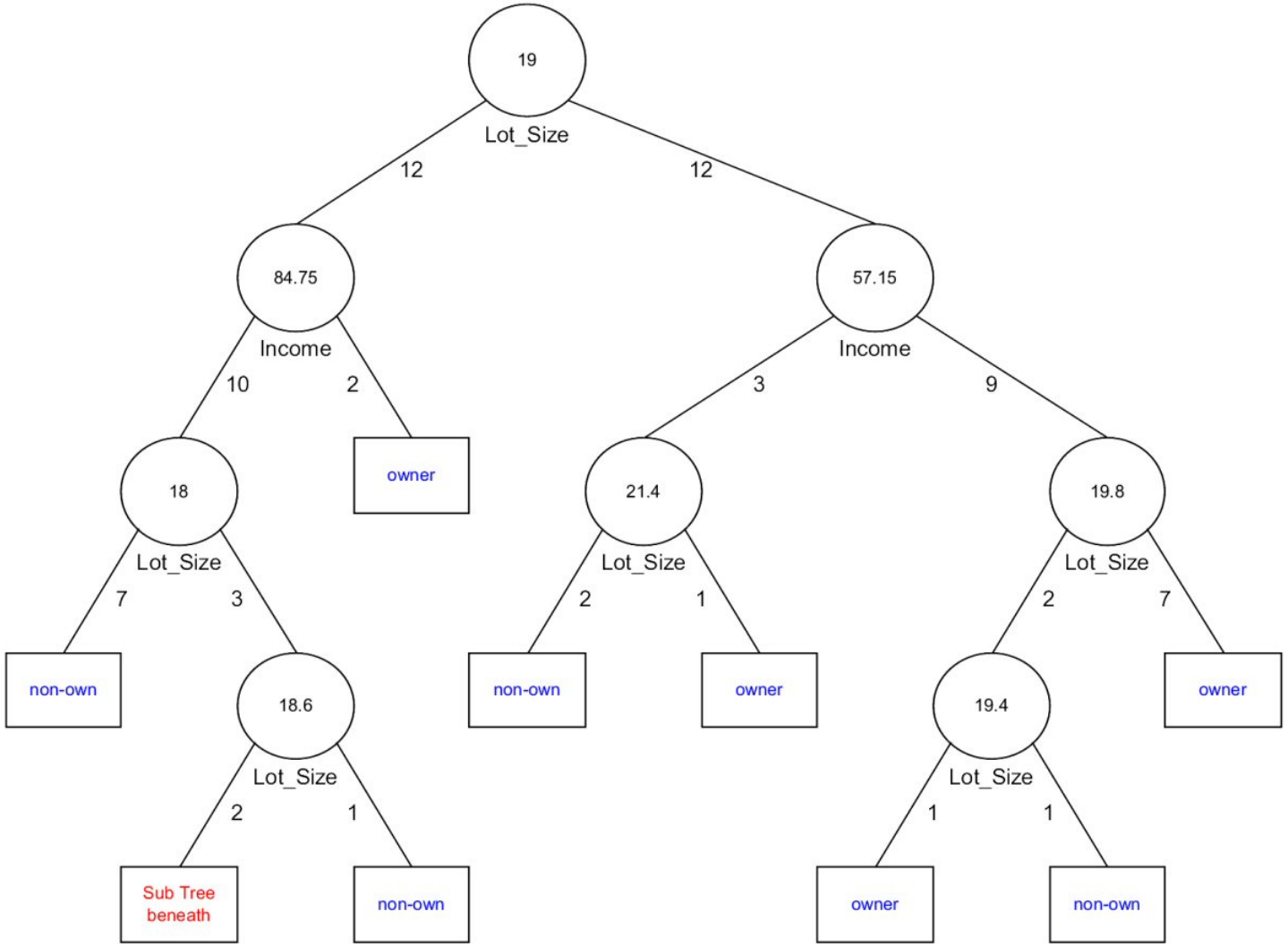
- 분할점은 나무에서 노드가 된다(중앙에 분할값을 갖는 원)
- 직사각형은 “잎”을 나타낸다(종료점, 더 이상 분할 없음, 측정된 분류값)
- 노드 사이의 선의개수는 케이스의 수를 지시한다.
- 규칙을 파생하는 나무를 읽는다.

