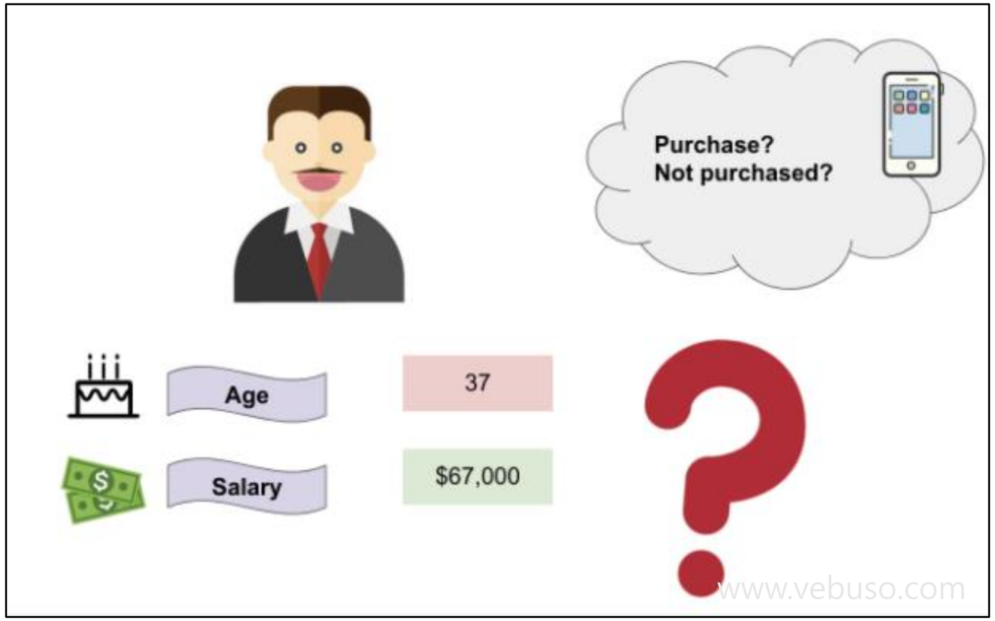
3. Supervised Learning

금융 데이터마이닝
2022 Summer
김아현

지도학습(Supervised Learing) 알고리즘

⊙로지스틱 회귀모형

- **로지스틱 회귀모형**
  - 타겟변수를 예측하기 위한 지도학습이며, 타겟 변수가 범주형일 때 사용되는 분류 알고리즘.
  - 타겟변수의 각 범주에 속할 확률을 특성 변수들의 함수로 표현되는 모형식을 정의.
  - 새로운 특성변수 값이 주어지면 추정된 모형식을 이용하여 타겟 변수의 각 범주 별 확률을구하고, 가장 높은 확률을 가지는 범주로 분류함.



지도학습(Supervised Learing) 알고리즘

⊙로지스틱 회귀모형

- **이항 로지스틱 회귀모형**
  - ▫ 모형 정의
    - ♦ 0 또는 1의 값을 가지는 타겟변수 Y 를 여러 특성변수 $x_1,\ldots,$ x_k를 이용한 회귀식 형태로 예측.
    - ♦ $p = P(Y = 1|x_1, \ldots, x_k)$라고 하면,

$$\log\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 x_1 + \ldots + \beta_k x_k$$

    - ♦ 위 모형은 다음과 같이 p에 관한 식으로 표현할 수 있음.

$$p = \frac{\exp(\beta_0 + \beta_1 x_1 + \ldots + \beta_k x_k)}{1 + \exp(\beta_0 + \beta_1 x_1 + \ldots + \beta_k x_k)}$$

지도학습(Supervised Learing) 알고리즘

⊙로지스틱 회귀모형

- **이항 로지스틱 회귀모형**
  - 모형식의 이해
    - 범주형 타겟변수의 범주가 두 개일 때, 관심 범주를 1, 다른 범주를 0으로 정의하면, 타겟변수 Y의 값이 1일 확률을 p 로 두고, 이를 $x_1,\ldots,\ x_k$을 이용하여 예측하고자 함.

$$P(Y=1)=p$$

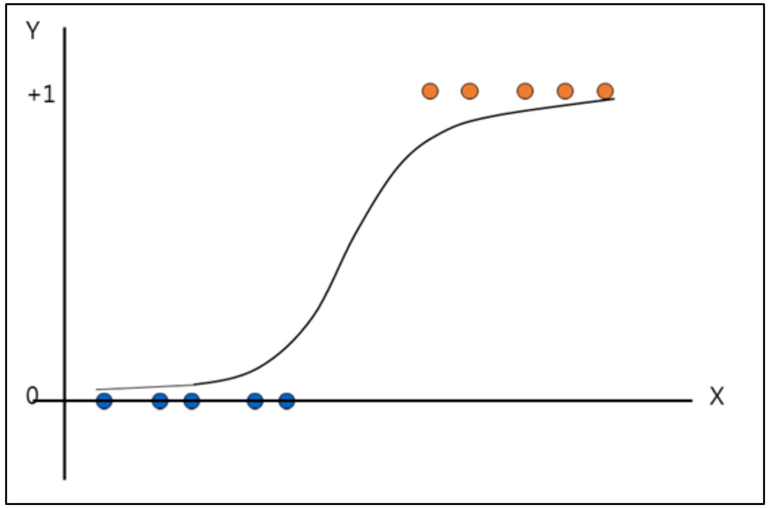
    - 여기서 확률 p 은 0과 1사이의 값이므로, $(-\infty,\infty)$의 값을 가지는 특성변수들 $x_1,\ldots,\ x_k$의 선형함수로 나타내는 것은 부적절함.
    - 이에 확률 p 대신 로그오즈 $\log(\frac{p}{1-p})$를 특성변수의 선형함수로 나타낸 것임.

$$logit(p)=\log(\frac{p}{1-p})=\beta_0+\beta_1x_1+\cdots\beta_kx_k$$

지도학습(Supervised Learing) 알고리즘

⊙로지스틱 회귀모형

- **이항 로지스틱 회귀모형**
  - 모형식의 이해
    - 특성변수가 1개인 경우 ($k = 1$)의 이항 로지스틱 회귀모형
      - $\beta_1 > 0$인 경우 $p = \frac{\exp(\beta_0+\beta_1 x_1)}{1+\exp(\beta_0+\beta_1 x_1)}$ 는 아래와 같은 비스듬한 S 곡선형태를 가짐.



- 타겟변수의 예측값은 언제나 0~1 사이.
- 특성변수가 $-\infty$ 값일 때 타겟변수 예측값은 0.
- 특성변수가 ∞ 값일 때 타겟변수 예측값은 1.

지도학습(Supervised Learing) 알고리즘

⊙로지스틱 회귀모형

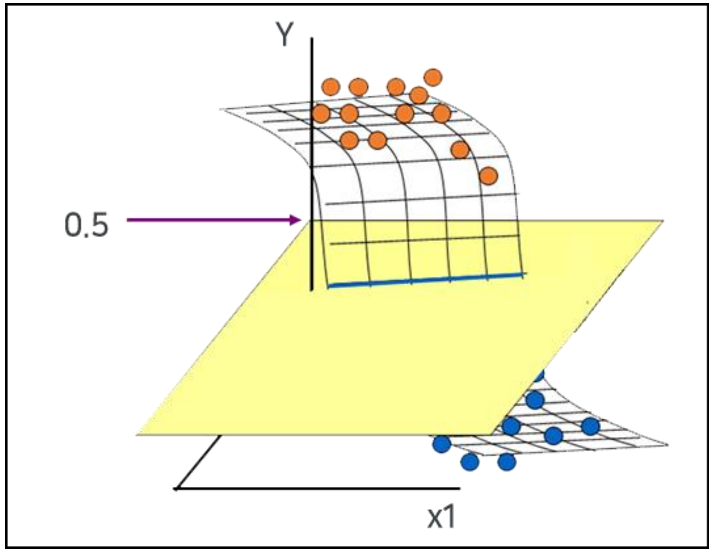
- **이항 로지스틱 회귀모형**
  - 추정 및 예측
    - 훈련자료가 주어졌을 때, 경사하강법 등을 이용하여 가장 적합한 $\beta_0, \beta_1, \dots, \beta_k$를 추정할 수 있음.
    - 특성변수가 1개인 경우 ($k = 1$)의 이항 로지스틱 회귀모형
      - n개의 훈련자료 $(x_i, y_i), i = 1, \dots, n$ 를 이용하여 β_0, β_1을 추정.
      - 새로운 자료의 특성 변수값 x_{new}가 주어졌을 때,
        $P(Y_{new} = 1)$는 $p_{new} = \frac{\exp(\beta_0+\beta_1 x_{new})}{1+\exp(\beta_0+\beta_1 x_{new})}$로 추정됨.
      - $p_{new} \geq cutoff$면, Y_{new}를 1로 예측
        $p_{new} < cutoff$면, Y_{new}를 0으로 예측

