Chapter4. Ensemble Learning

고려대학교 산업경영공학과

DMQA Lab.

임 새 린

목차

Bagging-based Ensemble

Bagging based Ensemble

Bagging

❖ Bootstrapping

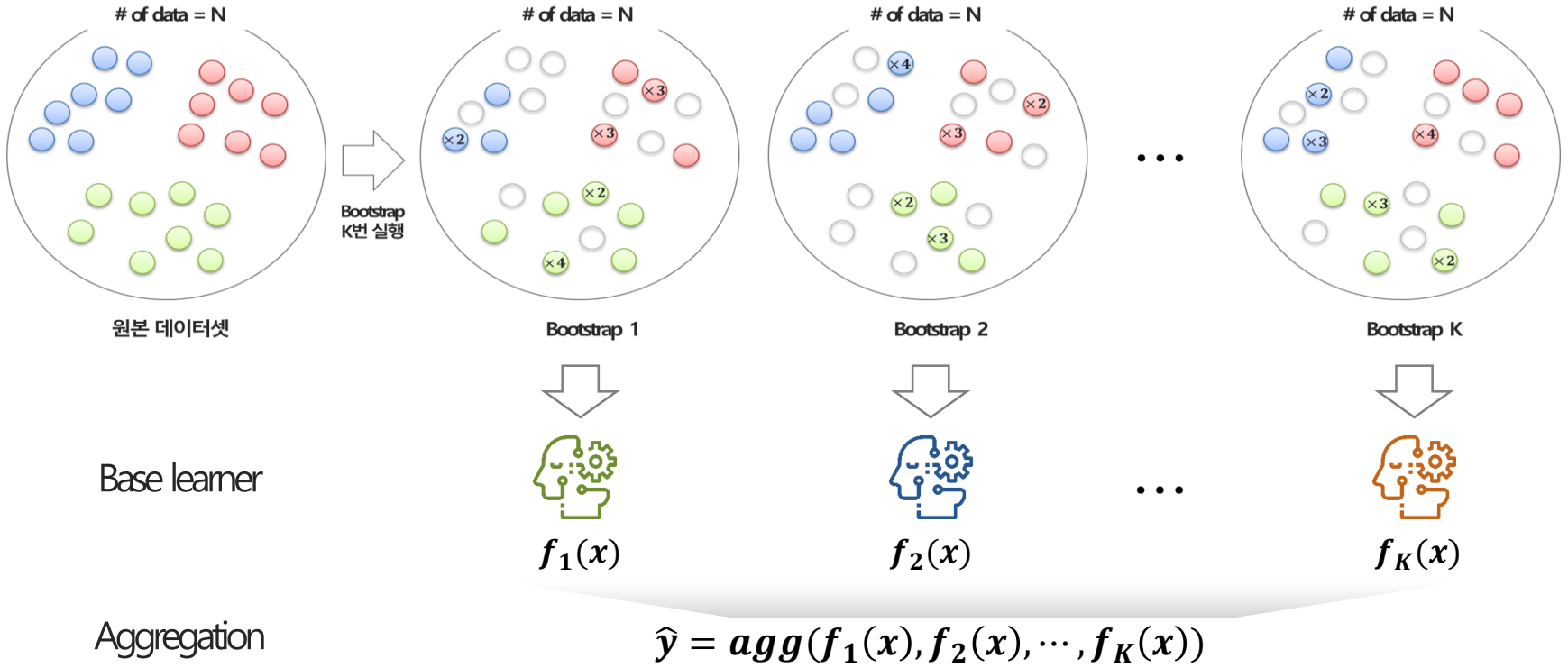
- 원본 데이터셋에서 무작위로 복원추출하는 샘플링 기법
- 복원추출하기 때문에 원하는 만큼 데이터셋을 늘릴 수 있음

Bagging based Ensemble

Bagging

❖ Bagging: Bootstrap Aggregating

- Bagging이란 원본 데이터셋으로부터 bootstrap을 여러 번 적용하여 원본 데이터와 동일한 사이즈를 가지는 bootstrap 데이터셋을 여러 개 만들고, 각 데이터셋마다 앙상블을 구성하는 base learner를 학습시키는 방법