  E.g., If lot size < 19, and if income > 84.75, then class = “owner”

잎 노드 라벨 결정

- 각각의 잎 노드 라벨은 그 안의 레코드들의 “투표”, 그리고 기준값에 의해 결정된다.
- 각각의 잎 노드 내의 레코드들은 학습 데이터에서 온다.
- 기본 기준값=0.5은 잎 노드의 라벨이 다수 클래스라는 것을 의미한다.
- 기준값 = 0.75: 대다수 75% 또는 “1” 노드로 라벨 붙이는 잎에서 “1” 레코드 이상을 요구한다.

모든 분할 후 나무

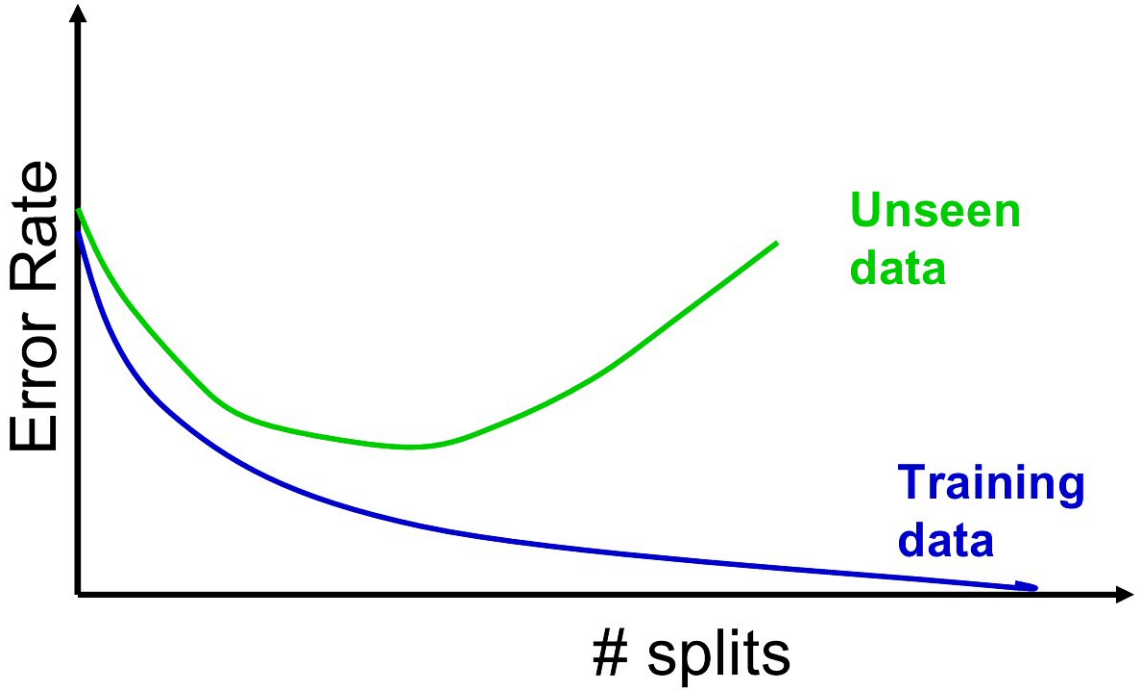

19
Lot_Size
12
12
84.75
57.15
Income
Income
10
2
3
9
18
owner
21.4
19.8
Lot_Size
Lot_Size
Lot_Size
7
3
2
1
2
7
non-own
18.6
non-own
owner
19.4
owner
Lot_Size
Lot_Size
2
1
1
1
Sub Tree beneath
non-own
owner
non-own

과적합 문제

나무의 성장을 멈춤

- 과정의 자연적 종결은 각각의 잎에서 100% 순수성이다.
- 이것은 데이터를 **과적합**하는데, 이는 데이터에서 소음의 적합으로 끝난다.
- 과적합은 새로운 데이터의 예측 정확성을 낮춘다.
- 어떤 지점을 지나, 검증 데이터의 오분류율이 증가하기 시작한다.

완전성장한 나무의 오분류율



CHAID

CART보다 오래된 CHAID는 나무의 성장을 제한하는 카이제곱 통계법을 사용한다.

분할은 순도의 증가가 통계적으로 유의미하지 않을 때 멈춘다.

