Tree-Based Methods

An Introduction to Statistical Learning

황성원

SW

Outline

Tree-Based Methods

Supervised Learning

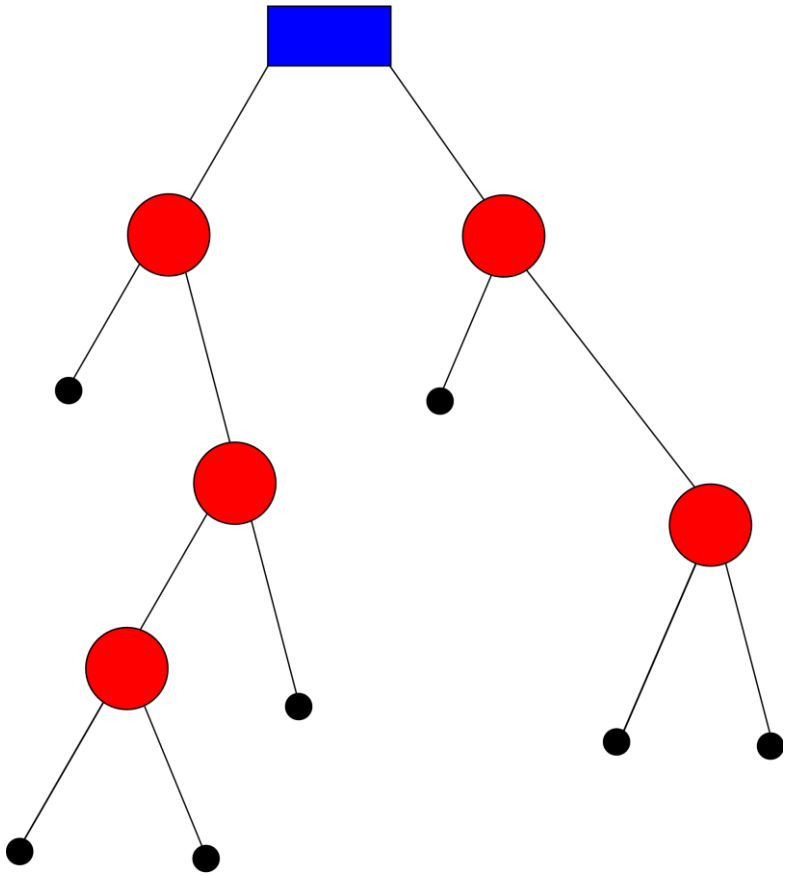
- **Decision Tree (의사결정트리) – 회귀/분류 트리**
- **+ 배깅, 랜덤 포레스트, 부스팅**

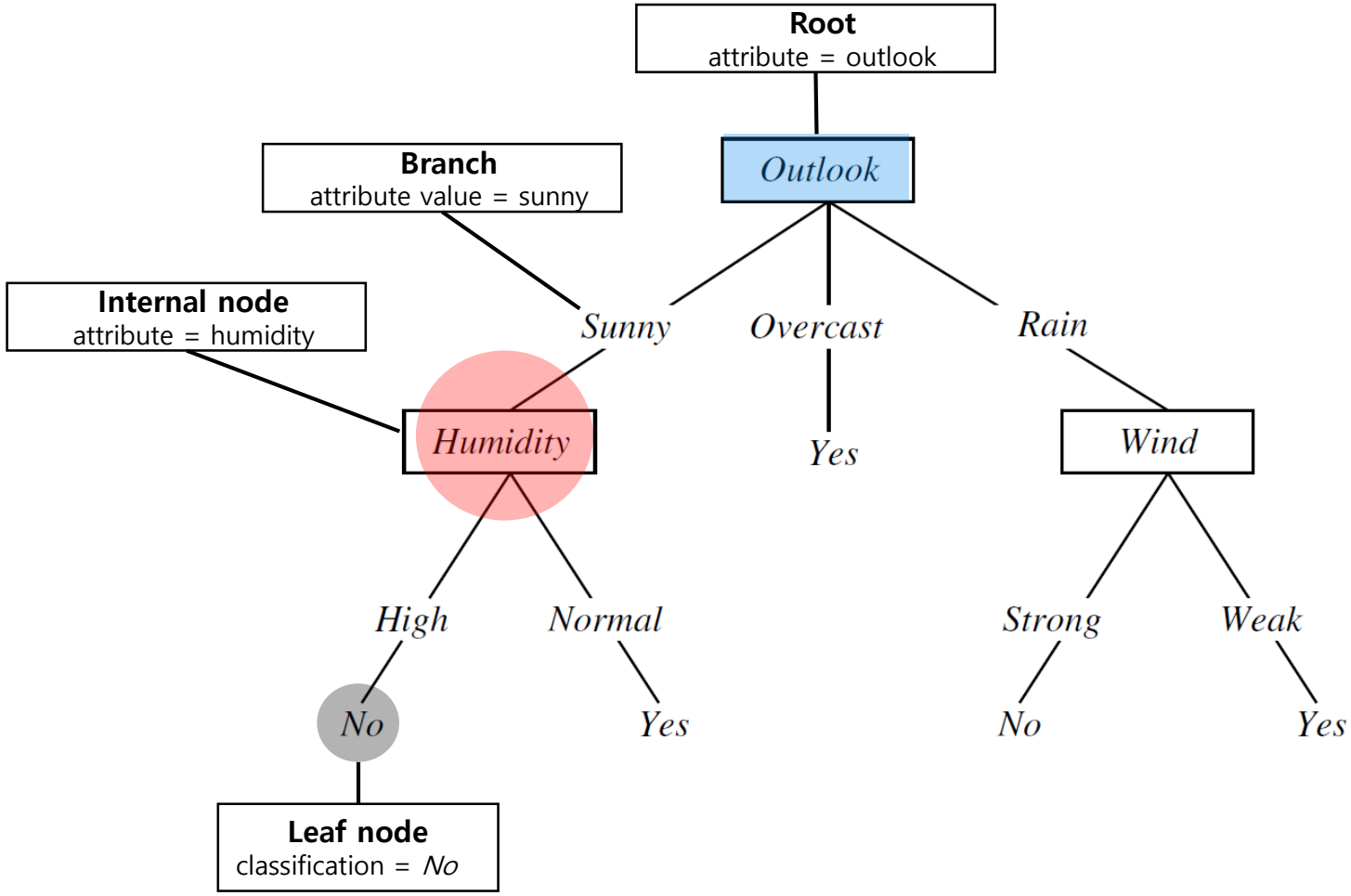
Unsupervised Learning

- **Clustering (군집화)**

의사 결정 트리 (Decision Tree) 란?