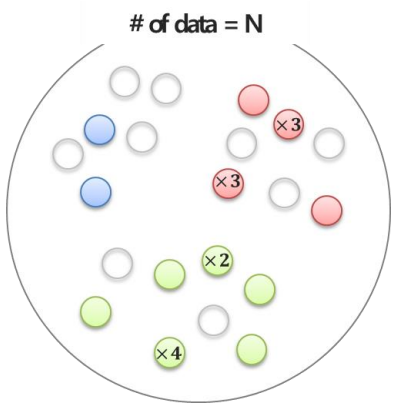


Bagging based Ensemble

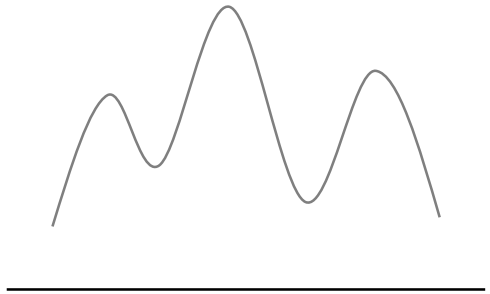
Bagging

❖ 학습 관점에서 Bagging의 장점

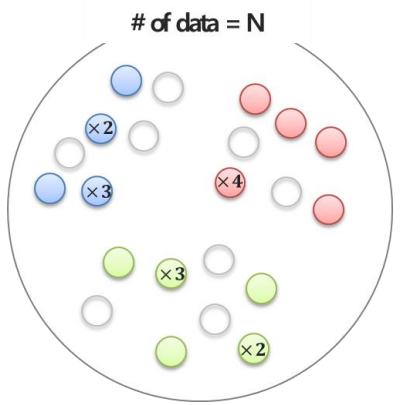
- 특정 관측치는 여러 번 선택되는 반면 아예 포함이 되지 않는 관측치도 있어서 다양한 분포를 가지는 데이터셋 생성
- 다양한 분포를 통해 학습하기 때문에 분포의 변화에 훨씬 강건한 앙상블 모델을 구축할 수 있음 → 예측값 분산 ↓
- 때문에 일반적으로 bagging 기법을 활용한 앙상블에서는 편향이 적고 분산이 큰 복잡한 모델을 활용