가지치기

- CART은 나무의 성장을 최대한까지 이르게 한다, 그러고 난 후에는 가지치기를 한다.
- 아이디어는 검증 오류가 나타나기 시작하는 지점을 찾는 것이다.
- 잎을 가지치기함으로써 연속적으로 더 작은 나무들을 생성한다.
- 각 가지치기 단계에서, 다중 나무가 가능하다.
- 그 단계에서 최적의 나무를 고르기 위해 *비용 복잡성*을 이용한다.

비용 복잡성

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

CC(T) = 나무의 비용 복잡성

Err(T) = 오분류된 레코드의 비율

α = 나무 사이즈에 부착된 벌점 요인(사용자 책정)

- 주어진 크기의 나무 중에서 비용 복잡성이 가장 낮은 것을 고른다.
- 각각의 나무 사이즈에 대해 이를 행한다.

가지치기에 검증 오류 사용

가지치기 과정은 일련의 다양한 크기의 나무와 그에 관련된 오분류율을 산출한다.

관련된 두 개의 나무:

.

- 최소 오류나무

  검증 데이터에서 가장 낮은 오분류율을 갖는다

- 최적의 가지치기된 나무

  최소 오차의 표준 오차 내에서 가장 작은 나무

  간결성/간명성을 더한다.

가지치기된 나무의 오분류율

# Decision Nodes	% Error Training	% Error Validation	
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	
14	1.16	1.533333	
13	1.16	1.6	
12	1.2	1.6	
11	1.2	1.466667	<-- Min. Err. Tree
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	<-- Best Pruned Tree
5	4.44	1.8	
4	5.08	2.333333	
3	5.24	3.466667	

UP Sell 적용 사례

- 우량 고객 30만 가운데 4,887명 플래티넘 카드 사용자
- 나머지 295,123 명 비 사용자 가운데 누구를 타켓팅 할 것인가?

- “기존 플래티넘 사용자와 유사한 구매 행태를 보이는 고객이 가능성이 높을 것”
- Upselling 을 위한 타겟 마케팅

- HOW?
  - 의사결정 나무 IF THEN rule

- ***특급호텔*** 11만원 이상 & ***항공사*** 이용
  - 787명 (Platinum 93.1%)
- ***골프장*** 48만원 이상 & ***일식*** 10만원 이상 & ***항공사*** 이용 안 함
  - 151명 (Platinum 92.7%)
- ***골프장*** 7만원 이상 & ***일식*** 24만원 미만 & ***특급호텔*** 11만원 미만 & ***항공사*** 이용
  - 90명 (Platinum 93.3%)

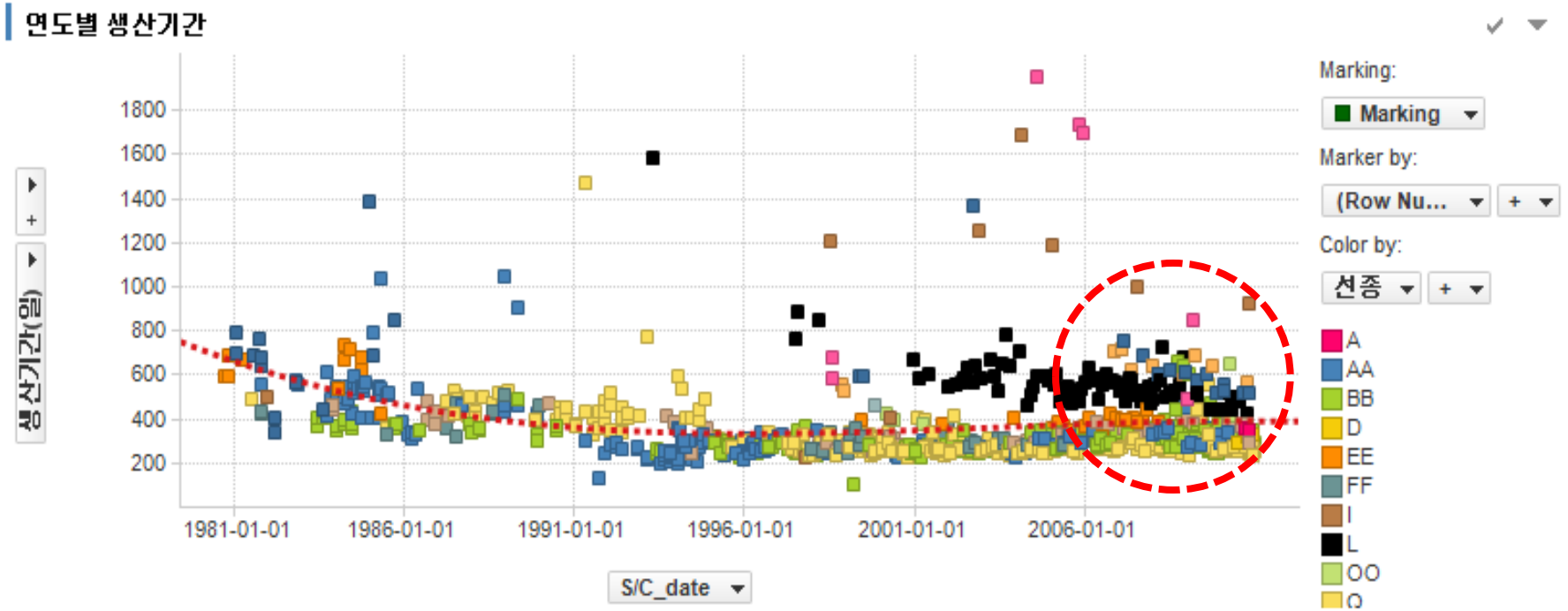
Platinum 지수*	고객 수
90~100	9,116
80~90	6,890
70~80	19,916
60~70	8,908
50~60	11,798
40~50	12,513
30~40	24,974
20~30	35,968
10~20	165,040