: 트리 구조를 통한 의사결정학습법



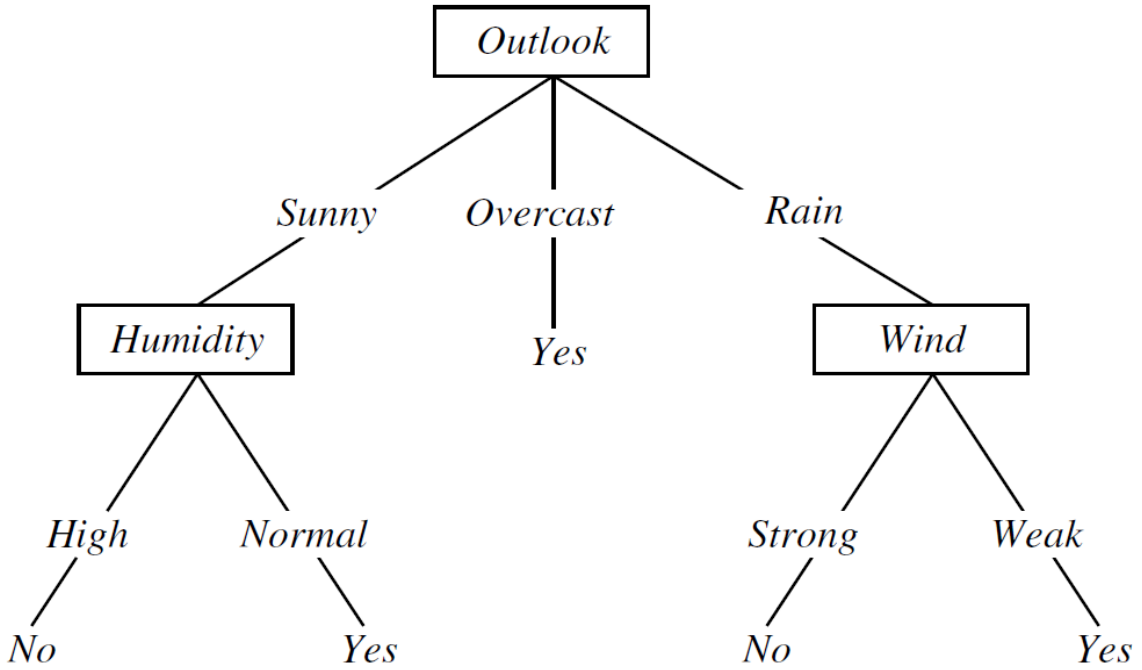


Root

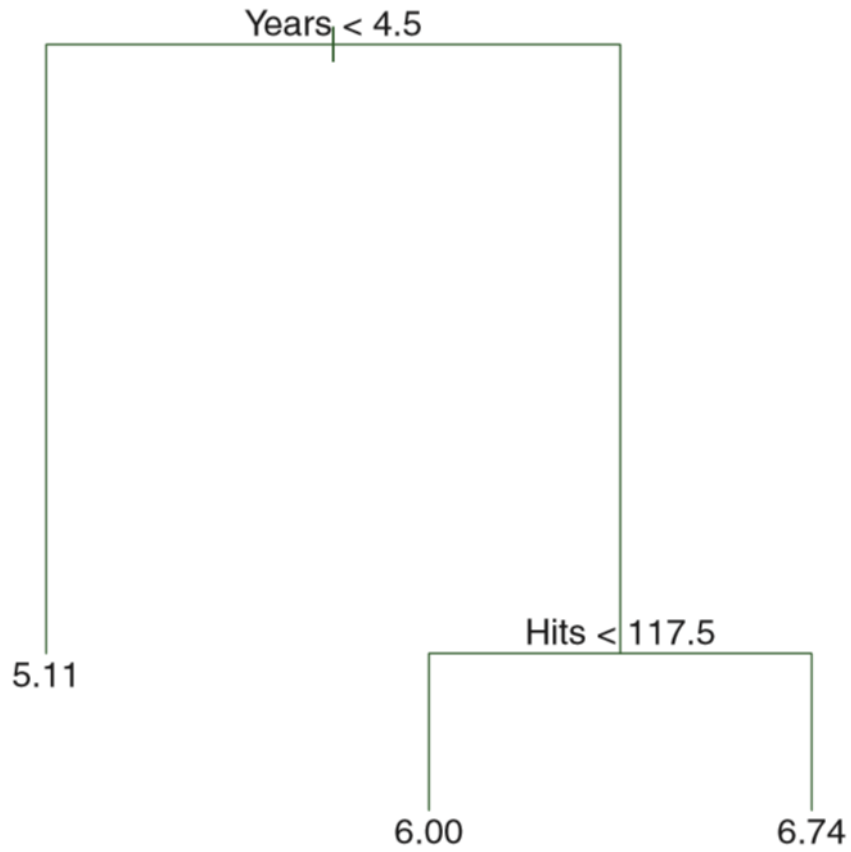
Internal node

Leaf or terminal node

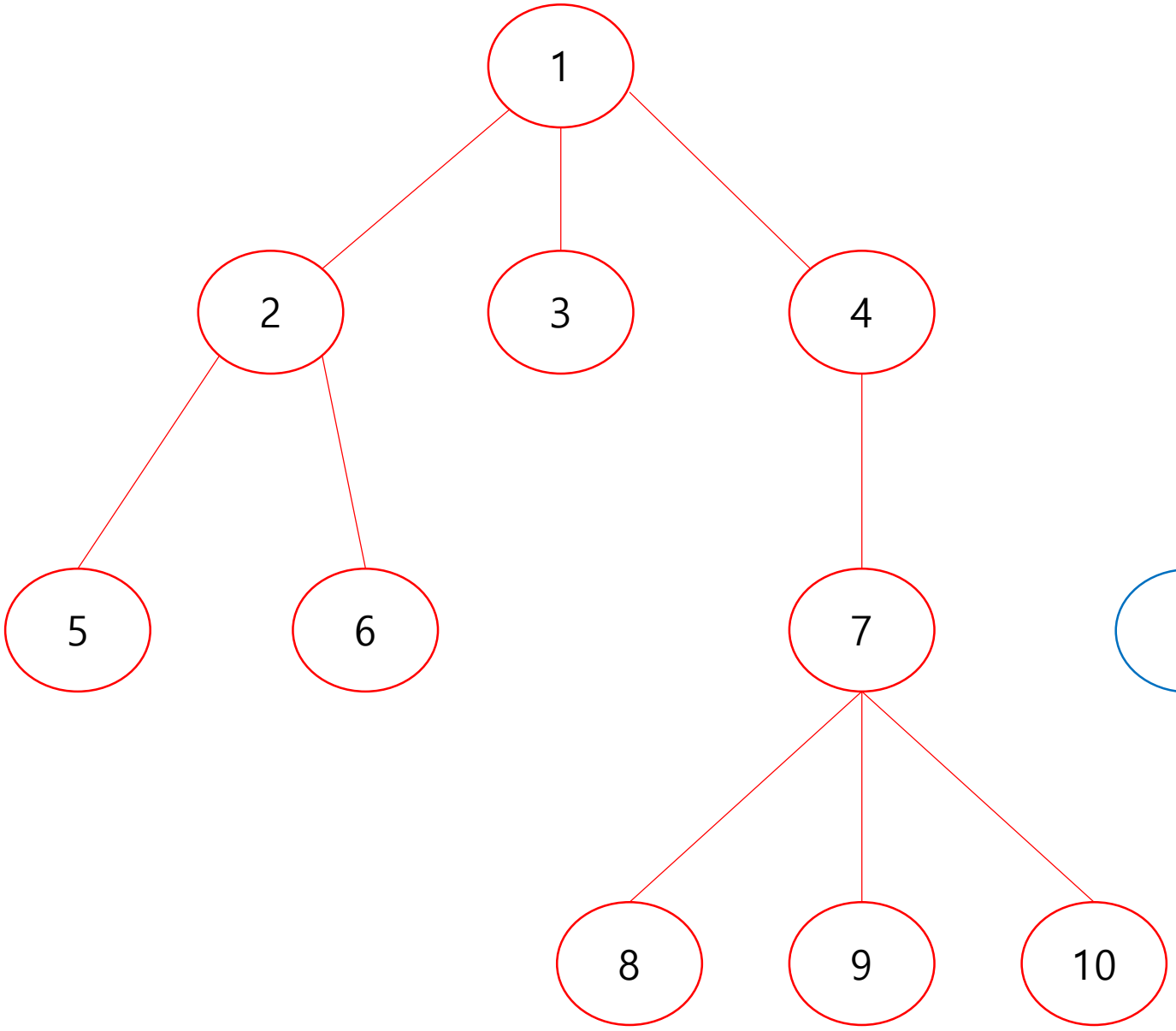
의사 결정 트리 (Decision Tree) 종류



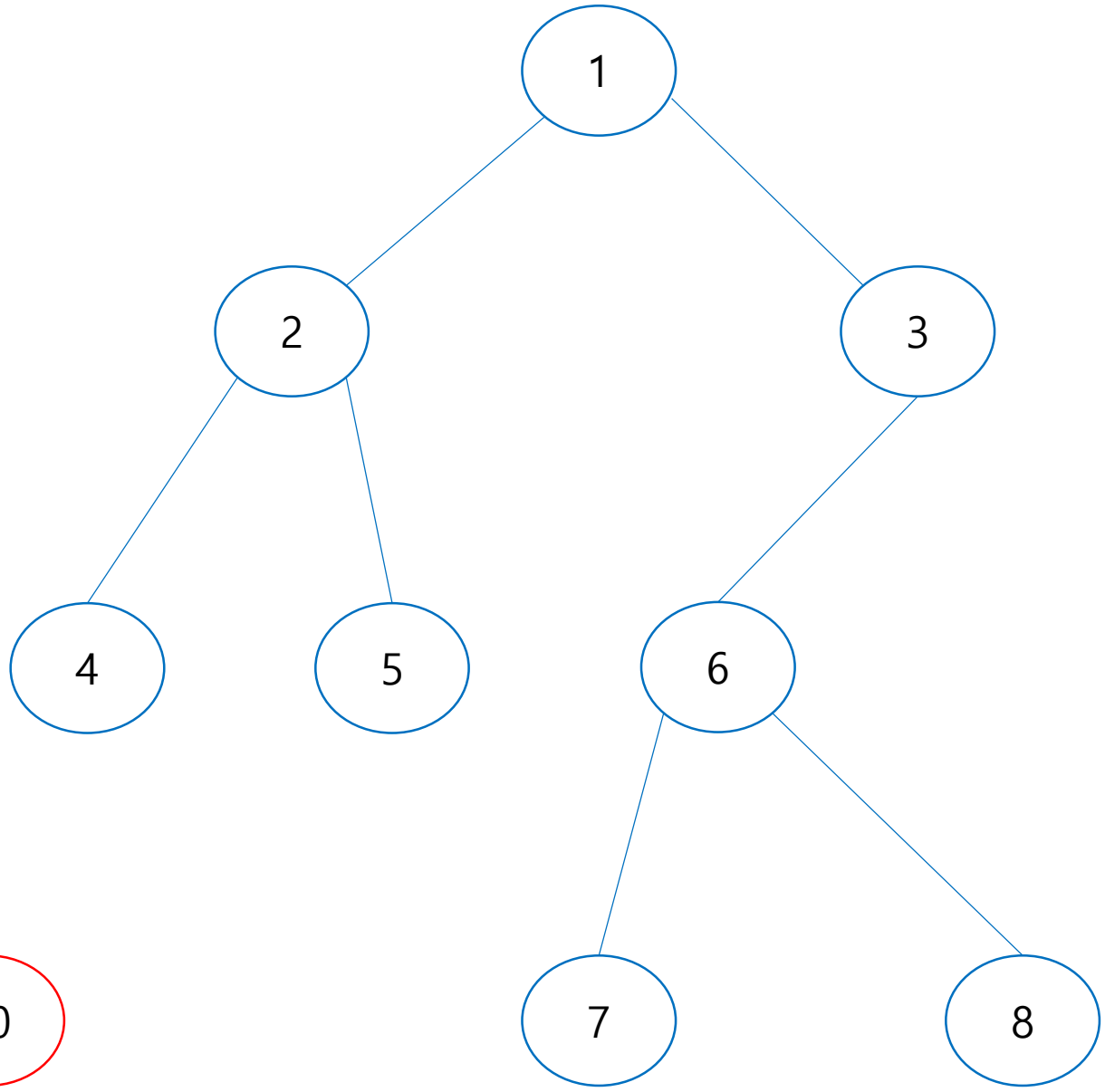
분류 트리



회귀 트리



트리

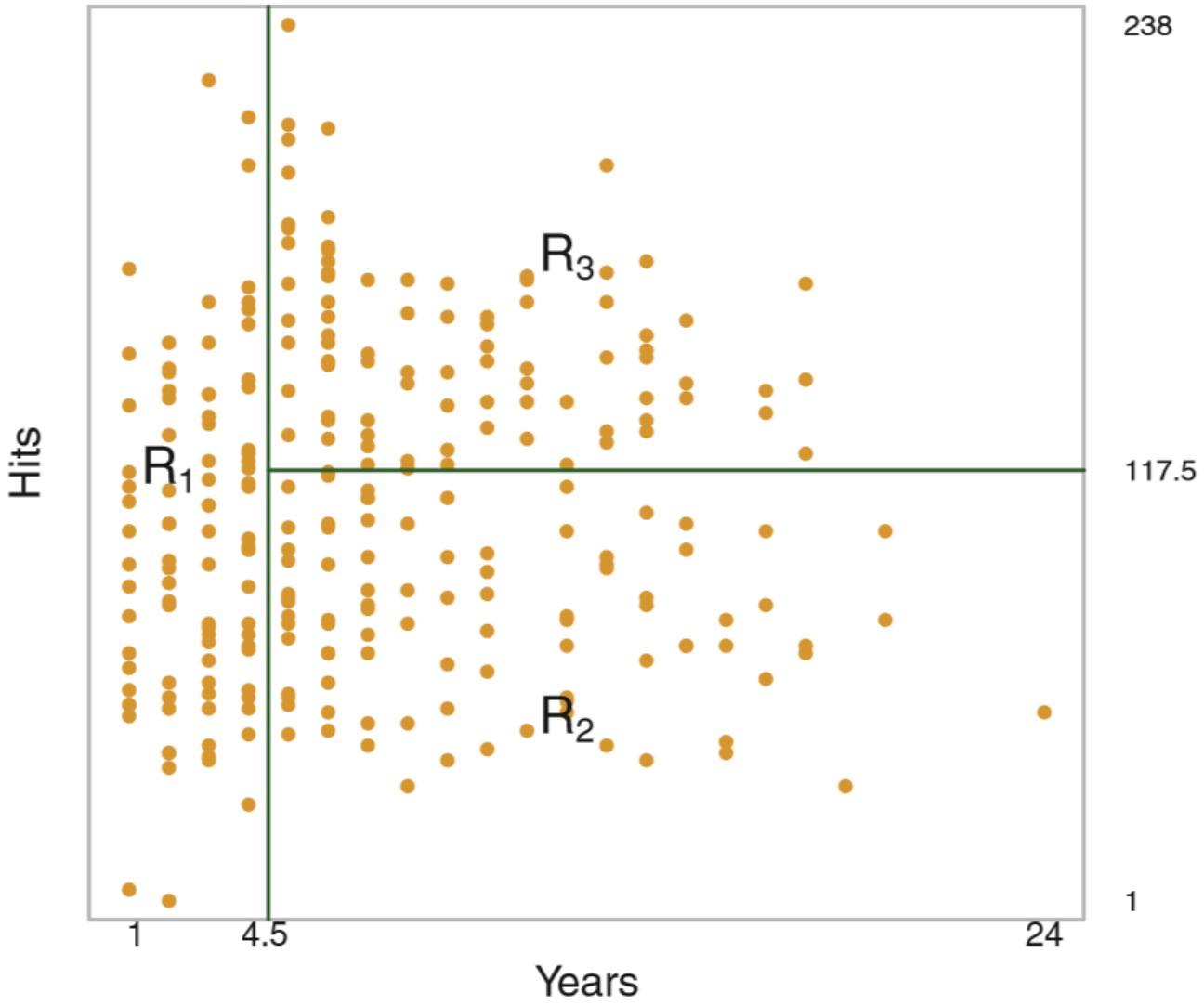


이진 트리

회귀 트리

(Regression Tree)

회귀 트리 - 결론부터



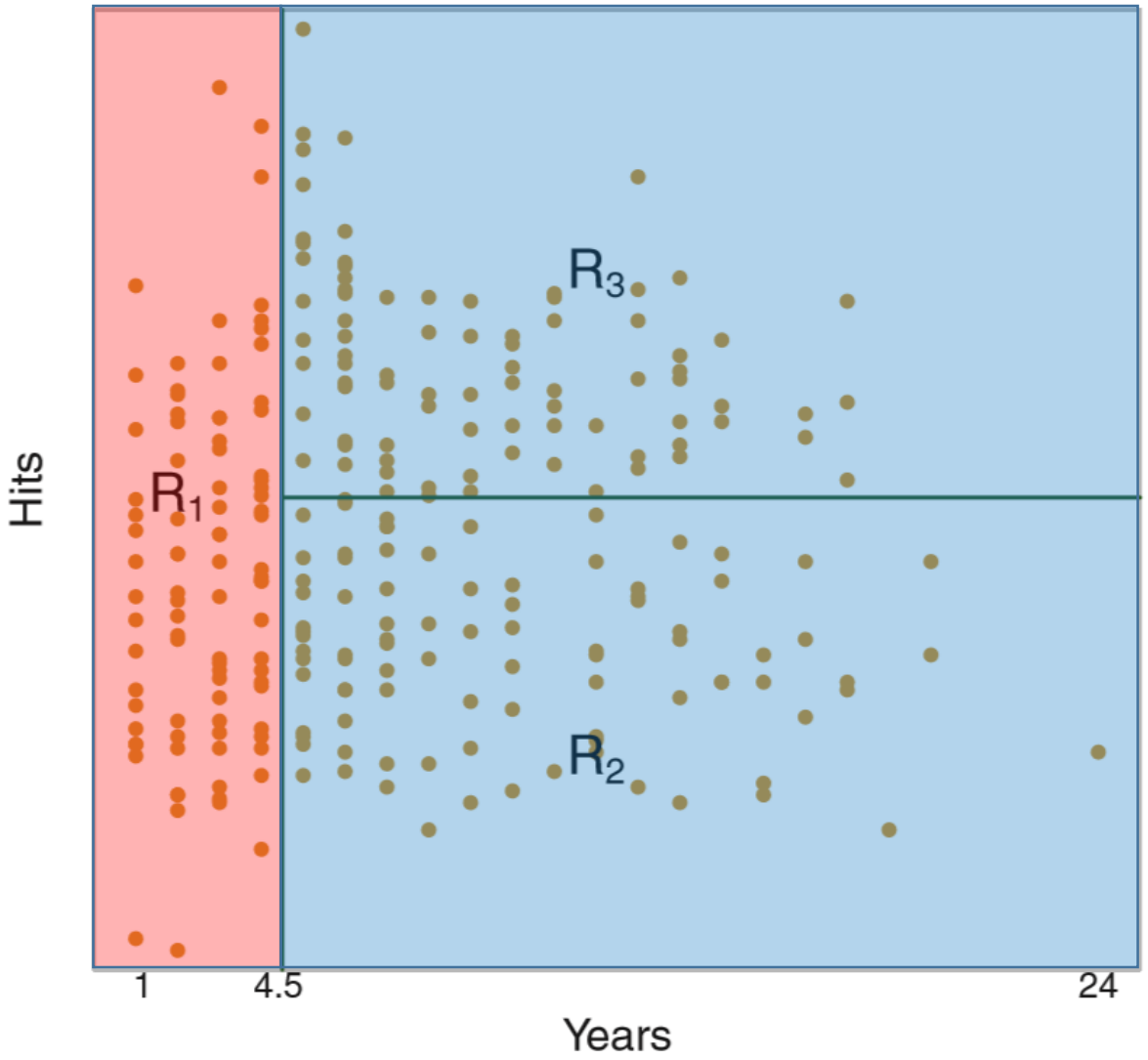
구역을 나누어 대표값(평균)을

지정하여 회귀 수행!

SW

회귀 트리 – 어떻게?