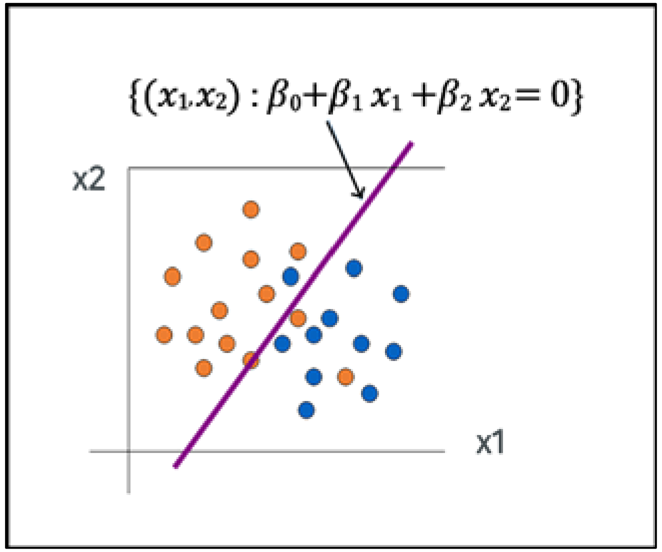
지도학습(Supervised Learing) 알고리즘

⊙로지스틱 회귀모형

- **로지스틱 회귀모형의 분류경계면**
  - 로지스틱 회귀모형은 **선형의 결정경계**를 가짐.
    - 특성변수가 2개인 이항 로지스틱 회귀모형에서 Cutoff=0.5 일때의 초평면 (Hyperplane)

$$p = \frac{e^{\beta_0+\beta_1 x_1+\beta_2 x_2}}{1+e^{\beta_0+\beta_1 x_1+\beta_2 x_2}}$$





지도학습(Supervised Learing) 알고리즘

⊙로지스틱 회귀모형

- **오즈비(odds ratio)**

$$p = \frac{e^{\beta_0+\beta_1 x_1+\beta_2 x_2}}{1+e^{\beta_0+\beta_1 x_1+\beta_2 x_2}} \Leftrightarrow \quad \frac{p}{1-p} = e^{\beta_0+\beta_1 x_1+\beta_2 x_2}$$

▫ 나머지 특성변수는 모두 고정시킨 상태에서, 하나의 특성변수 X_1만 1만큼 증가시켰을 때, 변화하는 odds의 비율(오즈비)는 e^{β_1}임을 알 수 있음.

$$\frac{odds(x_1+1, x_2)}{odds(x_1, x_2)} = \frac{e^{\beta_0+\beta_1(x_1+1)+\beta_2 x_2}}{e^{\beta_0+\beta_1 x_1+\beta_2 x_2}} = e^{\beta_1}$$

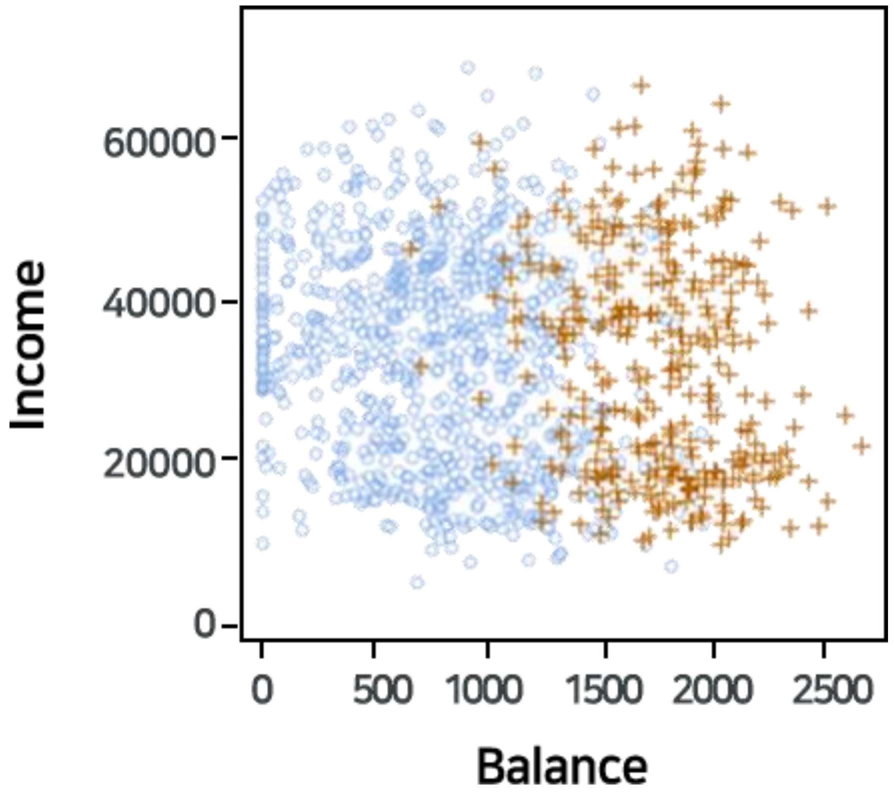
▫ X_1만 1만큼 증가하면, 성공(관심범주, Y=1)에 대한 오즈가 e^{β_1}배 변화함.

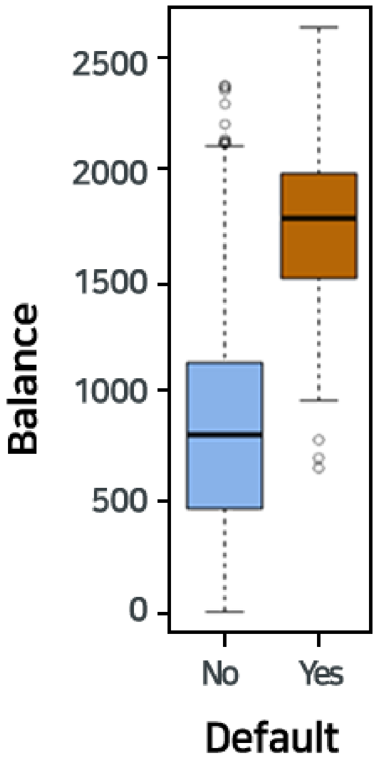
- $\beta_1 > 0$: 관심범주에 속할 확률이 증가함. X_1변수와 관심범주 간에는 양의 상관관계
- $\beta_1 > 0$: 관심범주에 속할 확률이 감소함. X_1변수와 관심범주 간에는 음의 상관관계

지도학습(Supervised Learing) 알고리즘

⊙로지스틱 회귀모형

- **로지스틱 회귀모형 적용사례**
  - 어느 신용카드 회사의 고객의 잔액(balance)을 특성변수로 하여, 고객의 연체(Default) 가능성을 예측하고자 함.