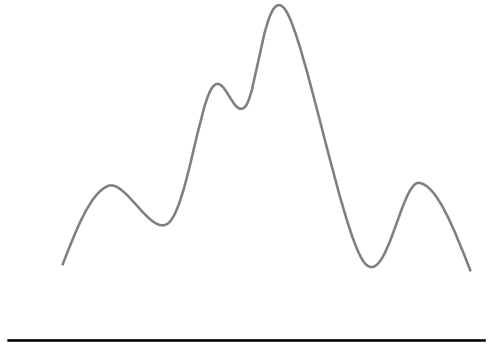
Bootstrap 1



Bootstrap 1
Data distribution



Bootstrap 2



Bootstrap 2
Data distribution

Bagging based Ensemble

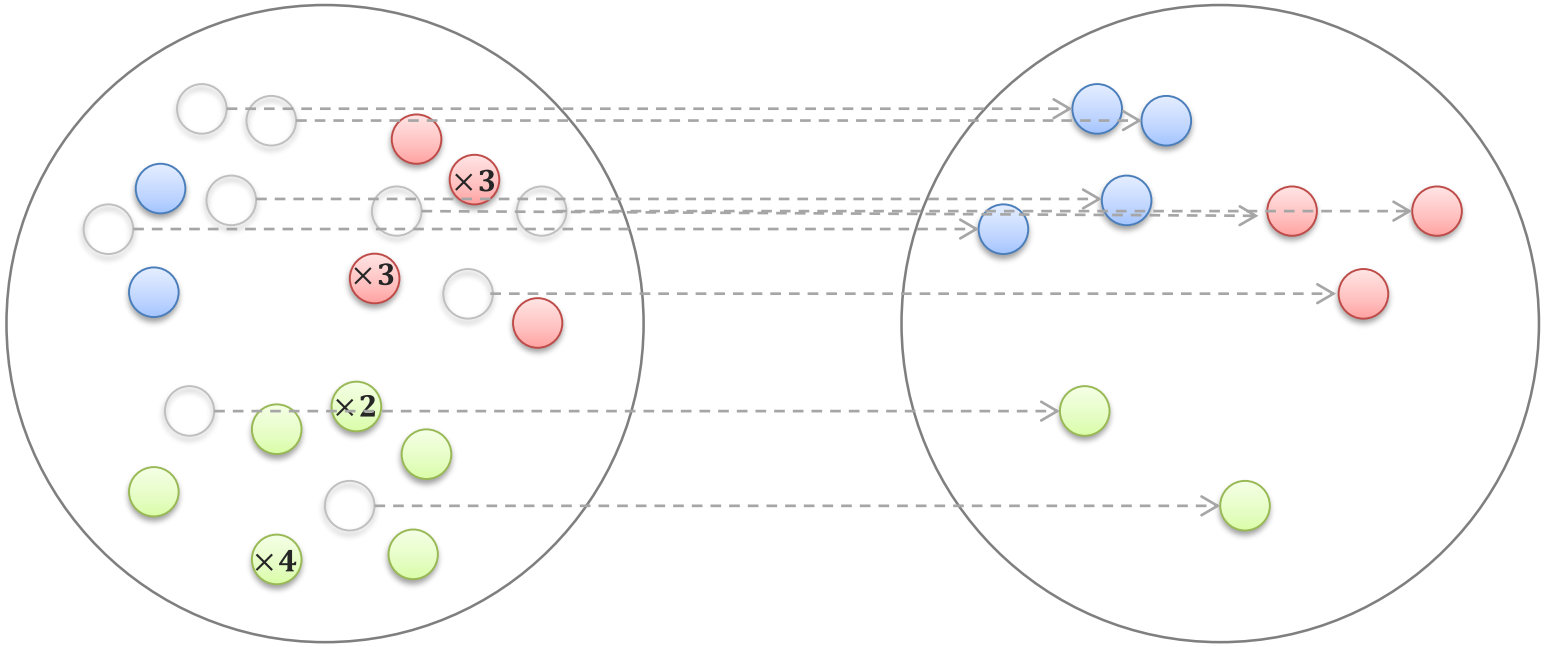
Bagging

❖ 검증 관점에서 Bagging의 장점

- 각 bootstrap마다 선택되지 못한 데이터들의 집합 OOD (Out of Bag)를 검증 집합으로 활용하여 일반화 성능을 확보할 수 있음
- 한 bootstrap에서 한 관측치가 N번의 복원추출에서 한번도 선택되지 않을 확률 = 0.368

$$p = \left(1 - \frac{1}{N}\right)^N$$

$$\lim_{N \to \infty} \left(1 - \frac{1}{N}\right)^N = e^{-1} = 0.368$$



Bootstrap K
학습 데이터로 활용(약 63%)

Out of Bag K
검증 데이터로 활용(약 37%)

Bagging based Ensemble

Random Forest

❖ Random Forest

- **Decision Tree를 base learner로 앙상블의 다양성을 확보하기 위해 두 가지 방법 활용**
- Bagging : 원 데이터셋 사이즈만큼 복원 추출하여 여러 개의 bootstrap 데이터셋을 생성
- 분기 변수 랜덤화 : Decision tree에서 분기를 할 때, 모든 변수를 대상으로 하는 것이 아닌 변수들의 부분 집합에서 랜덤으로 선택

X	Y
x_1	y_1
x_2	y_2
x_3	y_3
x_4	y_4
x_5	y_5
x_6	y_6
x_7	y_7

Original Set

Bagging

X	Y
x_3	y_3
x_6	y_6
x_6	y_6
x_1	y_1
x_3	y_3
x_5	y_5
x_1	y_1

Bootstrap 1

X	Y
x_6	y_6
x_4	y_4
x_2	y_2
x_4	y_4
x_1	y_1
x_5	y_5
x_2	y_2

Bootstrap 2

...