각 구역(박스)에서의 RSS값 최소로



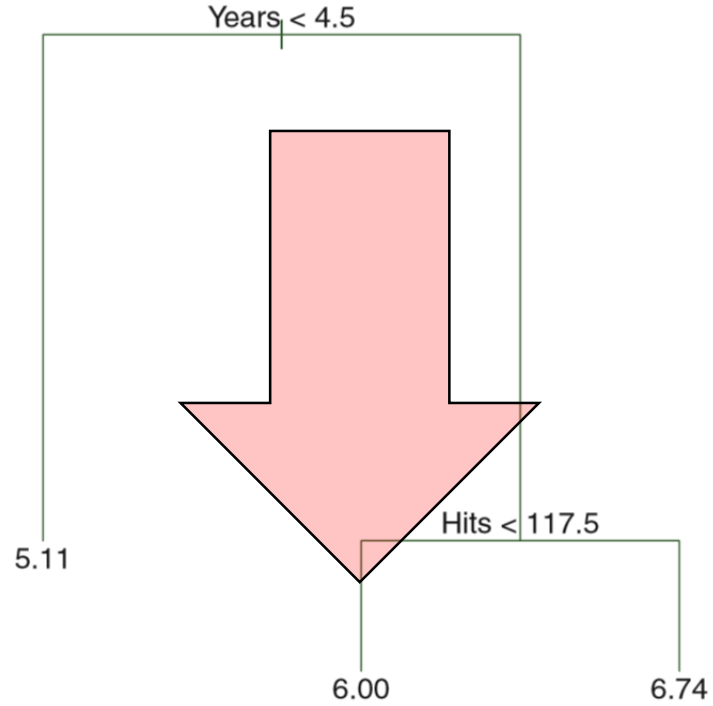
$$\Downarrow$$

$$\sum_{j=1}^{J} \sum_{i \in R_j} (y_i - \hat{y}_{R_j})^2$$

각 박스 내 개개의 관측치

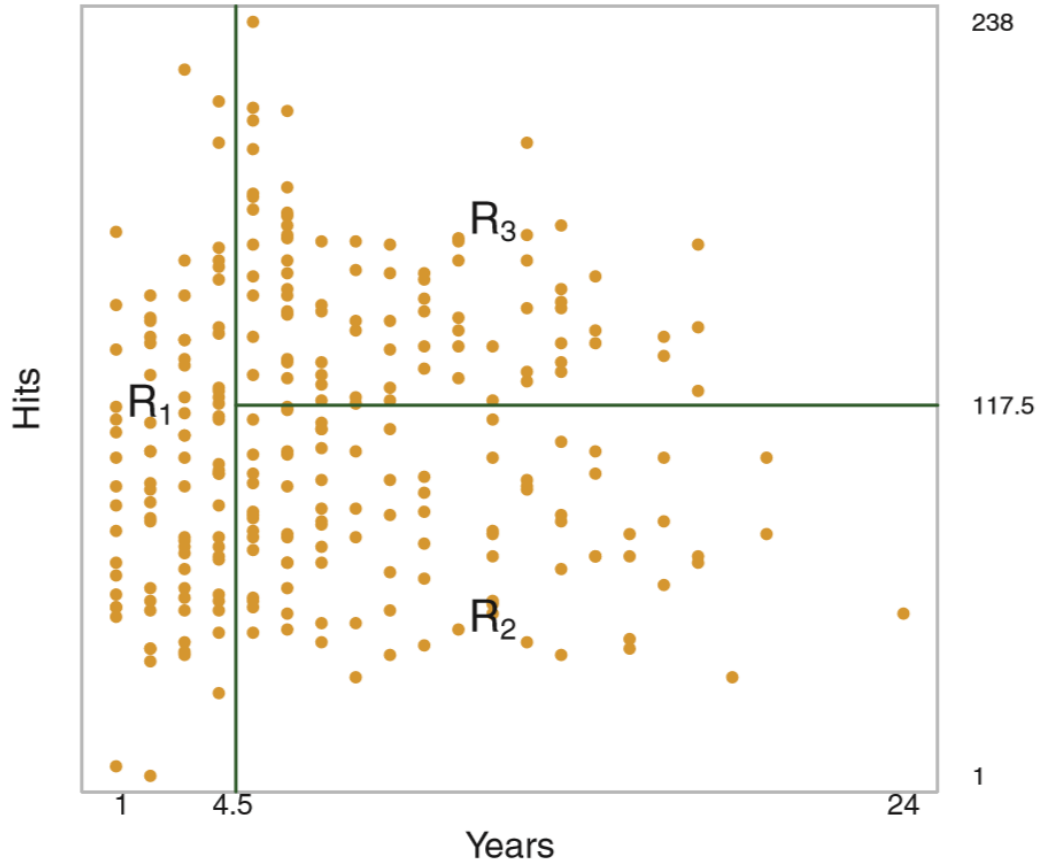
j번째 박스 내의 관측치의 평균

필요한 가정 – 재귀이진분할



Top-down(하향식) 방법

: 트리 맨 위에서 아래로 분할 하므로



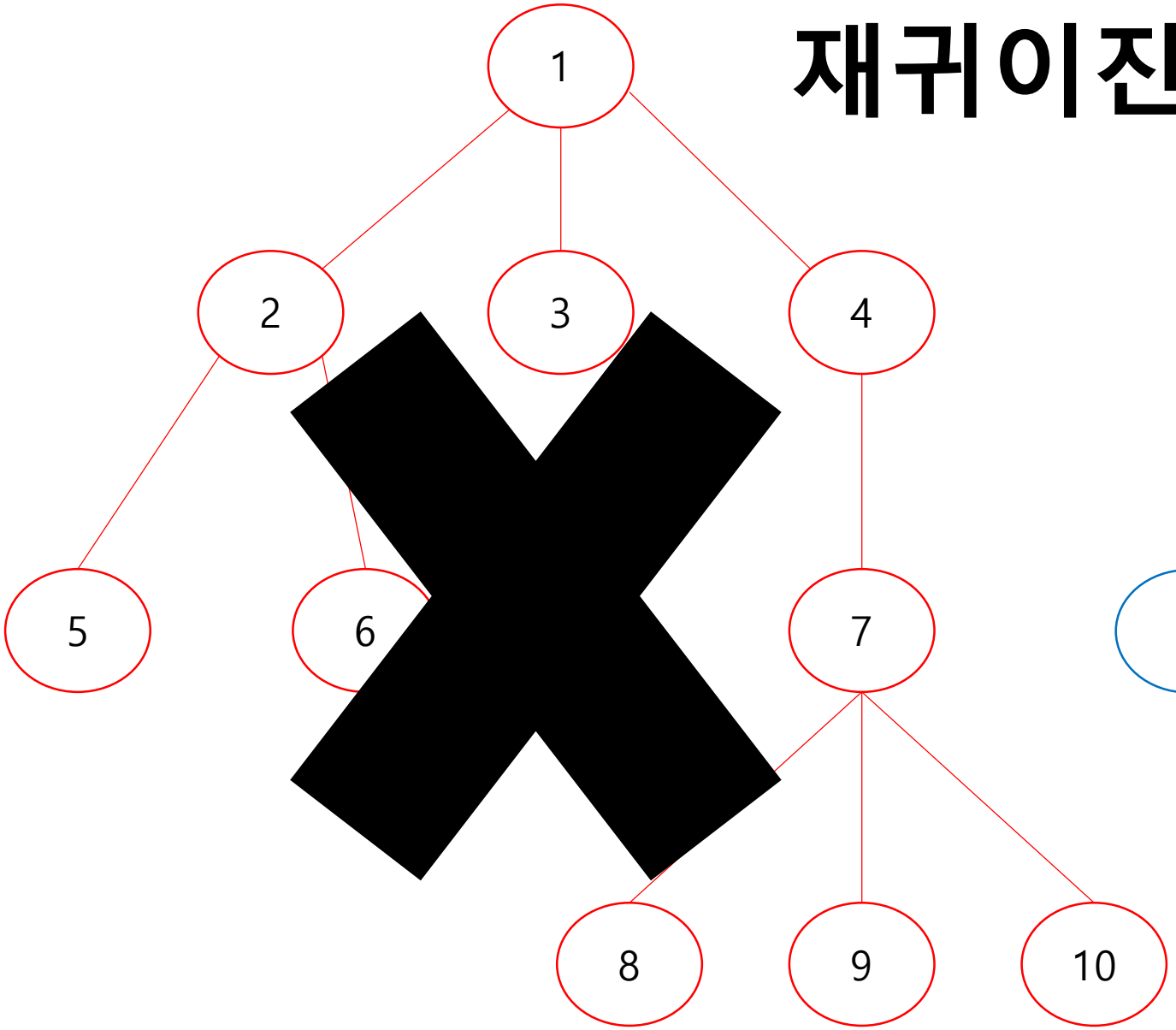
Greedy(그리디) 방법

: 각 단계에서 가장 만족할만한(탐욕적인)

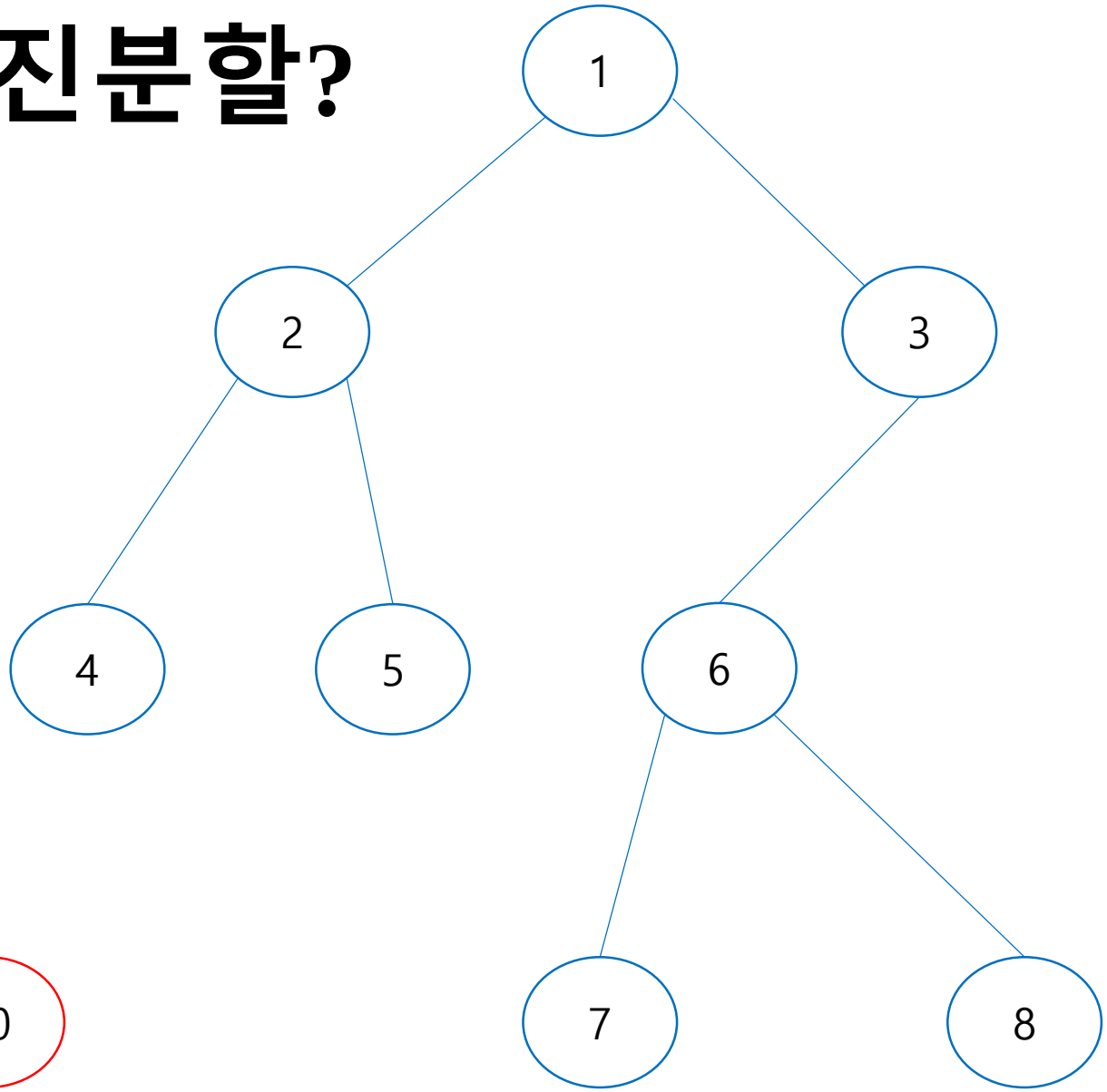
결과를 가져오므로

SW

재귀이진분할?