지도학습(Supervised Learing) 알고리즘

⊙로지스틱 회귀모형

- **로지스틱 회귀모형 적용사례**
  - ▫ 이항 로지스틱 회귀모형 추정 결과

	추정치
절편	-10.6513
잔액	0.0055

  - ▫ 추정된 모형식

$$\hat{p} = \frac{\exp(-10.6513 + 0.0055x)}{1 + \exp(-10.6513 + 0.0055x)}$$

    - ♦ 잔액이 $1 증가 시 연체가능성에 대한 오즈는 $e^{0.0055} = 1.0055$배 증가함

  - ▫ 잔액이 $2,000인 경우 연체 가능성 예측

$$\hat{p} = \frac{\exp(-10.6513 + 0.0055 \cdot 2000)}{1 + \exp(-10.6513 + 0.0055 \cdot 2000)} = 0.586$$

지도학습(Supervised Learing) 알고리즘

⊙ 의사결정나무 모델(Decision Tree)

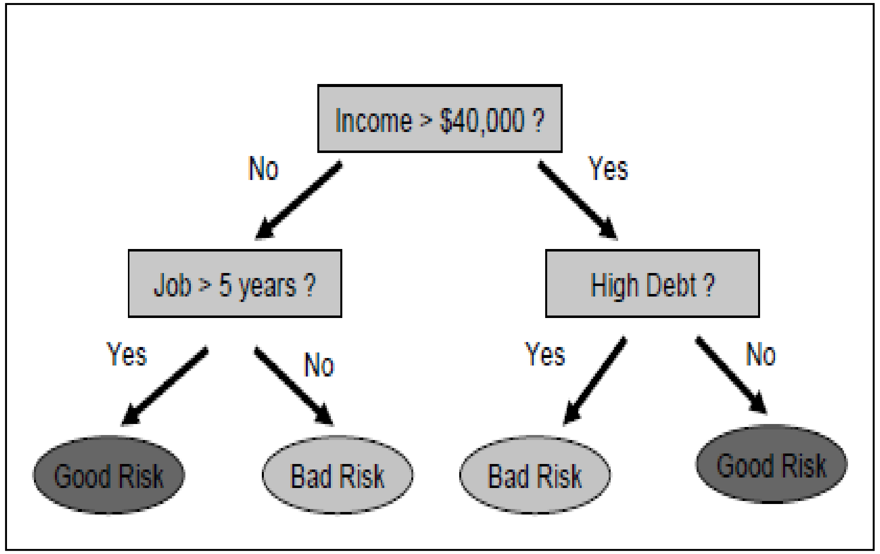
- **의사결정나무 개요**
  - 의사결정규칙(decision rule)을 나무 구조로 도표화하여 관심대상이 되는 집단을 몇 개의 소집단으로 분할하는 방식으로 분류 및 예측하는 분석 방법.
    - 타겟변수가 범주형인 경우 : 분류 나무(classification tree)
    - 타겟변수가 연속형인 경우 : 회귀 나무(regression tree)

  - 의사결정 나무의 종류
    - 의사결정나무 알고리즘은 CART, C4.5, CHAID, QUEST 등과 이들의 장점을 결합한 다양한 알고리즘이 있음.

지도학습(Supervised Learing) 알고리즘

⊙ 의사결정나무 모델(Decision Tree)

- 의사결정 나무의 구조



- 뿌리마디 (root node) : 시작되는 마디로 전체 자료로 구성됨
- 자식마디 (child node) : 하나의 마디로부터 분리되어 나간 2개 이상의 마디들
- 부모마디 (parent node) : 주어진 마디의 상위마디
- 끝마디 (terminal node) : 자식마디가 없는 마디
- 중간마디 (internal node) : 부모마디와 자식마디가 모두 있는 마디
- 깊이 (depth) : 뿌리마디부터 끝마디까지 중간마디의 수

지도학습(Supervised Learing) 알고리즘

⊙ 의사결정나무 모델(Decision Tree)

- **의사결정나무 분석절차**

나무의 성장 (growing)

각 마디에서 적절한 최적의 분리규칙(splitting rule)을 찾아 나무를 성장시킴. 정지 규칙(stopping rule)을 만족하는 경우는 성장을 중단.

가지치기 (pruning)

일반화 오차(generalized error)을 크게 할 위험이 높거나 부적절한 추론 규칙을 가지고 있는 가지를 제거.

타당성 평가

평가자료(test data)를 이용하여 의사결정나무를 평가.

해석 및 예측

구축된 나무 모형을 해석하고 분류 및 예측 모형을 설정.

지도학습(Supervised Learing) 알고리즘

⊙ 의사결정나무 모델(Decision Tree)

- **나무의 성장(growing)**
  - 상위노드로부터 하위노드로 나무 구조를 형성하는 매 단계마다 분리규칙(분류변수와 분류기준값)을 선택함.
  - 분리규칙의 형태(이진분할의 경우)
    - 연속형 특성변수 : 분리에 사용될 특성변수 X와 분리점 c를 이용하여 $X < c$면 왼쪽 자식마디, 그렇지 않으면 오른쪽 자식마디로 자료를 분리.
    - 범주형 특성변수 : 분리에 사용될 특성변수 X가 가지는 전체범주 중 부분집합인 A를 이용하여 $X \in A$면 왼쪽 자식마디, 그렇지 않으면 오른쪽 자식마디로 자료를 분리.
  - 분류기준은 해당 노드에서 그 기준으로 하위 노드를 분기하였을 때,
    - 각 하위 노드 내에서는 동질성
    - 하위 노드들 간에는 이질성

    이 가장 커지도록 하는 기준으로 선택됨.

지도학습(Supervised Learing) 알고리즘

⊙ 의사결정나무 모델(Decision Tree)

- **나무의 성장(growing)**
  - 분류나무(Classification Tree)의 분리규칙 탐색
    - 불순도
      - Y의 범주가 $j = 1, \dots, C$로 구성될 때 t 노드에서의 불순도 $\mathrm{imp}(t)$는 다음과 같이 정의됨.

        $p(j|t)$: t 마디로 분류했는데 범주 j에 속할 확률이라고 할 때,

        · 지니 불순도 (gini impurity) : imp$(t) = 1 - \sum_{j=1}^{C} p(j|t)^2$

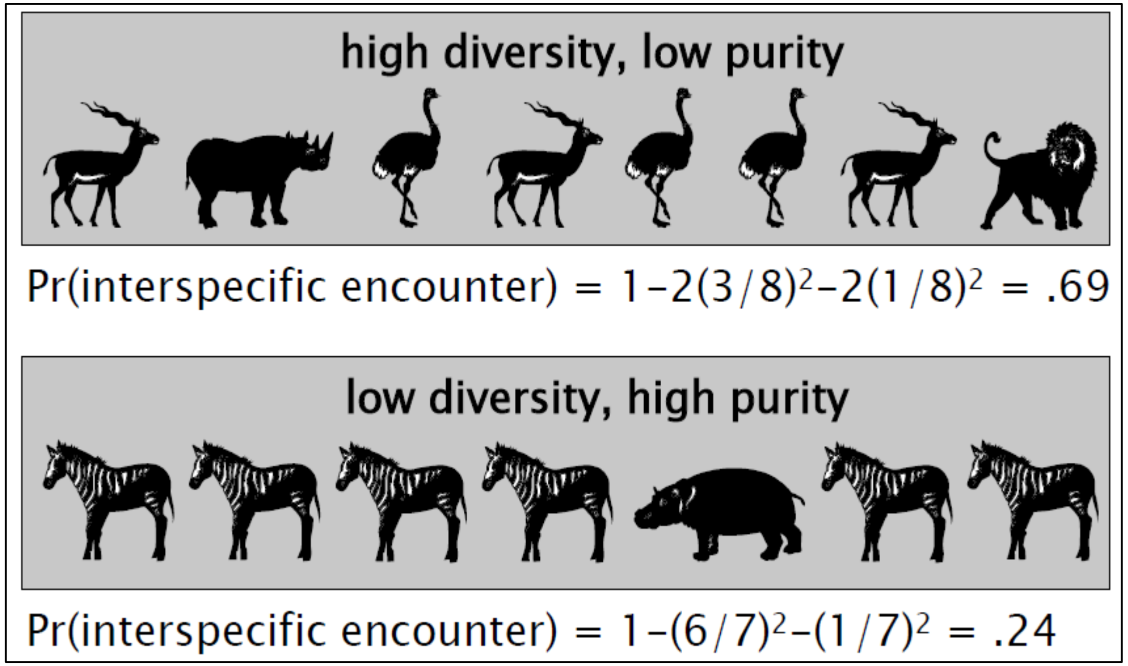
        · 엔트로피 불순도 (entropy impurity) : imp$(t) = -\sum_{j=1}^{C} p(j|t) \cdot \log p(j|t)$

      - 불순도가 클수록 자식노드 내 이질성이 큰 것→불순도가 가장 작아지는 방향으로 가지분할.