X	Y
x_2	y_2
x_1	y_1
x_3	y_3
x_7	y_7
x_4	y_4
x_2	y_2
x_3	y_3

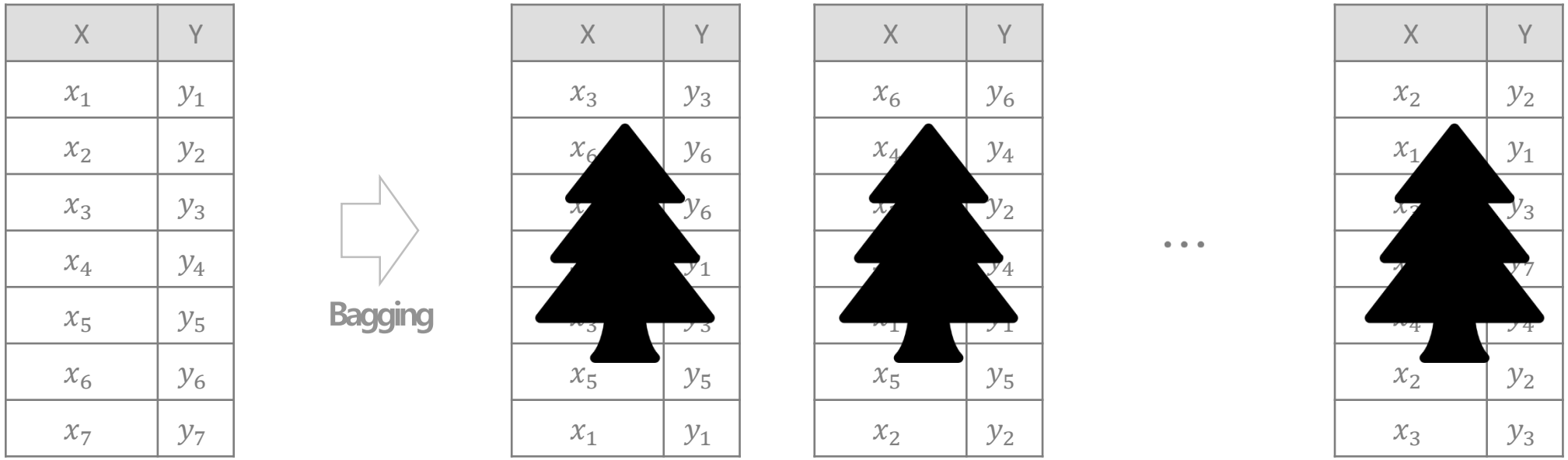
Bootstrap K

Bagging based Ensemble

Random Forest

❖ Random Forest

- **Decision Tree를 base learner로 앙상블의 다양성을 확보하기 위해 두 가지 방법 활용**
- Bagging : 원 데이터셋 사이즈만큼 복원 추출하여 여러 개의 bootstrap 데이터셋을 생성
- 분기 변수 랜덤화 : Decision tree에서 분기를 할 때, 모든 변수를 대상으로 하는 것이 아닌 변수들의 부분 집합에서 랜덤으로 선택



Bagging based Ensemble

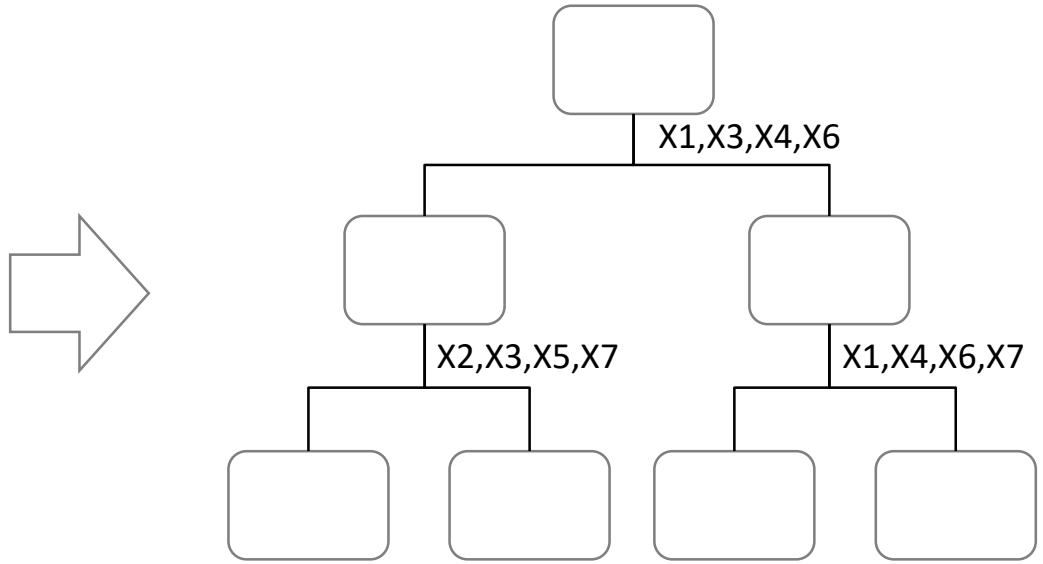
Random Forest

❖ Random Forest

- **Decision Tree를 base learner로 앙상블의 다양성을 확보하기 위해 두 가지 방법 활용**
- Bagging : 원 데이터셋 사이즈만큼 복원 추출하여 여러 개의 bootstrap 데이터셋을 생성
- 분기 변수 랜덤화 : Decision tree에서 분기를 할 때, 모든 변수를 대상으로 하는 것이 아닌 변수들의 부분 집합에서 랜덤으로 선택