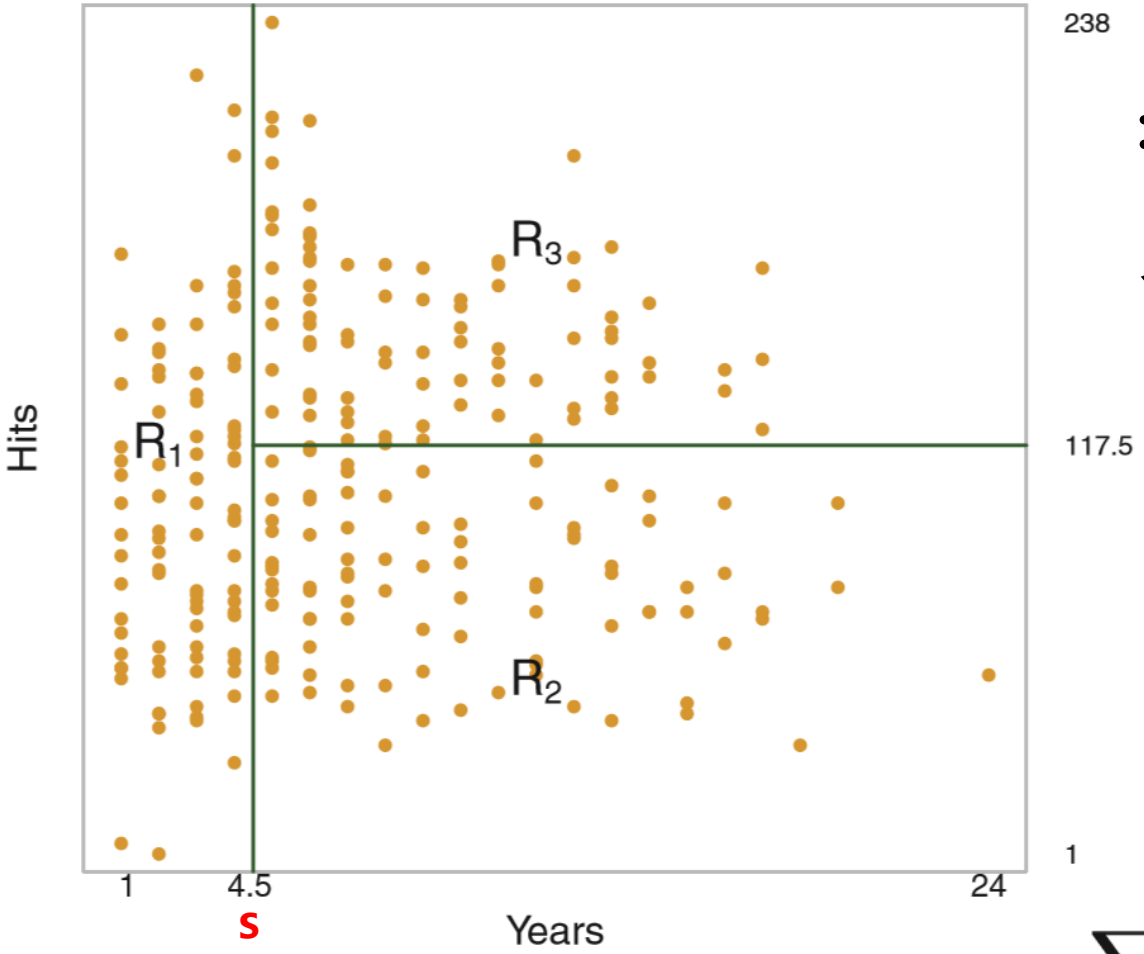


트리



이진 트리

재귀이진분할 알고리즘 Step 1



: 각 설명변수(입력)를 절단점 s로 구분하여

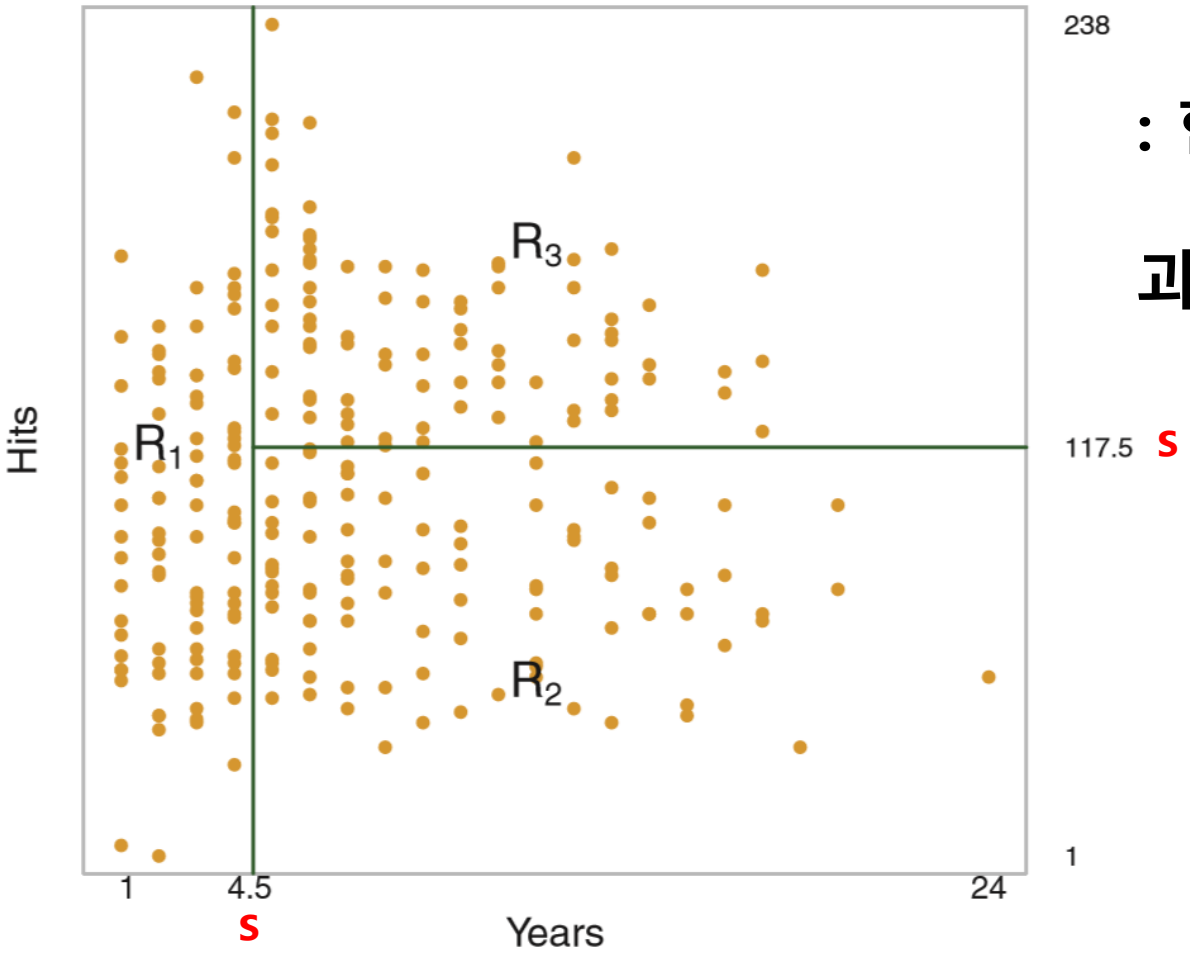
각 RSS의 합이 최소가 되는 경우를 구함!

임의의 j번째 설명변수와 절단점 s의 구역

$$R_1(j,s) = \{X | X_j < s\} \text{ and } R_2(j,s) = \{X | X_j \geq s\}$$

$$\sum_{i:\, x_i \in R_1(j,s)} (y_i - \hat{y}_{R_1})^2 + \sum_{i:\, x_i \in R_2(j,s)} (y_i - \hat{y}_{R_2})^2$$

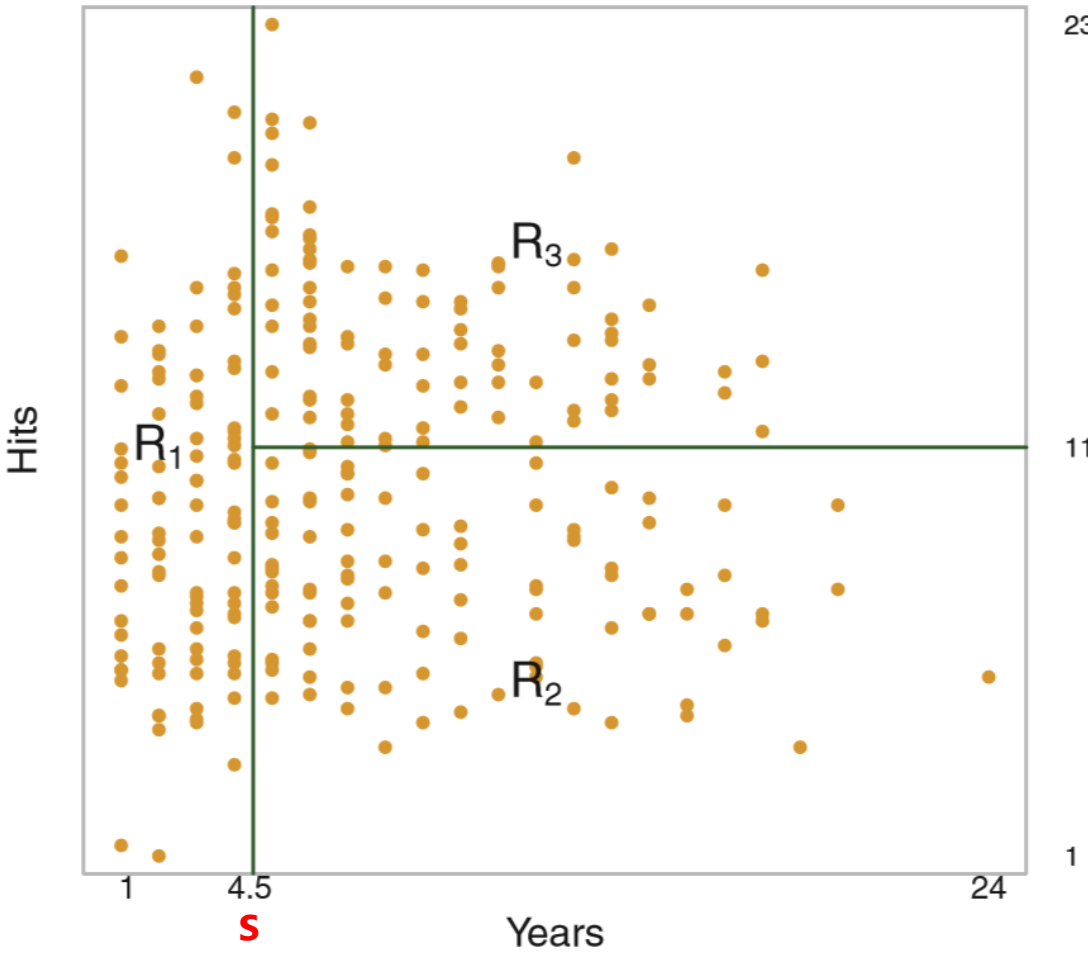
재귀이진분할 알고리즘 Step 2



: 한번 분할 이 후 나머지 구역에서 Step 1의 과정을 반복한다.

s

재귀이진분할 알고리즘 Step 3



: 어떤 정지기준이 만족될 때 까지 계속 한다.

예) 1. 관측치가 100개 이하인 경우,

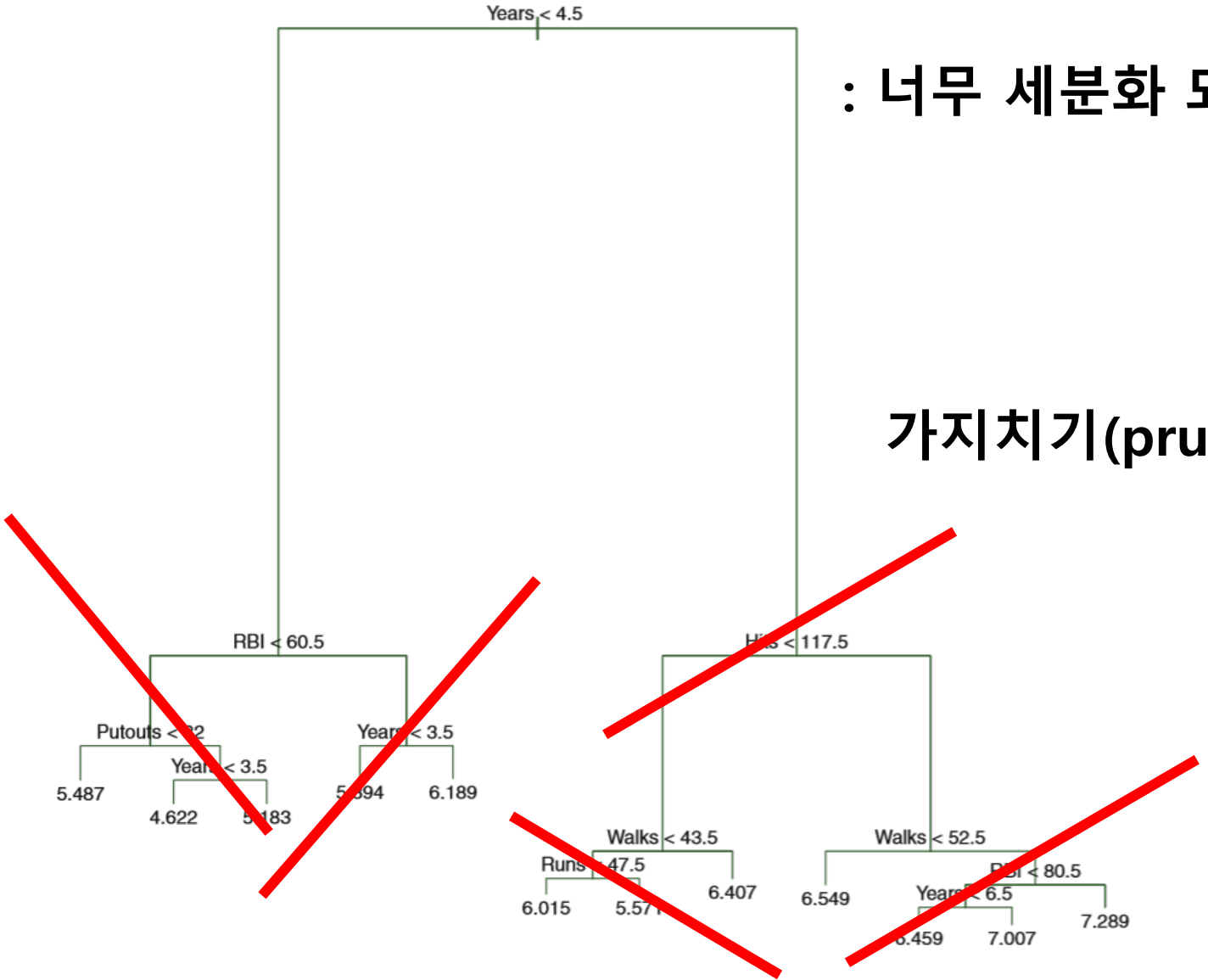
2. 불순도 또는 RSS가 0인 경우,

3. 불순도 또는 RSS의 감소량이 임계값 이하인 경우, 그만!