지도학습(Supervised Learing) 알고리즘

⊙ 의사결정나무 모델(Decision Tree)

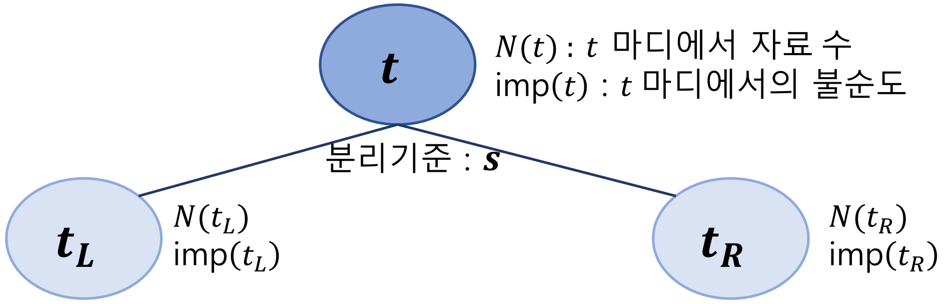
- **나무의 성장(growing)**
  - ▫ 분류나무(Classification Tree)의 분리규칙 탐색
    - ♦ 불순도
      - ▪ 불순도 계산 예시



지도학습(Supervised Learing) 알고리즘

⊙ 의사결정나무 모델(Decision Tree)

- **나무의 성장(growing)**
  - 분류나무(Classification Tree)의 분리규칙 탐색
    - 불순도의 향상된 정도 (Goodness of split) $G(s,t)$
      어떤 부모마디 t에서 분리기준 s로 분리한 뒤 생성된 두 자식마디를 t_L, t_R 이라고 할 때,



$$G(s,t) = \text{imp}(t) - \frac{N(t_L)}{N(t)} \cdot \text{imp}(t_L) - \frac{N(t_R)}{N(t)} \cdot \text{imp}(t_R)$$

- 모든 특성변수와 그 특성변수의 모든 가능한 분리점에 대하여 $G(s,t)$를 구한 뒤 $G(s,t)$가 가장 큰 특성변수 및 분리점을 해당 마디에서의 분리기준으로 정함.

지도학습(Supervised Learing) 알고리즘

⊙ 의사결정나무 모델(Decision Tree)

- **나무의 성장(growing)**
  - 분류나무(Classification Tree)의 분리규칙 탐색
    - 불순도의 향상된 정도를 이용한 분리기준 예시

X1 (성별)	X2 (학력)	Y(상품가입여부)
M	대졸	1
M	고졸	0
F	대졸	1
F	대졸	0
F	고졸	1

- 부모마디

  $\mathrm{imp}(t) = 1 - \left(\frac{2}{5}\right)^2 - \left(\frac{3}{5}\right)^2 = 0.48$

- 자식마디 (X1으로 분리하는 겨우 : $s1$)

  $\mathrm{Imp}(t_L) = 1 - \left(\frac{1}{2}\right)^2 - \left(\frac{1}{2}\right)^2 = 0.5$

  $\mathrm{Imp}(t_R) = 1 - \left(\frac{1}{3}\right)^2 - \left(\frac{2}{3}\right)^2 = 0.4444$

- 자식마디 (X2으로 분리하는 경우 : $s2$)

  $\mathrm{Imp}(t_L) = 1 - \left(\frac{0}{2}\right)^2 - \left(\frac{2}{2}\right)^2 = 0$

  $\mathrm{Imp}(t_R) = 1 - \left(\frac{2}{3}\right)^2 - \left(\frac{1}{3}\right)^2 = 0.4444$

- 불순도 감소량

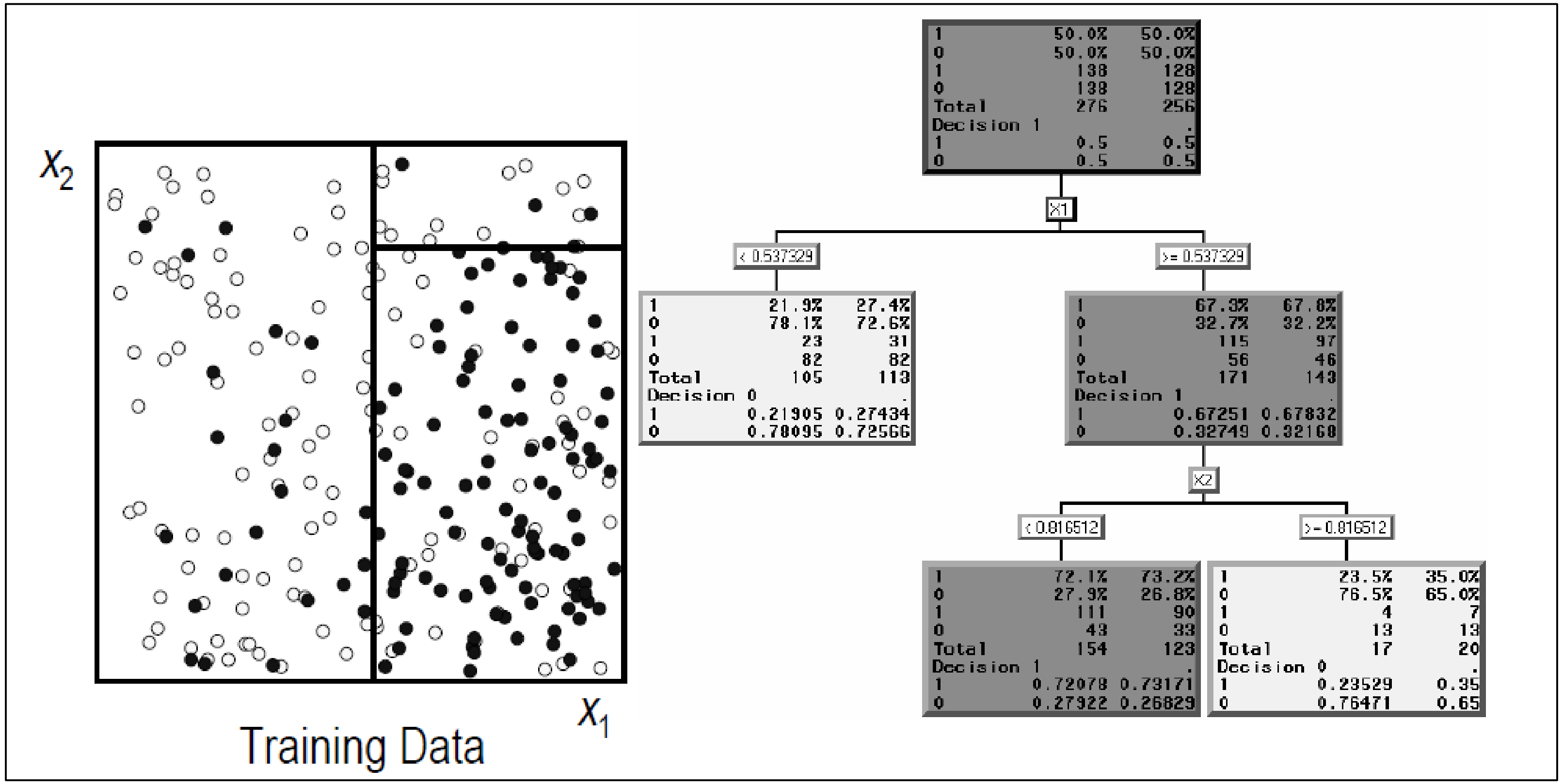
  $G(s1, t) = 0.48 - \frac{2}{5} \times 0.5 - \frac{3}{5} \times 0.4444 = 0.01336$

  $G(s2, t) = 0.48 - \frac{2}{5} \times 0 - \frac{3}{5} \times 0.4444 = 0.21336$

지도학습(Supervised Learing) 알고리즘

⊙ 의사결정나무 모델(Decision Tree)

- **나무의 성장(growing)**
  - ▫ 분류나무의 적합 예시
    - ♦ 분류경계면이 축에 수직임.