	X1	X2	X3	X4	X5	X6	X7
1	2	-18	115	0	34	0.68	57
2	4	-13	334	0	48	0.15	52
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
N-1	3	-15	186	1	51	0.44	59
N	8	-23	138	0	23	0.6	50

Bootstrap K

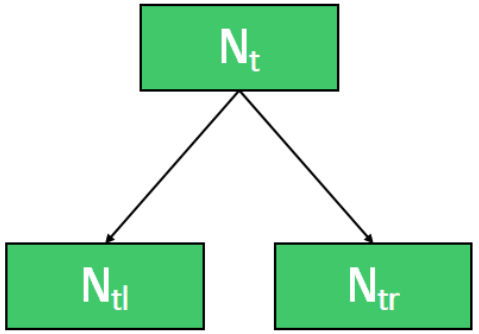


Bagging based Ensemble

Random Forest

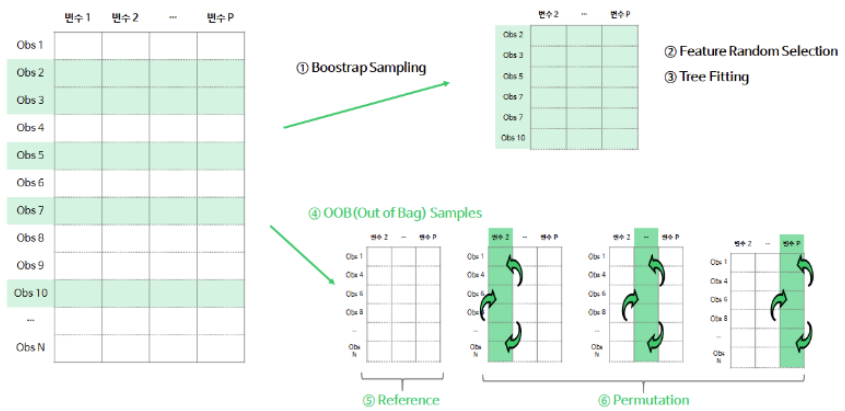
❖ Feature Importance

- 랜덤 포레스트는 OOB 데이터를 활용해서 변수의 중요도를 정량화할 수 있음
  - MDI(Mean Decrease in Impurity) importance : 변수가 분기될 때 impurity 감소분의 평균을 중요도로 정의
  - Permutation importance : 중요도를 확인할 변수의 값을 무작위로 바꾸고 성능 차이를 중요도로 정의
  - Drop column importance : 중요도를 확인할 변수를 제거하고 성능 차이를 중요도로 정의