지금까지의 순서로 만들어 진 것이 큰 트리 T0에 해당!

SW

재귀이진분할 문제점



: 너무 세분화 되어 Overfitting(과적합) 문제 발생!

가지치기(pruning)를 통한 서브트리 만들기!

가치치기의 종류

사전 가지치기

앞에서 말한 정지조건을 완화하여

극단이 아니라도 분할을 그만하도록

하여 트리의 크기를 적게 분할 한다.

→ 수평선효과(이후 더 좋은 분할이 나와도

이를 볼 수 없다는 단점이 있다)

사후 가지치기

극단적인 조건으로 매우 큰 트리 T0를

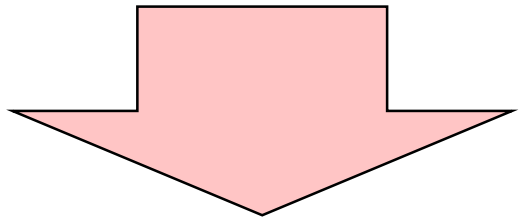
먼저 만든 후에 만들어진 가지들을 쳐내어

가지치기를 한다.

사후 가지치기는 어떻게?

서브트리 T의 터미널 노드 수

$$\sum_{m=1}^{|T|} \sum_{i:\ x_i \in R_m} (y_i - \hat{y}_{R_m})^2 + \alpha|T|$$



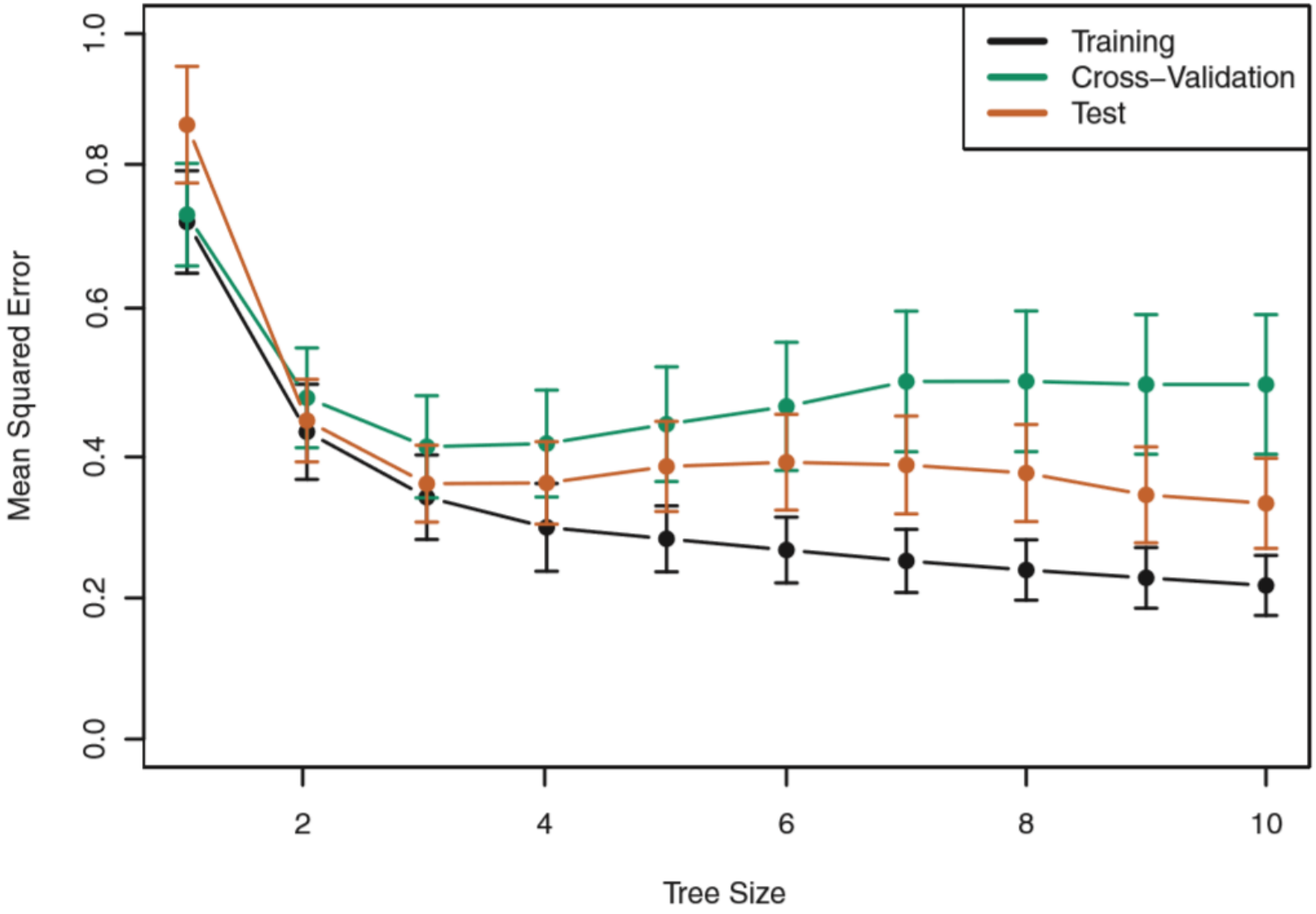
조율파라미터로써 서브트리의 복잡도와 훈련자료의 적합도 사이의 Trade-off 제어

마지막 단계

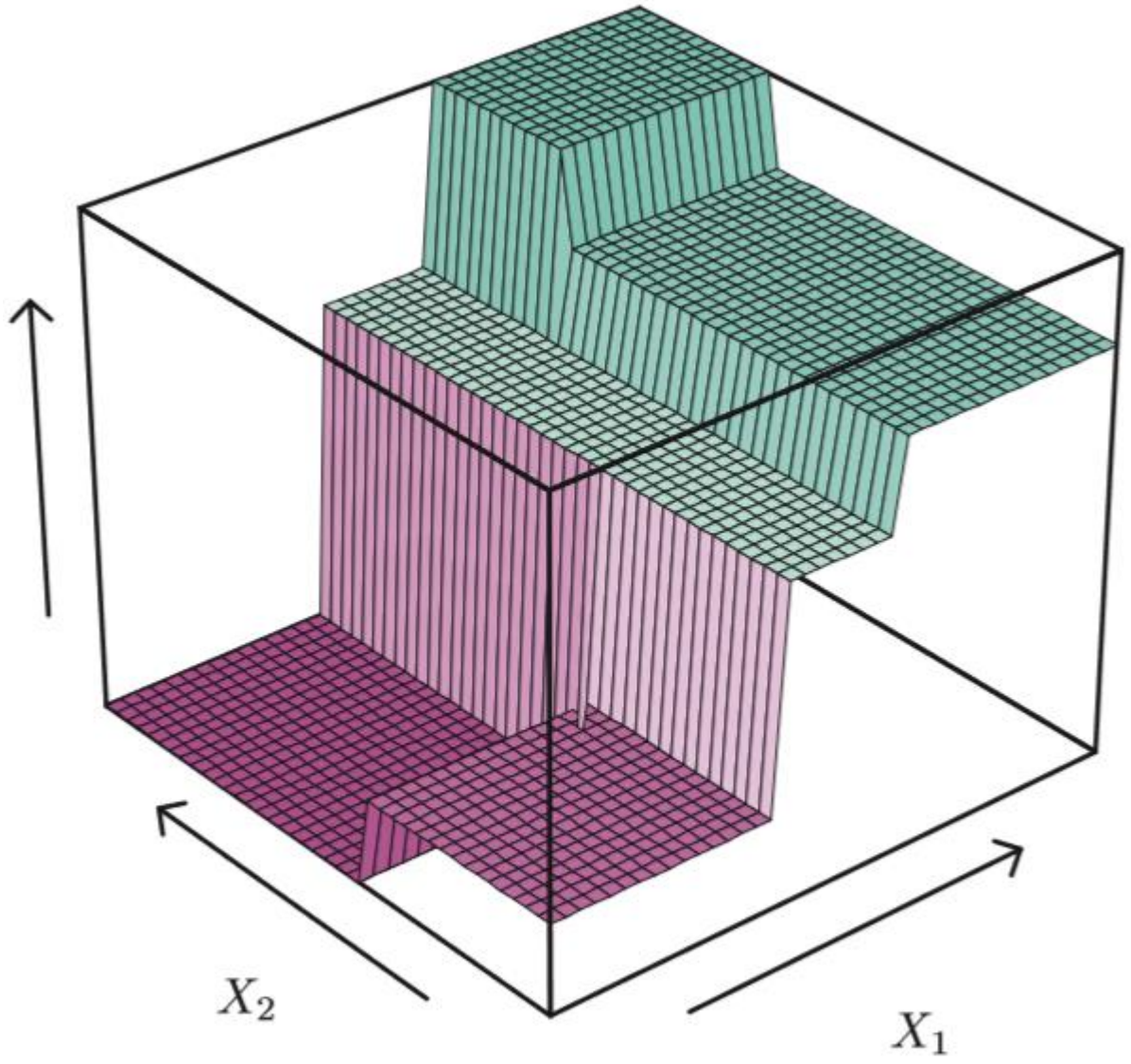
: Cross-Validation 으로

α 결정하여

최종 서브트리 선정!



회귀 트리 결과 예시



분류 트리

(Classification Tree)

대표적인 결정 트리 계통도

L. Breiman

CART (Classification And Regression Tree)

R. Quinlan

ID3 (Interative Dichotomizer 3)

기능 확장

C4.5

상용화

C5.0(Unix/Linux 버전)과 See5(Window 버전)

대표적인 결정 트리 계통도

특성	CART	ID3	C4.5
실수 데이터	부등호 질문	등식 질문	부등호 질문
트리 형태	이진 트리	트리	트리
가지치기	잎 노드 병합	X	규칙 집합
분류	지원	지원	지원
회귀	지원	X	X
손실특징	대리 분기	X	샘플 무시
다중 변수 질문	지원	X	X

분류 트리

구역 내 **평균**을 대표값으로!

RSS를 기준으로 분할

구역 내 **최빈값**을 대표값으로!

분류오류율 (또는 엔트로피, 지니지수, Infromation Gain 등)을 기준으로 분할

새로운 기준들

m번째 영역

Yes, No, Yes, No, No	Yes, Yes, Yes, No, No, Yes, No, Yes

분류오류율

$$E = 1 - \max_k(\hat{p}_{mk})$$

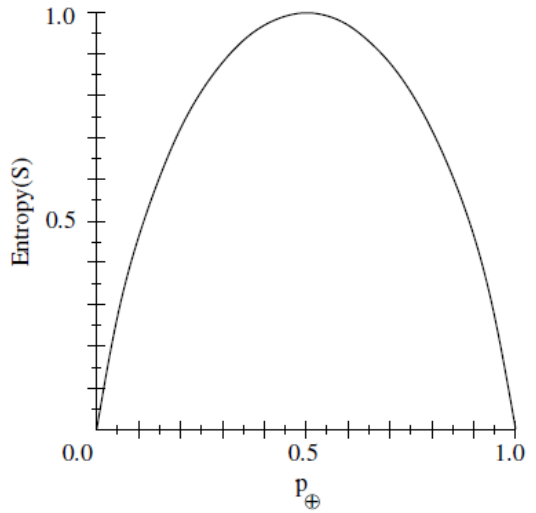
m번째 영역 내 k 클래스의 비율

지니 지수

$$G = \sum_{k=1}^{K} \hat{p}_{mk}(1 - \hat{p}_{mk})$$

교차 엔트로피

$$D = -\sum_{k=1}^{K} \hat{p}_{mk} \log \hat{p}_{mk}$$



SW

분류 트리 예: ID3 (Interative Dichotomizer 3)

Information Gain

The information gain of an attribute A relative to a collection of examples S is defined as

$$Gain(S, A) \equiv Entropy(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} Entropy(S_v)$$

where $Values(A)$ is the set of all possible values for attribute A, and S_v is the subset of S for which attribute A has value v

1. 원본 데이터의 엔트로피 구하기

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

Boolean classification에서 교차 엔트로피

$$Entropy(S) \equiv -p_{\oplus} \log_2 p_{\oplus} - p_{\ominus} \log_2 p_{\ominus}$$

- **Example**

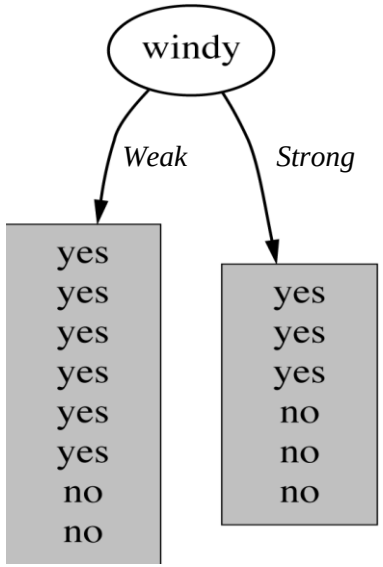
$$Entropy([9+, 5-])$$

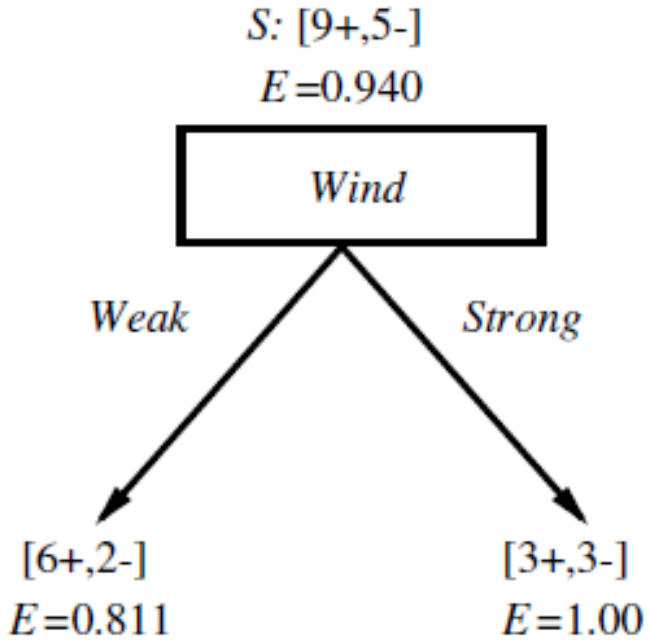
$$= -\left(\frac{9}{14}\right)\log_2\left(\frac{9}{14}\right) - \left(\frac{5}{14}\right)\log_2\left(\frac{5}{14}\right)$$

$$= 0.940$$

2. 가장 좋은 분류 특징은?