Sales bidding 적용 사례

호선의 생산기간 예측

호선의 생산기간 예측

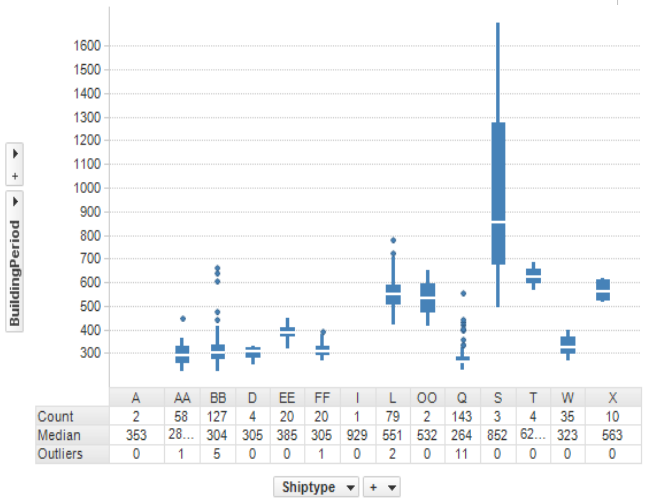
- 선박 영업 단계에서는 주문 선박 Spec 이 자세히 결정되지 않으나, 대략적인 생산기간 예측이 필요함
- 생산기간을 결정하는 예측 요인에 대한 규명이 필요함