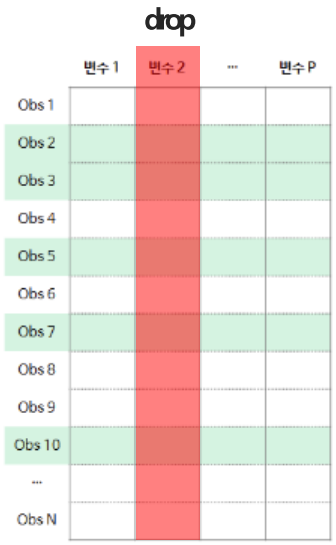


$$\Delta i(t) = i(t) - \frac{N_{tl}}{N_t} i(t_l) - \frac{N_{tr}}{N_t} i(t_r)$$

MID importance



Permutation importance



Drop column importance

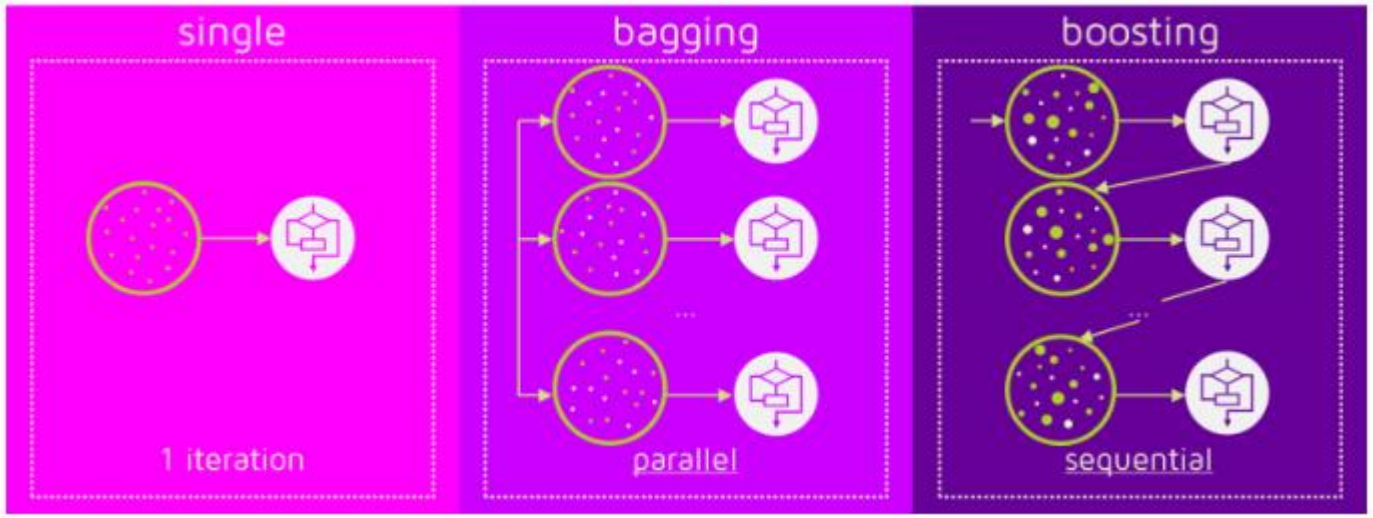
Boosting-based Ensemble

Boosting based Ensemble

boosting

❖ Bagging과 boosting

- Bagging은 bootstrap을 통해 다양한 데이터셋을 만들어 독립적으로 각 모델을 학습
- Boosting은 여러 weak learner을 어떤 가이드에 따라 **순차적**으로 학습



https://quantdare.com/what-is-the-difference-between-bagging-and-boosting/

Boosting based Ensemble: AdaBoost

AdaBoost

❖ AdaBoost Idea

- 모델 복잡도가 낮은 weak learner에 적절한 가이드를 주면 strong learner가 될 수 있지 않을까?
- 어떻게 가이드를 줄까? Weak learner를 순차적으로 학습시키면서 이전 learner들이 어려워 했던 관측치에 가중치를 주자!!