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No





$$Values\ (Wind) = Weak, Strong$$
$$S = [9+, 5-]$$
$$S_{Weak} \leftarrow [6+, 2-]$$
$$S_{Strong} \leftarrow [3+, 3-]$$

$$Entropy(S_{Weak}) = -\left(\frac{6}{8}\right)\log_2\left(\frac{6}{8}\right) - \left(\frac{2}{8}\right)\log_2\left(\frac{2}{8}\right)$$
$$= \mathbf{0.811}$$

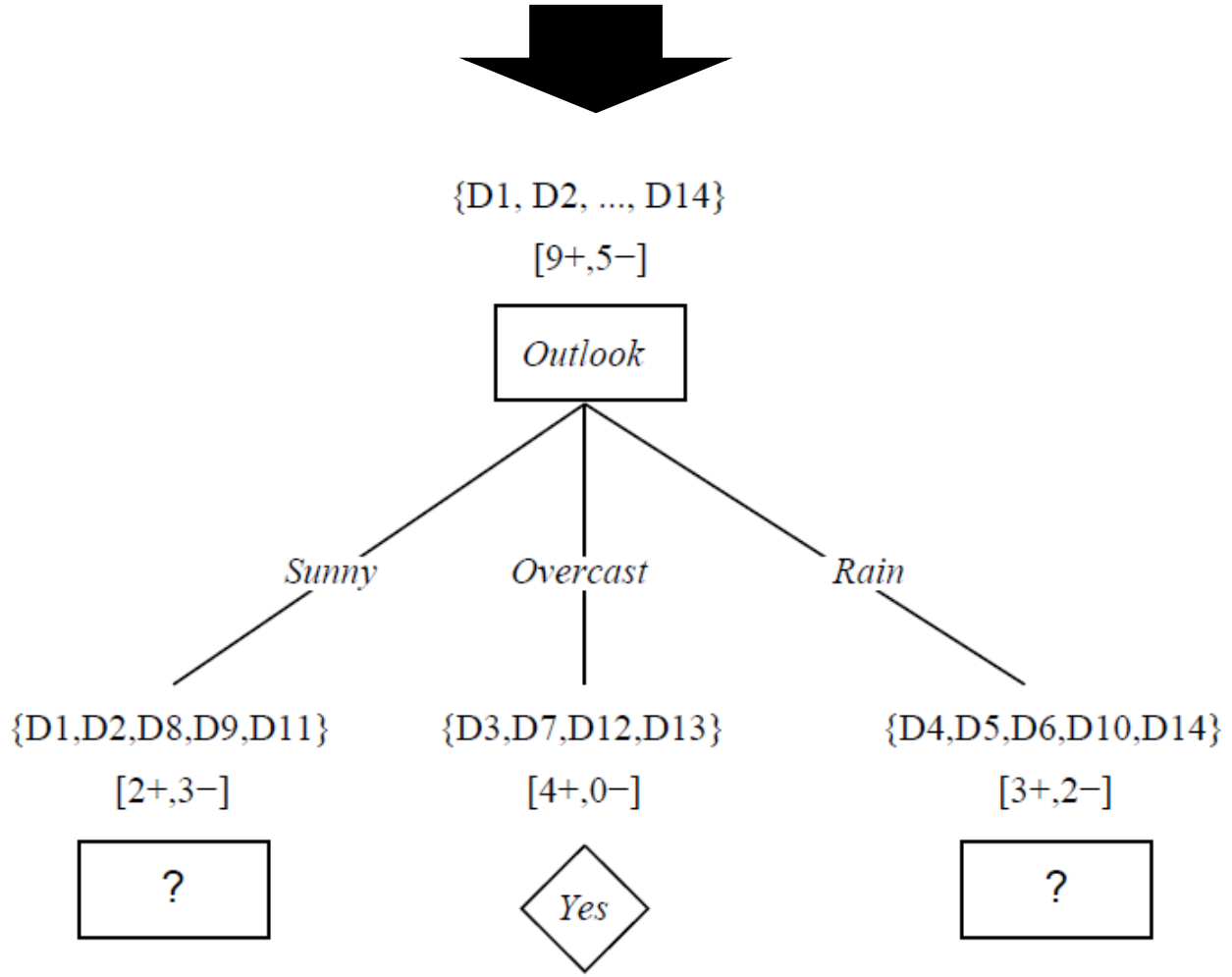
$$Entropy(S_{Strong}) = -\left(\frac{3}{6}\right)\log_2\left(\frac{3}{6}\right) - \left(\frac{3}{6}\right)\log_2\left(\frac{3}{6}\right)$$
$$= \mathbf{1.00}$$

$$Gain(S, Wind) = Entropy(S) - \sum_{v \in \{Weak,\ Strong\}} \frac{|S_v|}{|S|} Entropy(S_v)$$
$$= Entropy(S) - \left(\frac{8}{14}\right) Entropy(S_{Weak})$$
$$- \left(\frac{6}{14}\right) Entropy(S_{Strong})$$
$$= 0.940 - \left(\frac{8}{14}\right) 0.811 - \left(\frac{6}{14}\right) 1.00$$
$$= 0.048$$

3. Root node 선정 후, 반복!

$Gain(S, Outlook) = 0.246$
$Gain(S, Humidity) = 0.151$
$Gain(S, Wind) = 0.048$
$Gain(S, Temperature) = 0.029$

Information Gain이 가장 큰 Outlook을 Root node로 선정 후, 트리 분할



언제까지?

모든 특징들이 이미 Tree에 등장했거나,

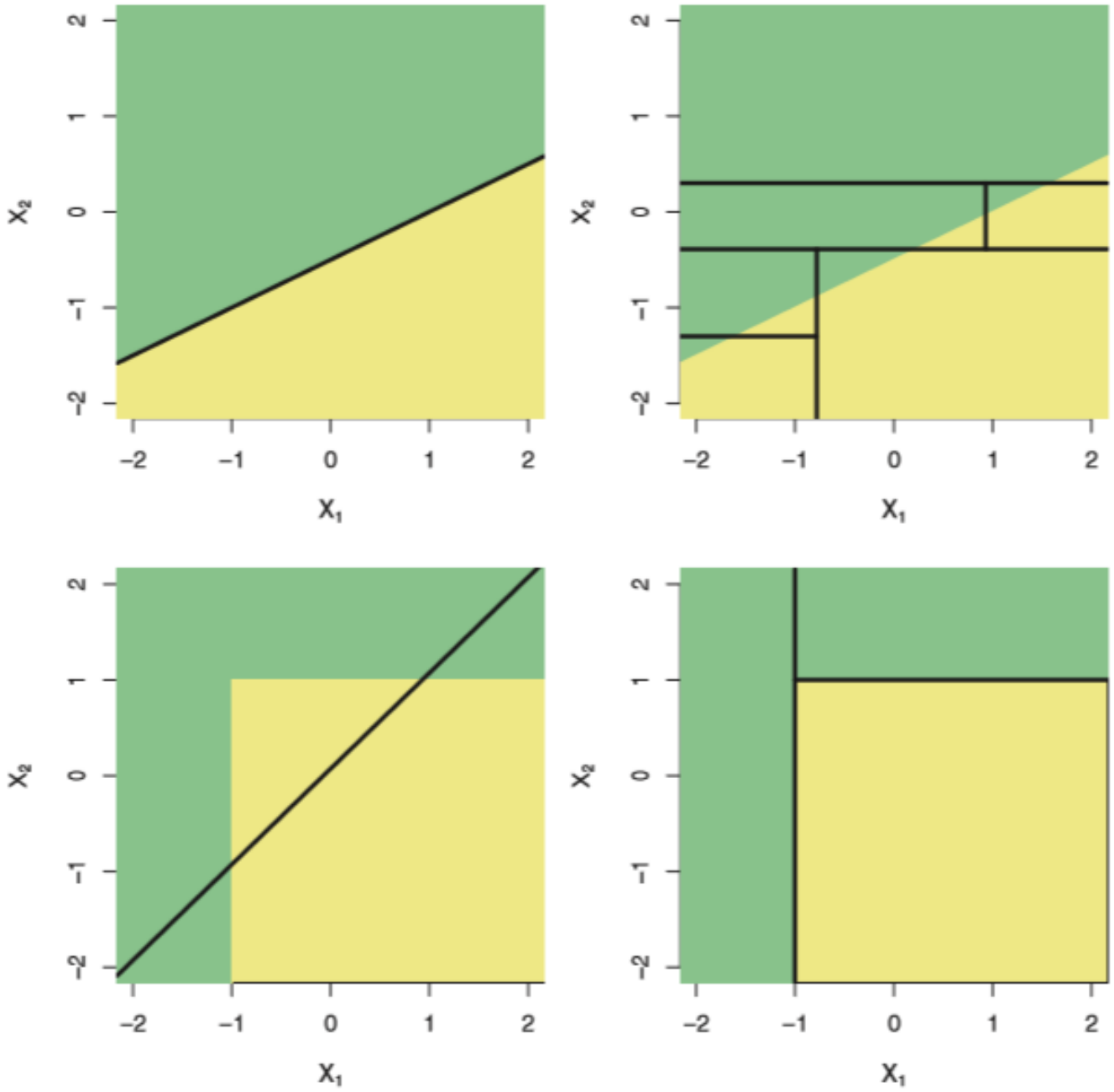
Leaf Node에서의 관측치들의 Entropy가 0이

될 때까지!

트리 요약 정리

1. 선형 모델과 비교하여 비선형적인

형태의 데이터에 더욱 잘 적합!



트리 요약 정리

2. 설명하기가 쉽다! (그래픽으로 나타내기 쉽고, 비전문가도 쉽게 해석 가능)

3. 가변수들을 만들지 않고 질적 설명변수들을 쉽게 처리 가능

4. 다른 분류기법들과 동일한 수준의 예측 정확도 제공하지 못함! (Overfitting 문제)

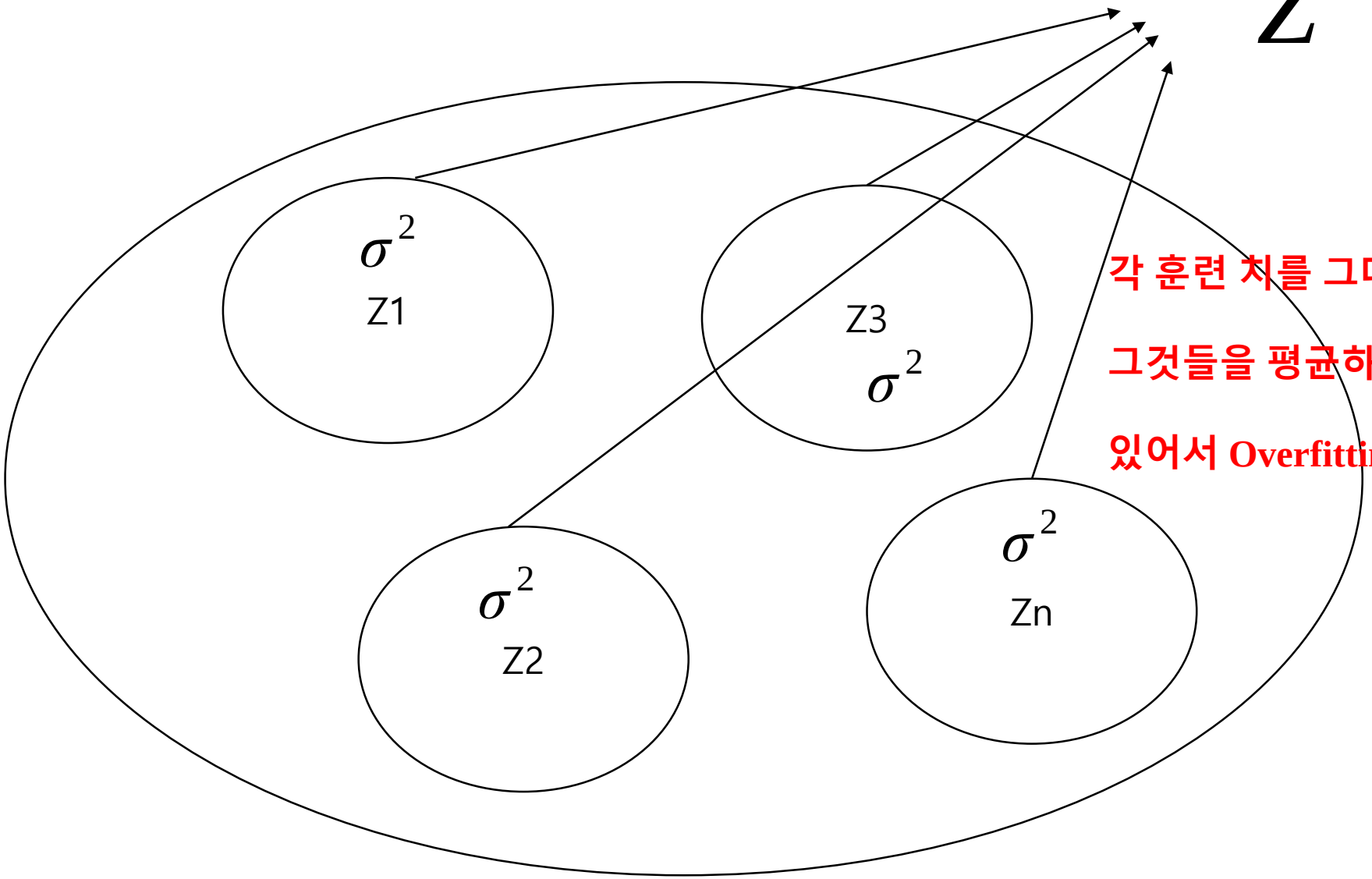
→ 따라서, 이를 개선하기 위해 배깅, 랜덤 포레스트, 부스팅이 필요!