지도학습(Supervised Learing) 알고리즘

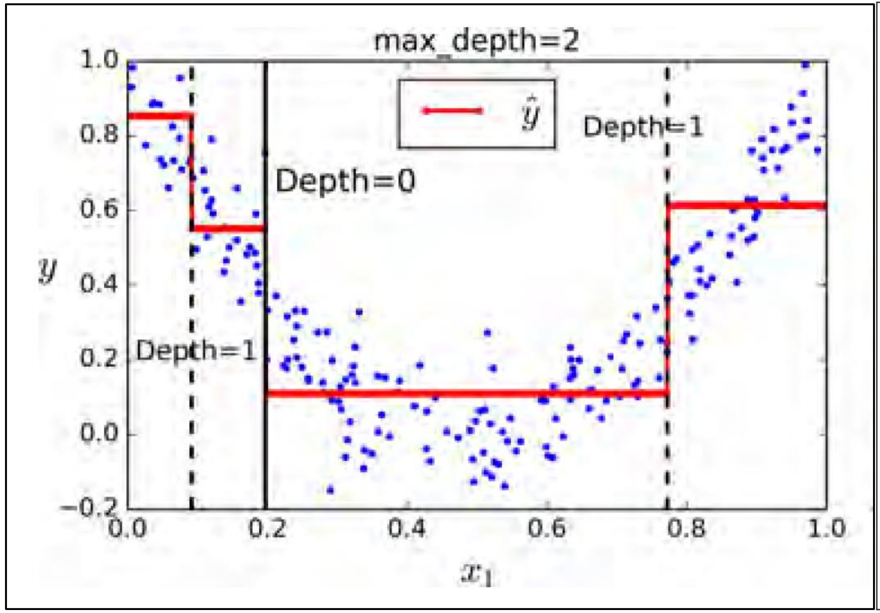
⊙ 의사결정나무 모델(Decision Tree)

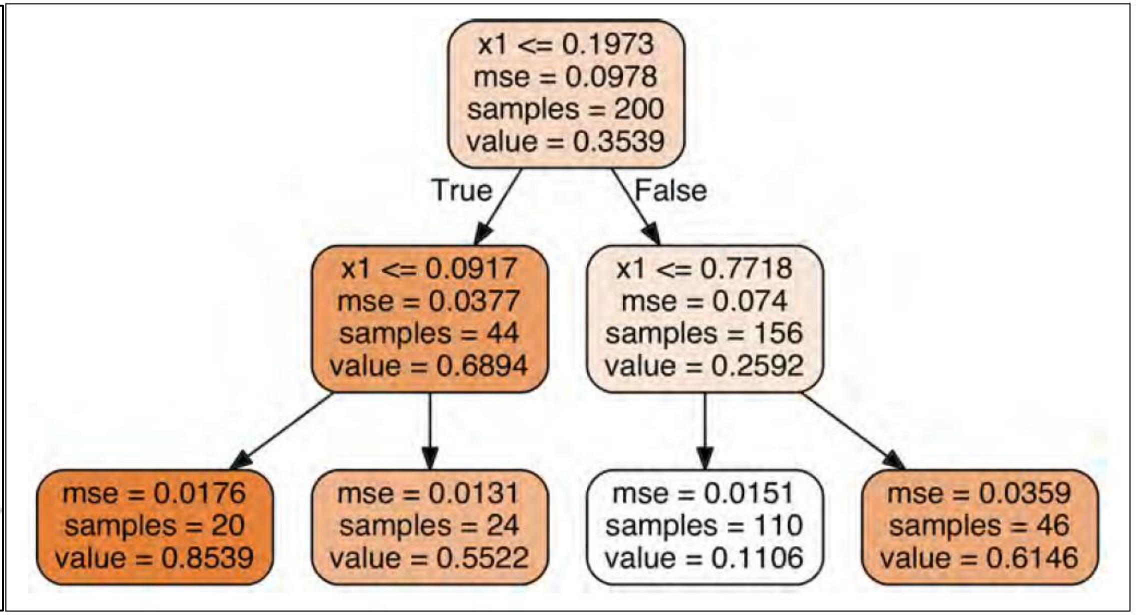
- **나무의 성장(growing)**
  - 회귀나무 (Regression Tree) 의 분리규칙 탐색
    - 분산의 감소량
      - 각 자식 노드 내에서의 타겟변수의 분산(MSE)이 작을수록, 그룹 내 이질성이 작은 것으로 볼 수 있음.
      - 자식 노드로 분리했을 때 분산의 감소량이 가장 커지도록 하는 분리규칙을 탐색.

지도학습(Supervised Learing) 알고리즘

⊙ 의사결정나무 모델(Decision Tree)

- **나무의 성장(growing)**
  - ▫ 회귀나무의 적합 예시





지도학습(Supervised Learing) 알고리즘

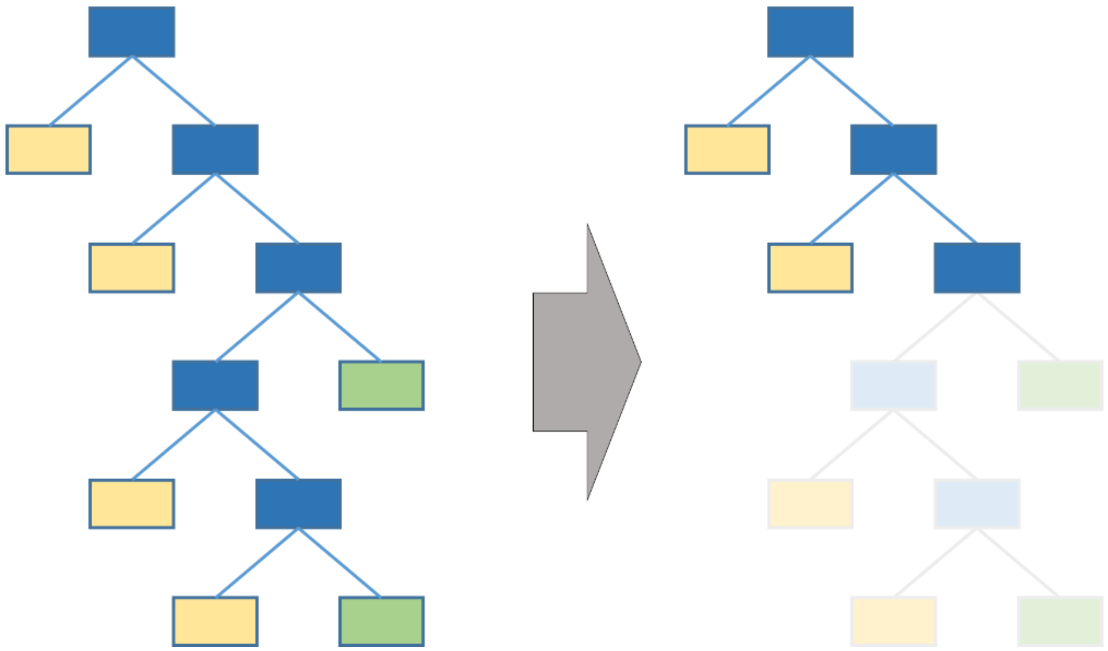
⊙ 의사결정나무 모델(Decision Tree)

- **나무의 성장(growing)**
  - 정지규칙 (stopping rule)
    - 다음의 경우에 더 이상 분리하지 않고 나무가 성장을 멈추도록 함.
      - 모든 자료의 타겟변수 값이 동일할 때
      - 마디에 속하는 자료의 개수가 일정 수준보다 적을 때
      - 뿌리마디로부터의 깊이가 일정 수준 이상일 때
      - 불순도의 감소량이 지정된 값보다 적을 때

지도학습(Supervised Learing) 알고리즘

⊙ 의사결정나무 모델(Decision Tree)

- **가지치기 (pruning)**
  - 지나치게 많은 마디를 가지는 의사결정 나무는 새로운 자료에 적용할 때 예측 오차가 매우 커지는 과적합(overfitting) 상태가 됨.
  - 가지치기는 이를 방지하기 위한 방법으로, 성장이 끝난 나무의 가지를 제거하여 적당한 크기를 가지도록 함.
  - 적당한 크기를 결정하는 방법은 검증용 자료(validation data)에 대한 예측 오류가 가장 작은 나무 모형을 찾는 것이 일반적이며, 이 과정은 보통 의사결정나무 알고리즘 내에 자동화되어 있음.