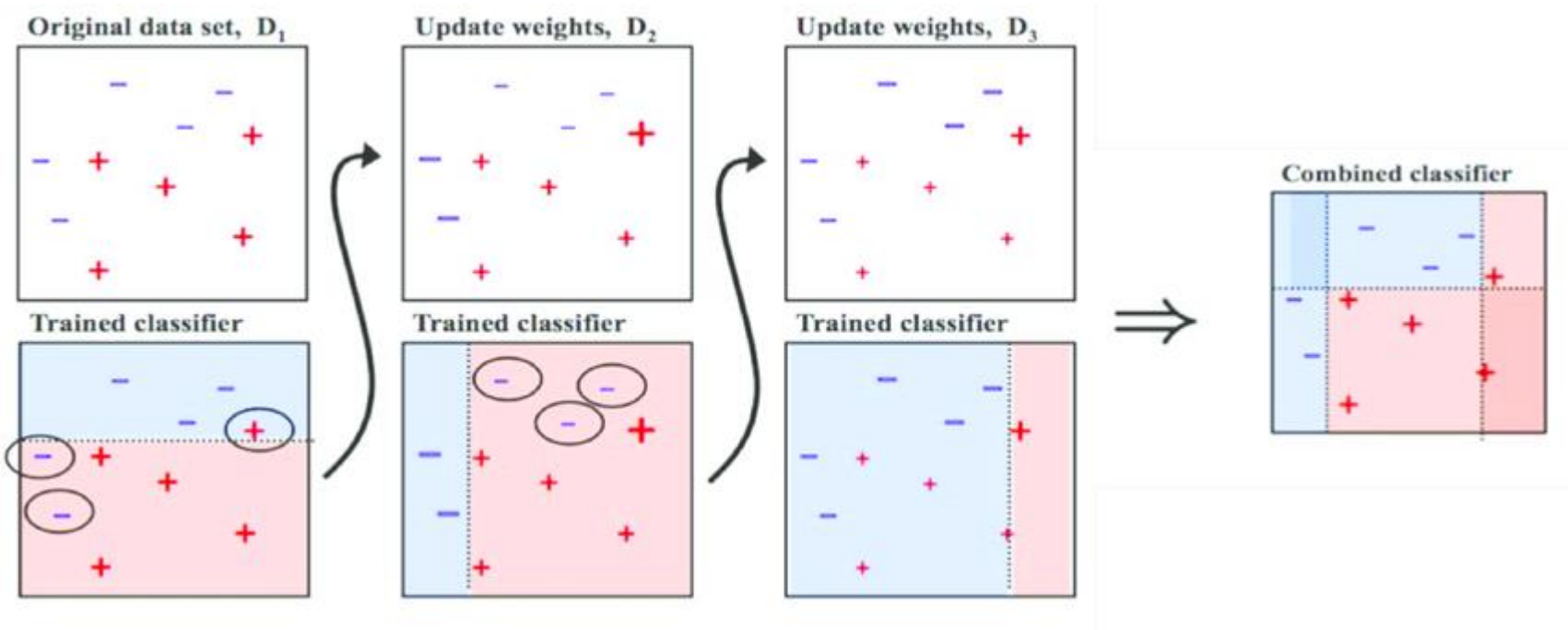


그림 출처: Medium (Boosting and Bagging explained with examples)

Boosting based Ensemble: AdaBoost

AdaBoost

❖ AdaBoost Algorithm

- 모델 복잡도가 낮은 weak learner에 적절한 가이드를 주면 strong learner가 될 수 있지 않을까?
- 어떻게 가이드를 줄까? Weak learner를 순차적으로 학습시키면서 이전 learner들이 어려워 했던 관측치에 가중치를 주자!!

Algorithm 2 Adaboost

Input: Required ensemble size T

Input: Training set $S = \{(x_1, y_1), (x_2, y_2), ..., (x_N, y_N)\}$, where $y_i \in \{-1, +1\}$

Define a uniform distribution $D_1(i)$ over elements of S.

for $t = 1$ to T **do**

Train a model h_t using distribution D_t.

Calculate $\epsilon_t = P_{D_t}(h_t(x) \neq y)$

If $\epsilon_t \geq 0.5$ break

Set $\alpha_t = \frac{1}{2} \ln \left(\frac{1-\epsilon_t}{\epsilon_t} \right)$

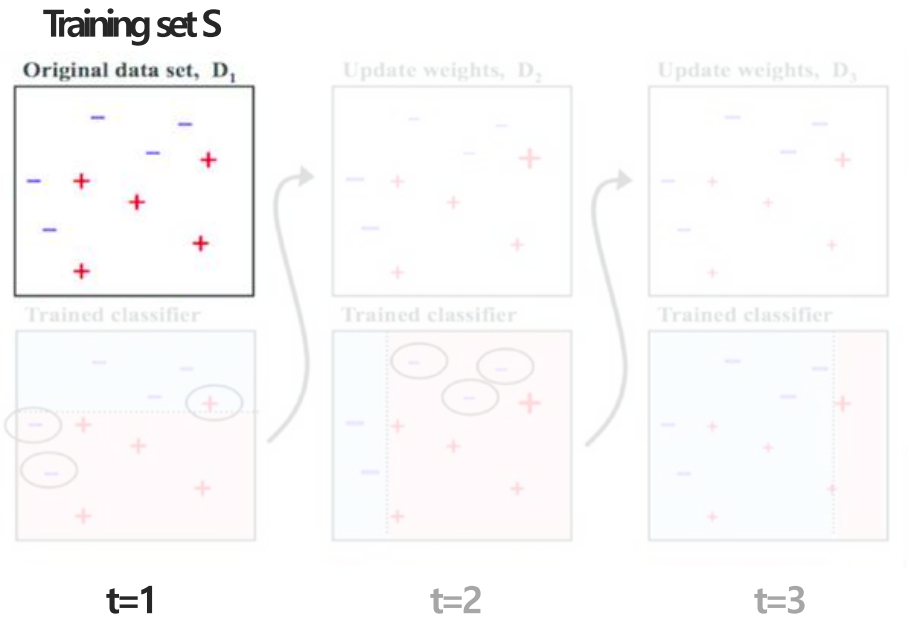
Update $D_{t+1}(i) = \frac{D_t(i) \exp(-\alpha_t y_i h_t(x_i))}{Z_t}$

where Z_t is a normalization factor so that D_{t+1} is a valid distribution.

end for

For a new testing point (x', y'),

$H(x') = sign\left(\sum_{t=1}^{T} \alpha_t h_t(x') \right)$



Boosting based Ensemble: AdaBoost

AdaBoost

❖ AdaBoost Algorithm

- 모델 복잡도가 낮은 weak learner에 적절한 가이드를 주면 strong learner가 될 수 있지 않을까?
- 어떻게 가이드를 줄까? Weak learner를 순차적으로 학습시키면서 이전 learner들이 어려워 했던 관측치에 가중치를 주자!!