배깅

(Bagging)

배깅, 왜 효과가 있나?

$$\bar{Z} \quad \frac{\sigma^2}{n}$$

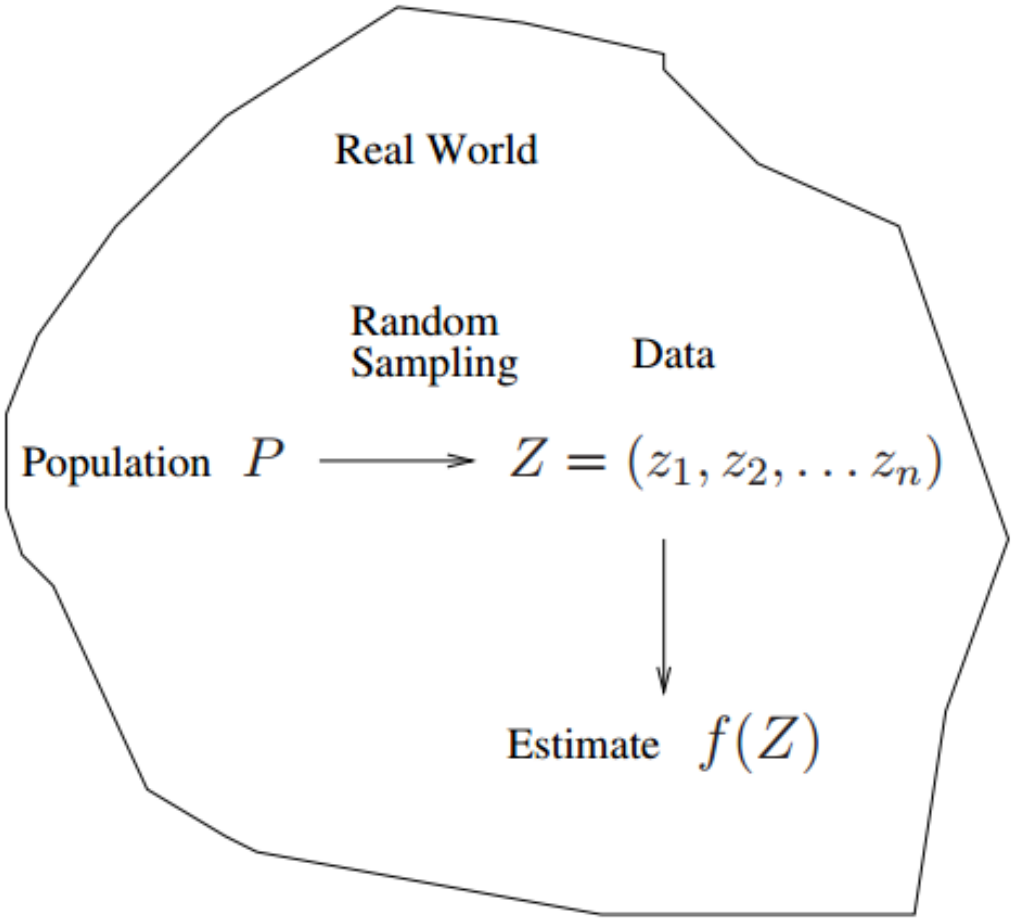


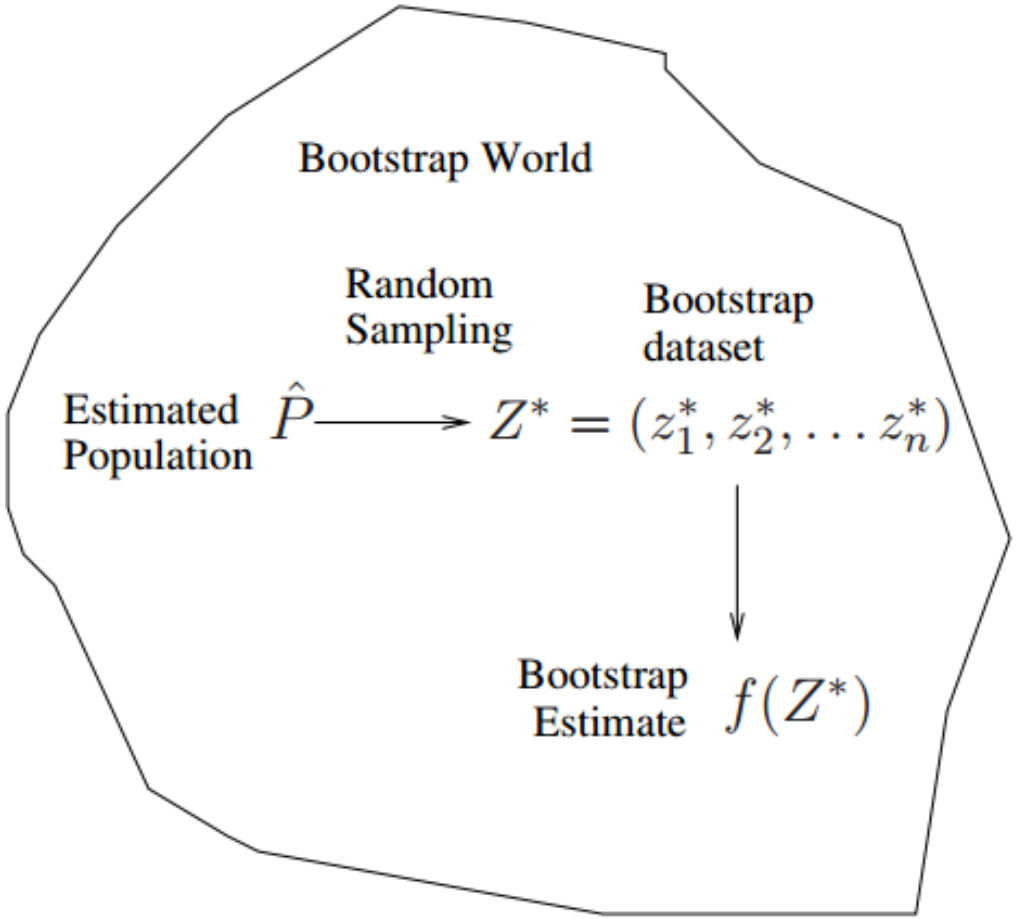
각 훈련 치를 그대로 사용 하는 것이 아니라

그것들을 평균하면 분산을 떨어뜨리는 효과가

있어서 Overfitting을 막아준다.

Bootstrap

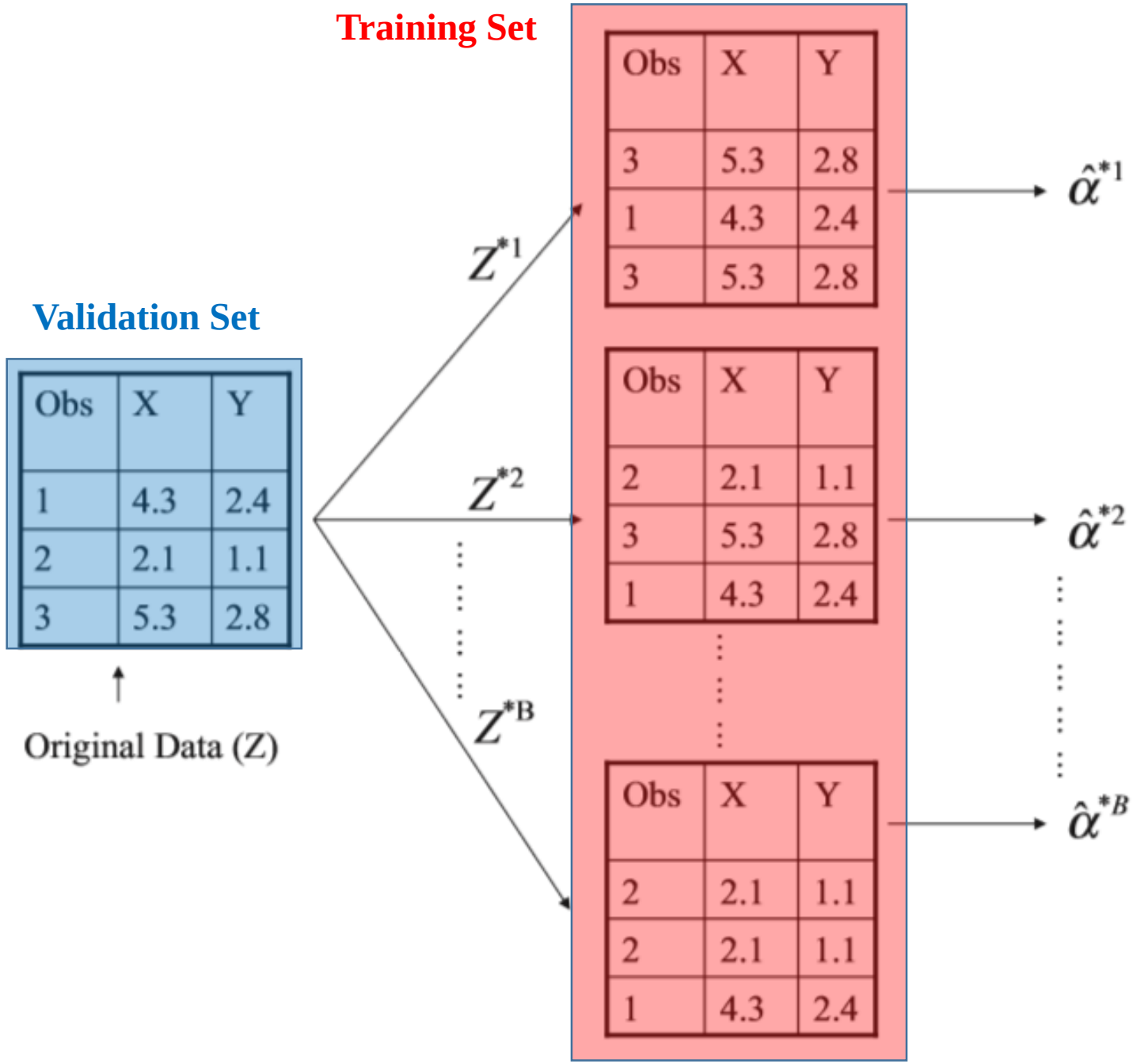

Real World
Random Sampling
Data
Population P
$Z = (z_1, z_2, \ldots z_n)$
Estimate $f(Z)$


Bootstrap World
Random Sampling
Bootstrap dataset
Estimated Population $\hat{P}$
$Z^* = (z_1^*, z_2^*, \ldots z_n^*)$
Bootstrap Estimate $f(Z^*)$

Bootstrap

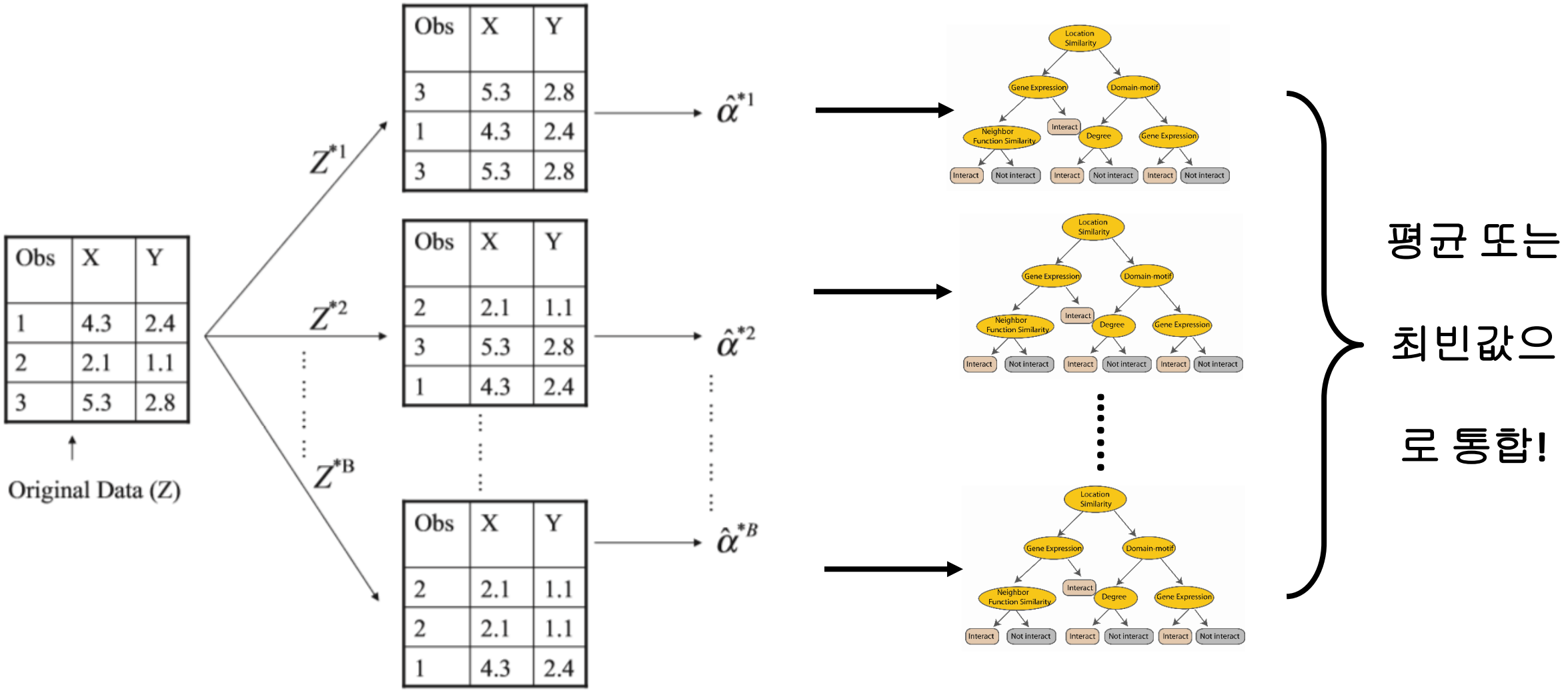
반복을 허용하고,

관찰 값에서 Resampling 한다!