지도학습(Supervised Learing) 알고리즘

⊙ 의사결정나무 모델(Decision Tree)

- **의사결정나무 모형의 특징**

장점	단점
• 이해하기 쉬운 규칙을 생성함 : if-then-else 방식 • 특성변수 및 타겟변수 둘 다 연속형, 범주형 자료 모두 취급함. • 데이터의 전처리가 거의 필요하지 않음. • 이상치에 덜 민감. • 모형에 가정이 필요없는 비모수적 모형.	• 훈련결과가 불안정함. • 모든 분할은 축에 수직임. • 나무가 깊어질수록 과적합으로 예측력이 저하되며, 해석이 어려워짐.

지도학습(Supervised Learing) 알고리즘

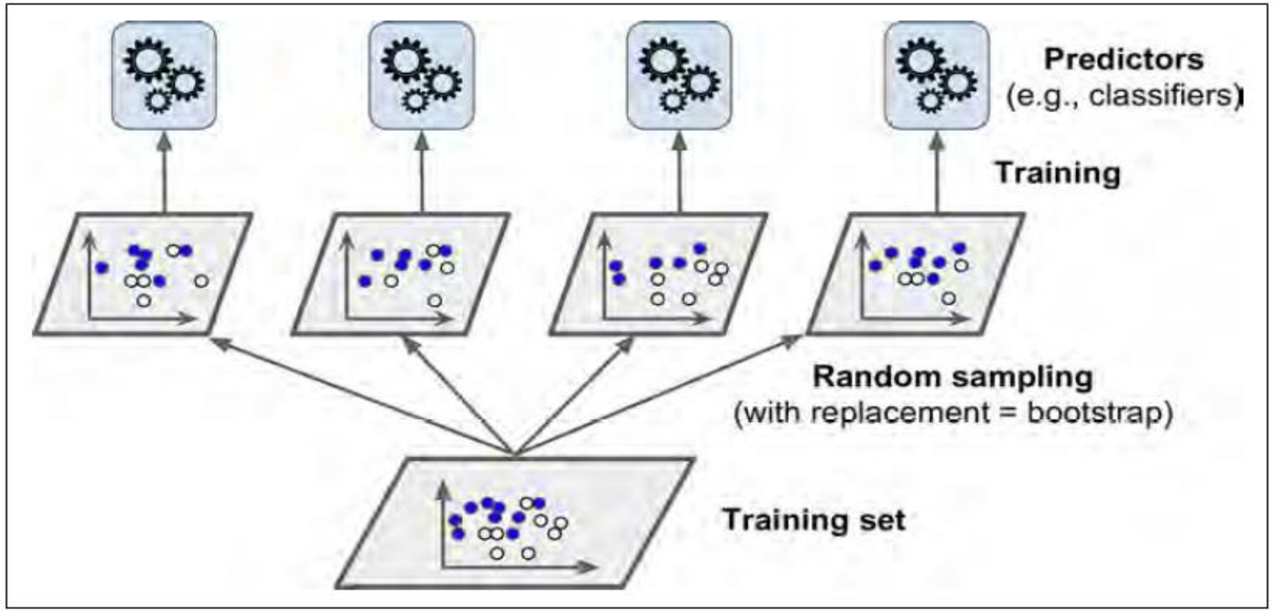
⊙ 앙상블 학습 (Ensemble Learning)

- **앙상블 학습 개요**
  - 대중의 지혜 (wisdom of the crowd)
    - 여러 예측기(분류 또는 회귀 모형)들로부터 예측을 수집하면 가장 좋은 모델 하나보다 더 좋은 예측을 얻을 수 있음.
  - 단일 예측기를 여러 개를 통합하여 만든 예측기를 앙상블(Ensemble)이라고 함.
  - 대표적인 앙상블 학습 방법
    - 배깅(Bagging)
    - 랜덤포레스트(Random Forest)
    - 부스팅(Boosting)
    - 그래디언트 부스팅(Gradient Boosting), XGBoost

지도학습(Supervised Learing) 알고리즘

⊙ 앙상블 학습 (Ensemble Learning)

- **배깅(bagging)**
  - 배깅 알고리즘 개요
    - 주어진 훈련 데이터에 대하여 bootstrap 자료 (n개의 훈련 데이터 중 중복을 포함하여 n개를 샘플링해서 만든 자료)를 M회 반복 생성하여 각각에 대한 예측 모형을 생성한 후 결합하여 최종적으로 하나의 예측 모형을 만드는 방법.
    - 새로운 관찰치에 대한 예측은
      - 회귀모형 : 각 예측기에서 얻은 예측값들의 평균
      - 분류모형 : 각 예측기에서 얻은 범주들 중 다수결 투표로 적용하여 하나를 선택



지도학습(Supervised Learing) 알고리즘

⊙ 앙상블 학습 (Ensemble Learning)

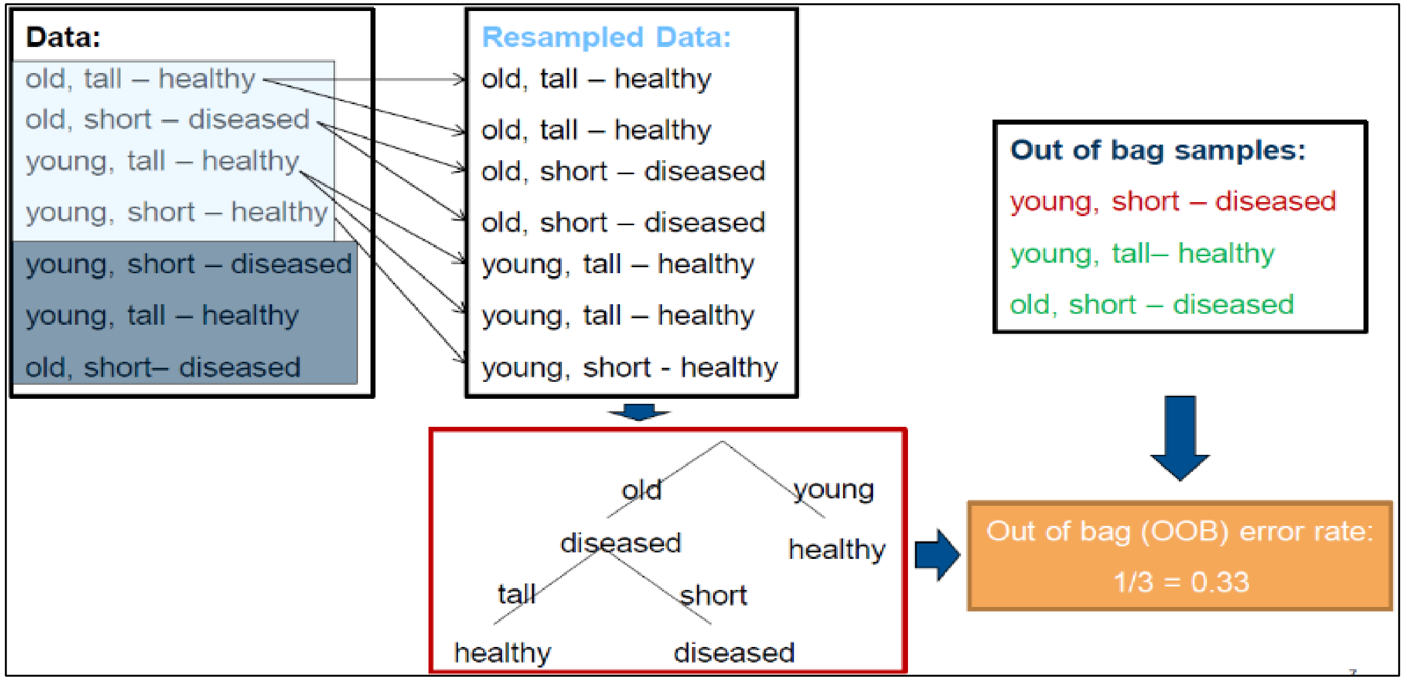
- **배깅(bagging)**
  - 배깅 알고리즘 성능 이해
    - 배깅은 개별 학습기를 통합하는 과정에서 개별 학습기의 오차가 averaging되는 효과가 있어, 하나의 학습기만 사용하는 것 보다 오차의 분산이 작아지게 됨.
      - 오차의 분산은 특히 개별 학습기 간에 연관성이 낮을수록 더욱 작아지게 되는데, 배깅의 경우 bootstrap 샘플링으로 훈련 자료를 다양하게 만들어 줌으로써 개별 학습기 간의 상관관계가 낮아지도록 의도한 것임.
    - 한편, 배깅을 구성하는 개별 학습기는 모든 자료를 다 사용한 것이 아니기 때문에 훈련 데이터 전체를 사용한 것 보다 조금 더 편향되어 있음.
    - 그럼에도 배깅은 하나의 학습기만 사용하는 경우에 비해 결과적으로 성능이 더 좋아지는 경우가 많음. 이는 편향에 따른 성능 저하에 비해 분산이 줄어드는 것에 따른 성능 향상 효과가 더 크기 때문으로 이해됨.