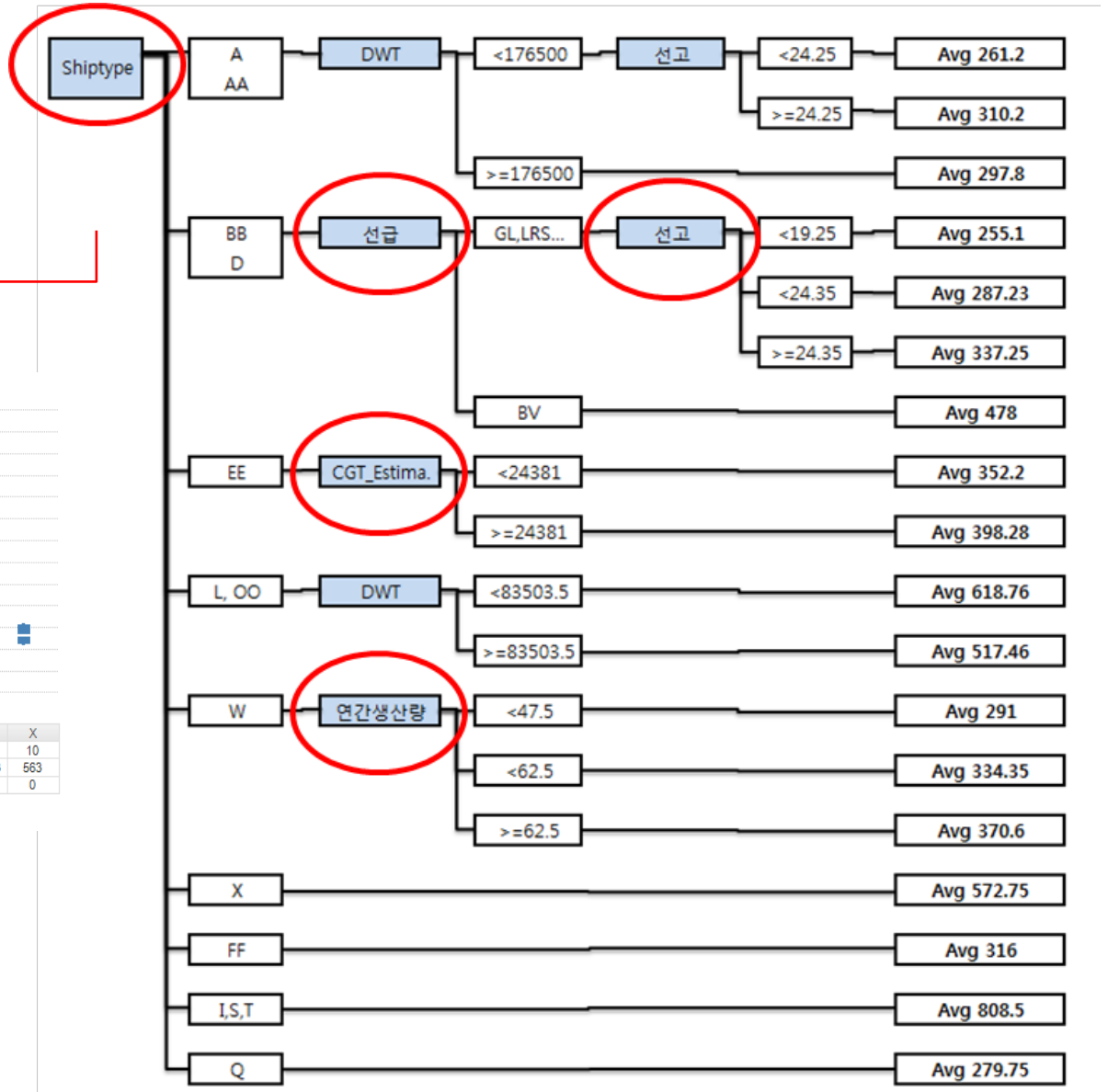
호선 별 생산 기간은 감소 추세이나 2008년 이후 전반적인 생산기간 증가 추세임

특정 선종의 경우 제작 이후 생산 기간이 꾸준히 감소하는 경향이 있음 (L 선종)

호선의 생산기간 예측



	A	AA	BB	D	EE	FF	I	L	OO	Q	S	T	W	X
Count	2	58	127	4	20	20	1	79	2	143	3	4	35	10
Median	353	28...	304	305	385	305	929	551	532	264	852	62...	323	563
Outliers	0	1	5	0	0	1	0	2	0	11	0	0	0	0



회귀나무

예측을 위한 회귀나무

- 연속적 결과 변수를 사용
- 절차는 분류나무와 유사
- 분할이 많이 시도됨, 불손도를 최소화하는 것을 선택

CT와의 차이

- 예측은 직사각형에서 수치형 타깃 변수의 **평균**으로 계산된 (CT에서는 다수결투표)
- 불순도는 잎 평균의 **제곱편차의 합**으로 측정된다.
- 성능은 RMSE (근의 평균제곱 오류)로 측정된다.

나무의 장점

- 사용하고 이해하기에 용이
- 해석하고 시행하기에 쉬운 규칙 생성
- 변수 선택과 축소가 자동
- 통계적 모델 가정을 요구하지 않음
- 실측 데이터를 광범위하게 다루지 않고도 작업 가능

단점

- 수평적 또는 수직적 분할에 의해 잘 포착되지 않는 데이터에서의 구조가 있을 때 잘 작동하지 않음
- 한 번에 하나의 변수를 다루기 때문에 변수들 사이의 상관관계를 포착할 방법이 없음

요약

- 분류나무와 회귀나무는 새로운 레코드를 예측하거나 분류하는 쉽고 투명한 방법이다.
- 나무는 일련의 규칙의 그래프적 표현이다.
- 나무는 학습 데이터의 과적합을 피하기 위해 가지치기를 해야만 한다.
- 나무가 데이터 구조에 대한 어떠한 가정도 갖지 않기 때문에, 보통 다량의 샘플이 필요하다.