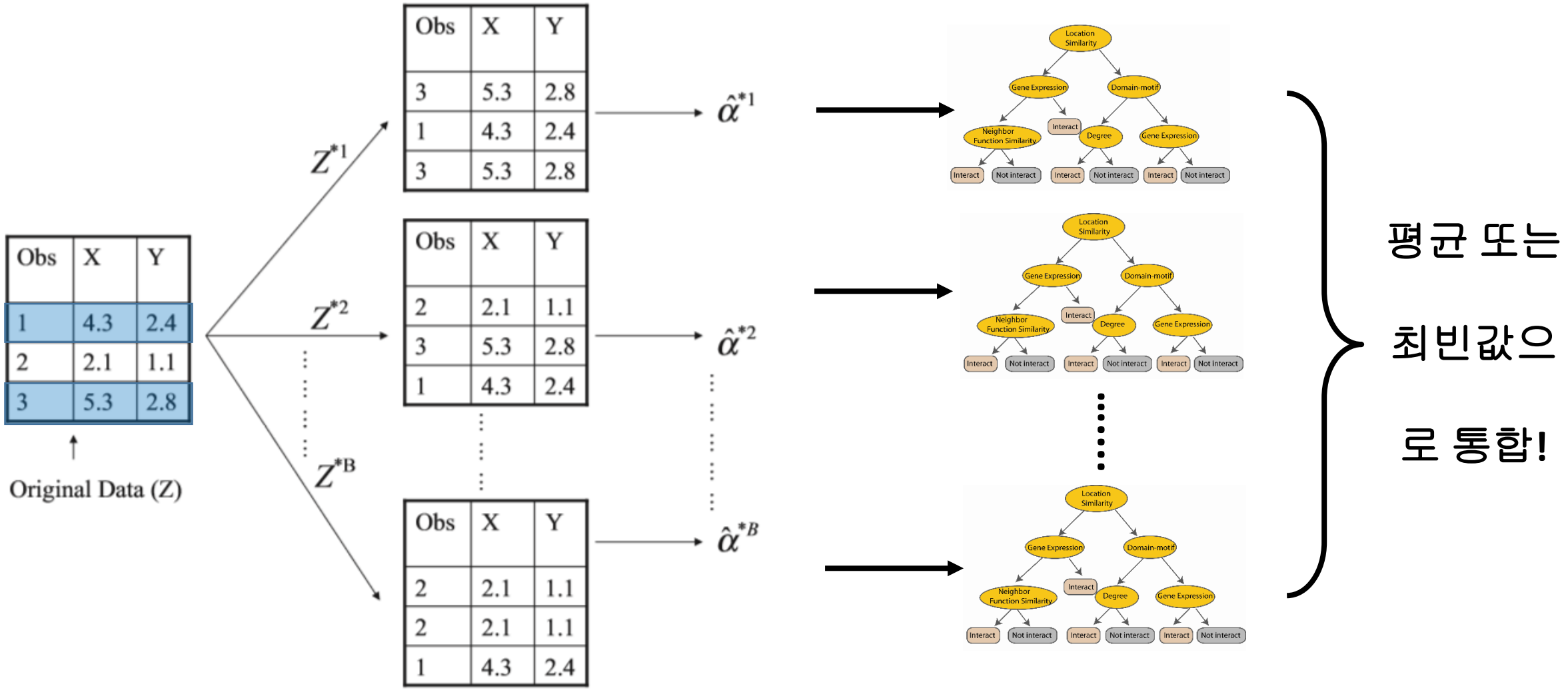


SW

배깅(Bagging) = Boostrap + Aggregating



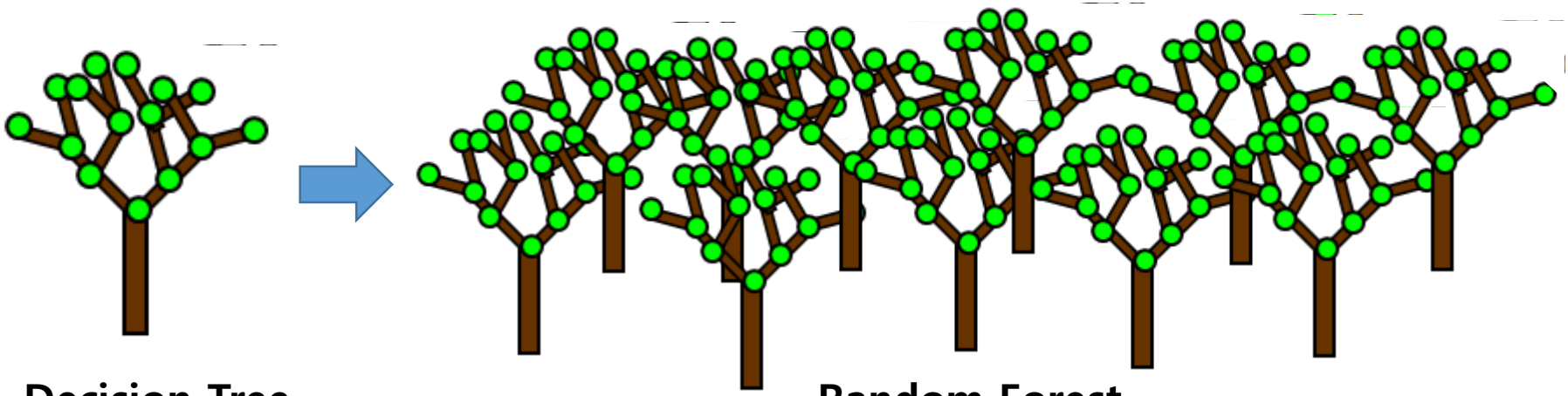
OOB(Out Of Bag) 오차로 검증!



랜덤 포레스트

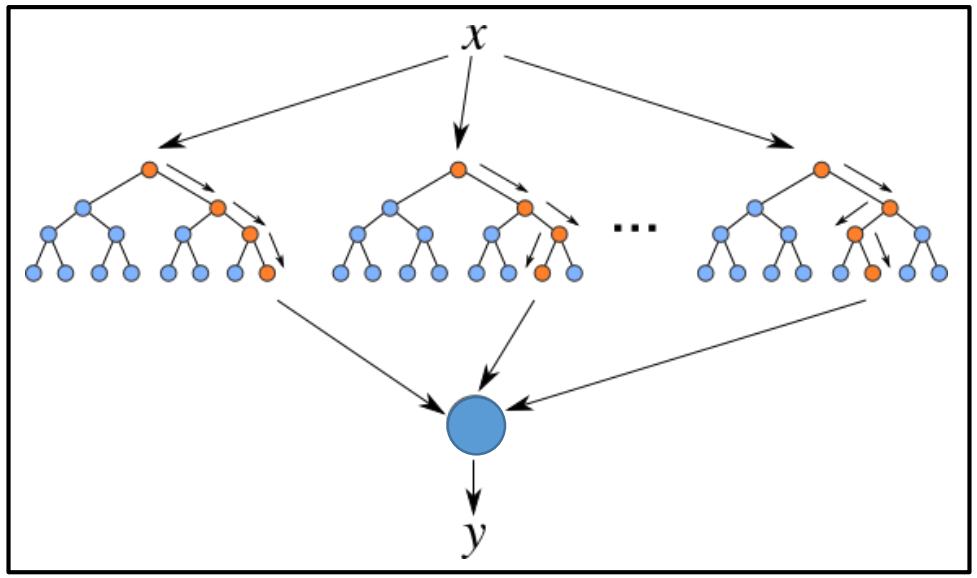
(Random Forest)

랜덤 포레스트



Decision Tree

Random Forest



특징을 다 쓰는게 아니라 몇 개만 추출하여 배깅

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

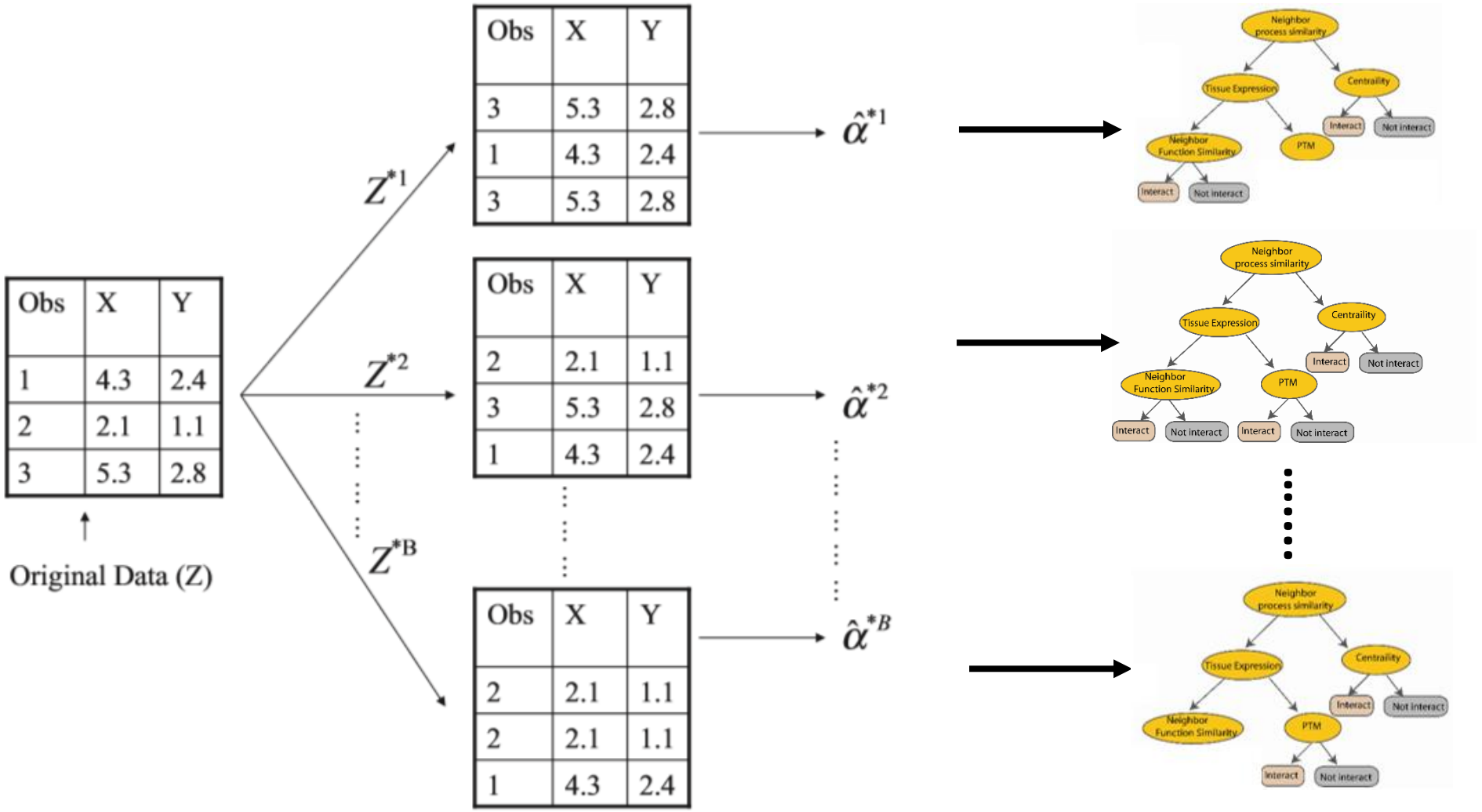
샘플에 사용할 특징 수 m = p (총 특징 수)

→ 배깅

m < p 예를 들어, $m \approx \sqrt{p}$

→ 랜덤 포레스트

랜덤 포레스트


Obs X Y
1 4.3 2.4
2 2.1 1.1
3 5.3 2.8
Original Data (Z)
Z^{*1}
Z^{*2}
Z^{*B}
Obs X Y
3 5.3 2.8
1 4.3 2.4
3 5.3 2.8
Obs X Y
2 2.1 1.1
3 5.3 2.8
1 4.3 2.4
Obs X Y
2 2.1 1.1
2 2.1 1.1
1 4.3 2.4
$\hat{\alpha}^{*1}$
$\hat{\alpha}^{*2}$
$\hat{\alpha}^{*B}$
Neighbor process similarity
Tissue Expression
Centrality
Neighbor Function Similarity
PTM
Interact
Not interact
평균 또는
최빈값으
로 통합!

정리

Tree-Based Methods

Supervised Learning

- **Decision Tree (의사결정트리) – 회귀/분류 트리**
- **+ 배깅, 랜덤 포레스트, 부스팅**

Unsupervised Learning

- **Clustering (군집화)**

Thank you!