    - 배깅에 가장 적합한 알고리즘은 편의가 적고 분산이 큰(low bias and high variance), 과대적합된 모형임. 특히 가지치기를 하지 않은(unpruned) 의사결정나무가 배깅과 결합하면 우수한 성능을 기대할 수 있음.

지도학습(Supervised Learing) 알고리즘

⊙ 앙상블 학습 (Ensemble Learning)

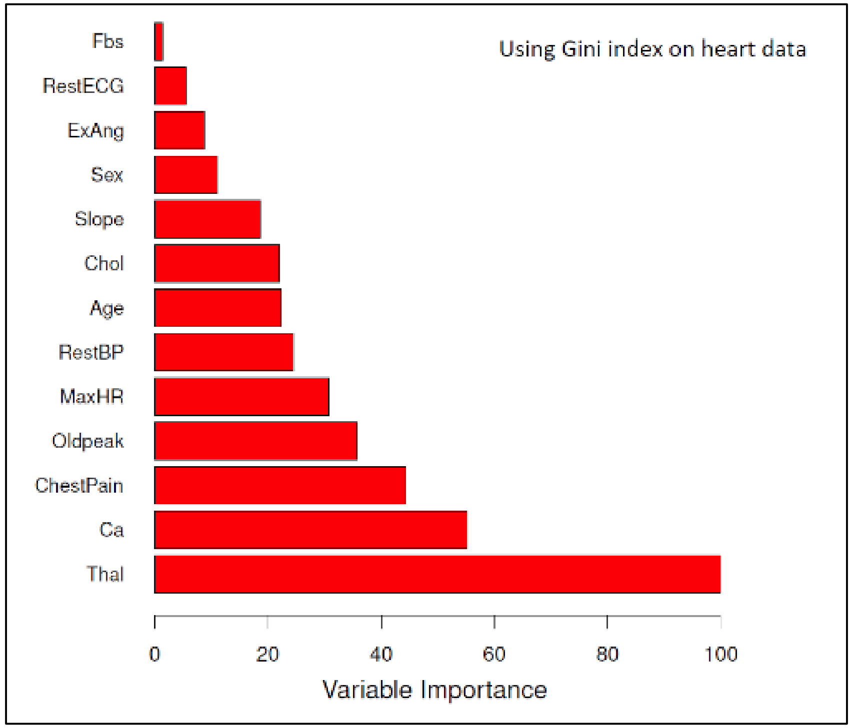
- **배깅(bagging)**
  - OOB 평가
    - Bootstrap 방식을 적용할 때, 매번 전체 훈련 자료 중 약 37%는 선택되지 않게 되는데 이를 OOB(out-of-bag) 샘플이라고 함.
    - 각 학습기가 훈련되는 동안 OOB 샘플은 사용하지 않으므로, 평가 데이터나 교차검증 방식을 쓰지 않아도, OOB 샘플로 앙상블의 성능을 평가할 수 있음. 이를 OOB 평가라고 함.
    - 예측력을 평가하기 위해 추가적인 계산이 거의 필요하지 않다는 장점이 있어 많이 활용됨.



지도학습(Supervised Learing) 알고리즘

⊙ 앙상블 학습 (Ensemble Learning)

- **배깅(bagging)**
  - ▫ 변수 중요도 (Variable Importance Measure)
    - ♦ 배깅은 하나의 의사결정나무를 이용하는 것보다 정확도가 향상되는 대신 해석력을 잃게 됨. 이러한 단점을 보완하기 위하여 배깅 분석 시 변수 중요도를 함께 고찰함.
    - ♦ 배깅을 구성하는 각 의사결정나무마다, 각 특성변수 X_i 별로, 해당 변수로 노드가 분리되었을 때, 불순도가 감소한 정도를 다 더한 뒤, M개의 의사결정나무에 대해 이를 평균한 값으로 정의.



지도학습(Supervised Learing) 알고리즘

⊙ 앙상블 학습 (Ensemble Learning)

- **랜덤 포레스트 (Random Forests)**
  - 랜덤 포레스트 알고리즘 개요
    - 의사결정나무를 기반 알고리즘으로 하는 배깅에서 더 많은 무작위성을 주기 위하여 특성변수 선택에 임의성을 추가한 방법.
      - 원 훈련 자료로부터 bootstrap 샘플을 M번 추출한 뒤 각 샘플에 대해 의사결정나무를 형성한 뒤 이를 통합하여 예측을 한다는 점은 배깅과 동일.
      - 배깅과의 차이점은 의사결정 나무의 각 노드에서 주어진 p개의 특성변수를 모두 검토한 뒤 최적의 분할을 선택하는 방법 대신, $k(< p)$개의 특성변수만 임의로 추출한 뒤, 추출된 변수 내에서 최적의 분할을 만든다는 것.
      - 배깅의 구조에서 변수선택에 임의성을 더해주면, 개별 학습기가 더욱 다양해지고 개별 학습기 간 연관성이 더 낮아질 것이라는 의도임.
    - 랜덤포레스트는 튜닝할 하이퍼파라미터가 많지 않아 사용하기가 쉬운데, 예측력과 성능도 뛰어난 편이라, 매우 인기있는 알고리즘으로 자리매김하였음.

지도학습(Supervised Learing) 알고리즘

⊙ 앙상블 학습 (Ensemble Learning)

- **랜덤 포레스트 (Random Forests)**
  - 랜덤 포레스트의 하이퍼 파라미터
    - 개별 학습기에서 후보 특성변수의 개수 k
      - 주어진 데이터에 가장 적절한 값을 교차검증 등으로 튜닝
      - 랜덤 포레스트를 고안한 Leo Breiman는 다음과 같은 초기값을 제안함.
        - 회귀 : $k = p/3$
        - 분류 : $k = \sqrt{p}$
    - 트리의 개수 M
      - 랜덤 포레스트나 배깅 알고리즘에서 M은 모델의 복잡도와 무관하며, M을 증가시킨다고 해도 과적합을 야기하지는 않음.
      - 일반적으로 랜덤 포레스트나 배깅에서 M을 증가시키면 평가 데이터의 오차는 점차 감소하다가 일정 수준에서 수렴하는 경향을 보이는 경우가 많음. 이는 개별 학습기가 서로 연관되어 있어 M의 증가로 얻을 수 있는 앙상블의 분산감소효과에 한계가 있기 때문으로 이해됨.

지도학습(Supervised Learing) 알고리즘

⊙ 앙상블 학습 (Ensemble Learning)

- **랜덤 포레스트 (Random Forests)**
  - 랜덤 포레스트 성능 이해
    - 랜덤 포레스트는 변수 선택의 임의성으로 인해 개별 학습기의 편향이 증가하며, 이는 성능을 떨어뜨리는 요소가 됨.
    - 그러나 랜덤포레스트의 분산은 배깅과 마찬가지로 개별 학습기들 간의 상관계수에 의존함. 임의의 후보군 안에서 특성변수를 선택함으로써, 개별 학습기 간의 상관계수가 배깅보다 더욱 낮아지고, 결과적으로 분산이 작아지게 됨. 즉, 분산 감소에 따른 성능 향상을 기대할 수 있음.
    - 결국 배깅과 마찬가지로 랜덤 포레스트의 성공도 편향에 따른 성능 저하에 비해, 분산이 줄어드는 것에 따른 성능 향상 효과가 훨씬 더 크기 때문인 것으로 이해할 수 있음.
    - 랜덤 포레스트도 편향이 적고, 분산이 큰 알고리즘과 결합하였을 때 성능이 크게 향상되며, 특히 가지를 치치 않은(unpruned) 의사결정나무와 결합한 랜덤 포레스트가 일반적으로 많이 활용됨.

지도학습(Supervised Learing) 알고리즘

⊙ 앙상블 학습 (Ensemble Learning)

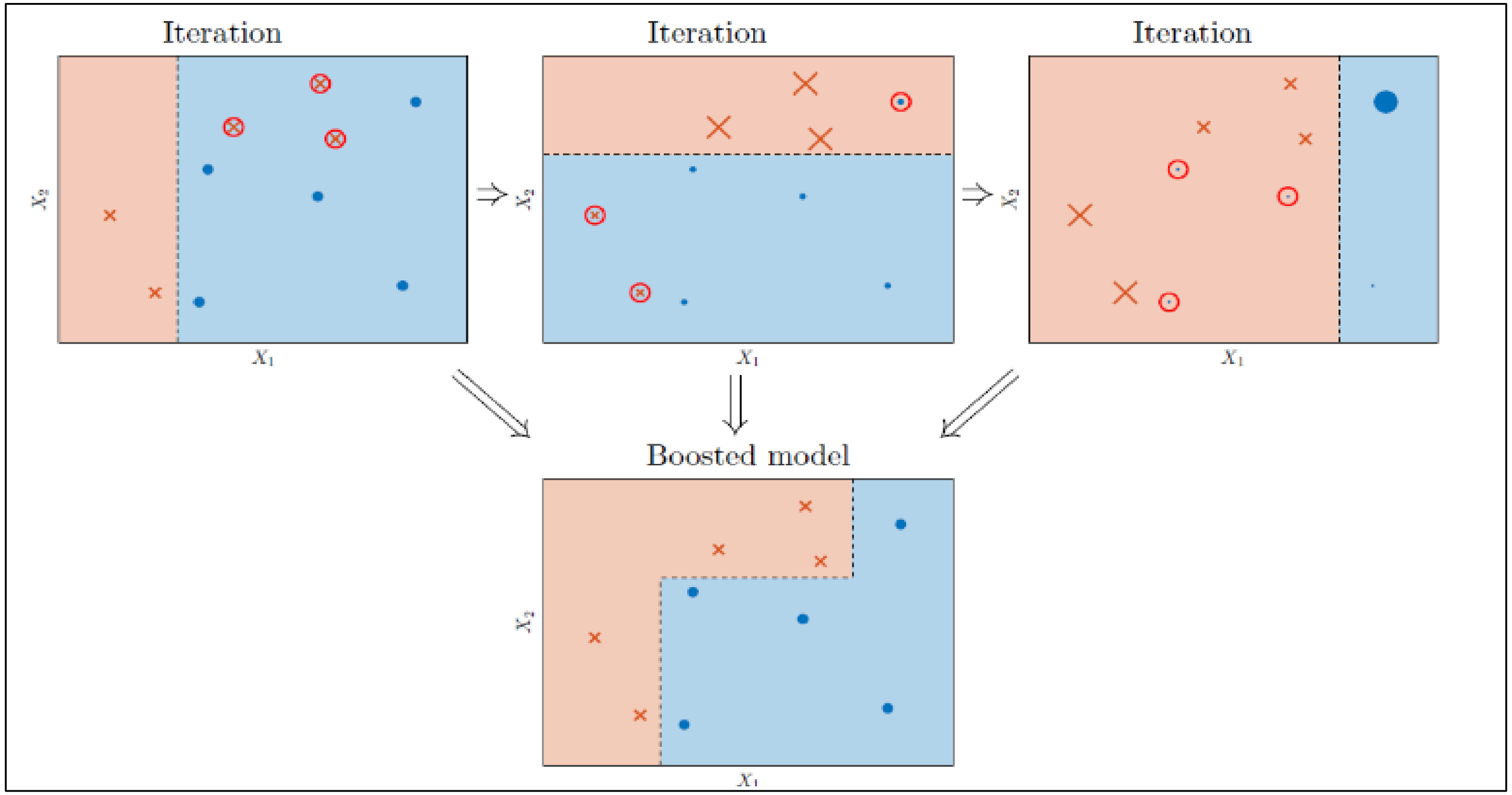
- **부스팅(Boosting)**
  - 부스팅 알고리즘 개요
    - 머신러닝 알고리즘의 아이디어 가운데 가장 성공적인 것 중 하나임.
    - 예측력이 약한 학습기(weak learner, 랜덤하게 예측하는 것 보다 약간 좋은 예측력을 가지는 학습기)들을 결합하여 예측력이 최적에 가까운 강한 모형 (strong learner)을 만들고자 함.
    - AdaBoost 알고리즘
      - 이전 예측기에서 오분류된 샘플의 가중치는 증가(boost)시키고 정분류된 샘플의 가중치를 감소시킴. 이렇게 새로운 예측기는 학습하기 어려운 샘플을 더 잘 맞추도록 순차적으로 개선함.
      - 지정된 예측기의 수에 도달하거나 완벽한 예측기가 만들어지면 중단됨.
      - 훈련을 마친 예측기를 결합할 때, 각 예측기의 성능을 반영한 가중치를 구한 뒤, 가중 결합되도록 함.
      - 부스팅은 잘못 예측된 것의 가중치를 높여가는 방식이기 때문에 노이즈가 많은 자료에 대해서는 성능이 많이 떨어지며, M이 증가함에 따라 과적합(overfitting)의 문제를 야기할 수 있어 교차검증 등으로 적절한 M을 튜닝할 필요가 있음.

지도학습(Supervised Learing) 알고리즘

⊙ 앙상블 학습 (Ensemble Learning)

• 부스팅(Boosting)

- 부스팅 알고리즘 개요



지도학습(Supervised Learing) 알고리즘

⊙ 앙상블 학습 (Ensemble Learning)

- **부스팅(Boosting)**
  - 부스팅의 성능 이해
    - 배깅, 랜덤포레스트 등이 분산 감소를 목적으로 하는 기법이라면, 부스팅은 편향을 줄여 줌으로써 성능을 개선하고자 하는 기법임.
    - 부스팅도 개별 학습기를 결합한 것으로 분산 감소효과를 기대할 수 있지만, 그 효과가 그리 크지 않다는 견해가 많음.
    - 부스팅은 편향이 크고, 분산이 작은(high bias and low variance) 알고리즘, 즉 약한 학습기(weak learner)와 결합하는 것이 좋음. 대표적으로 그루터기 트리모형(stump tree)가 많이 이용됨.