Algorithm 2 Adaboost

Input: Required ensemble size T

Input: Training set $S = \{(x_1, y_1), (x_2, y_2), ..., (x_N, y_N)\}$, where $y_i \in \{-1, +1\}$

Define a uniform distribution $D_1(i)$ over elements of S.

for $t = 1$ to T **do**

Train a model h_t using distribution D_t.

Calculate $\epsilon_t = P_{D_t}(h_t(x) \neq y)$

If $\epsilon_t \geq 0.5$ break

Set $\alpha_t = \frac{1}{2} \ln\left(\frac{1-\epsilon_t}{\epsilon_t}\right)$

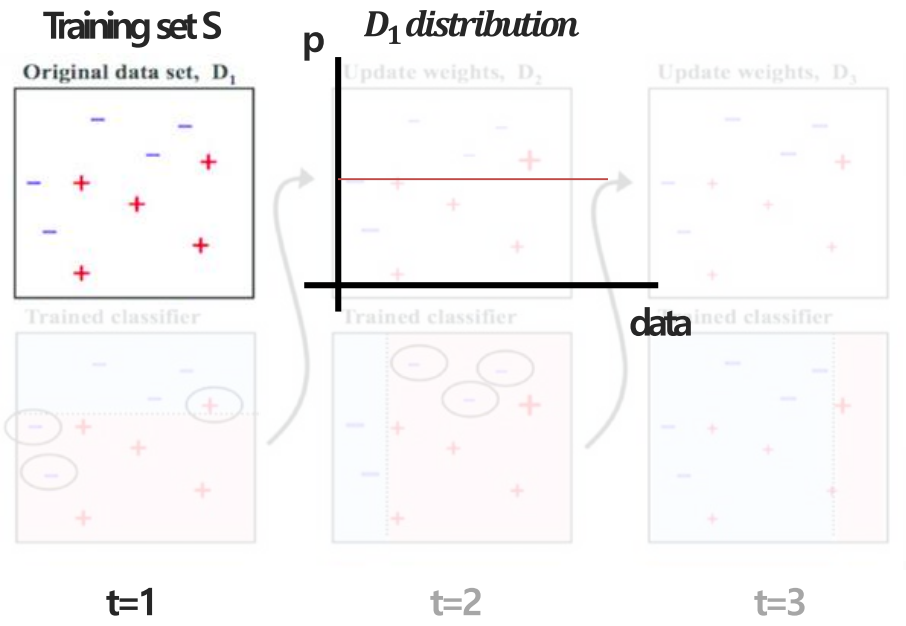
Update $D_{t+1}(i) = \frac{D_t(i) \exp(-\alpha_t y_i h_t(x_i))}{Z_t}$

where Z_t is a normalization factor so that D_{t+1} is a valid distribution.

end for

For a new testing point (x', y'),

$H(x') = sign\left(\sum_{t=1}^{T} \alpha_t h_t(x')\right)$



Boosting based Ensemble: AdaBoost

AdaBoost

❖ AdaBoost Algorithm

- 모델 복잡도가 낮은 weak learner에 적절한 가이드를 주면 strong learner가 될 수 있지 않을까?
- 어떻게 가이드를 줄까? Weak learner를 순차적으로 학습시키면서 이전 learner들이 어려워 했던 관측치에 가중치를 주자!!

Algorithm 2 Adaboost

Input: Required ensemble size T

Input: Training set $S = \{(x_1, y_1), (x_2, y_2), ..., (x_N, y_N)\}$, where $y_i \in \{-1, +1\}$

Define a uniform distribution $D_1(i)$ over elements of S.

for $t = 1$ to T **do**

Train a model h_t using distribution D_t.

Calculate $\epsilon_t = P_{D_t}(h_t(x) \neq y)$

If $\epsilon_t \geq 0.5$ break

Set $\alpha_t = \frac{1}{2} \ln\left(\frac{1-\epsilon_t}{\epsilon_t}\right)$

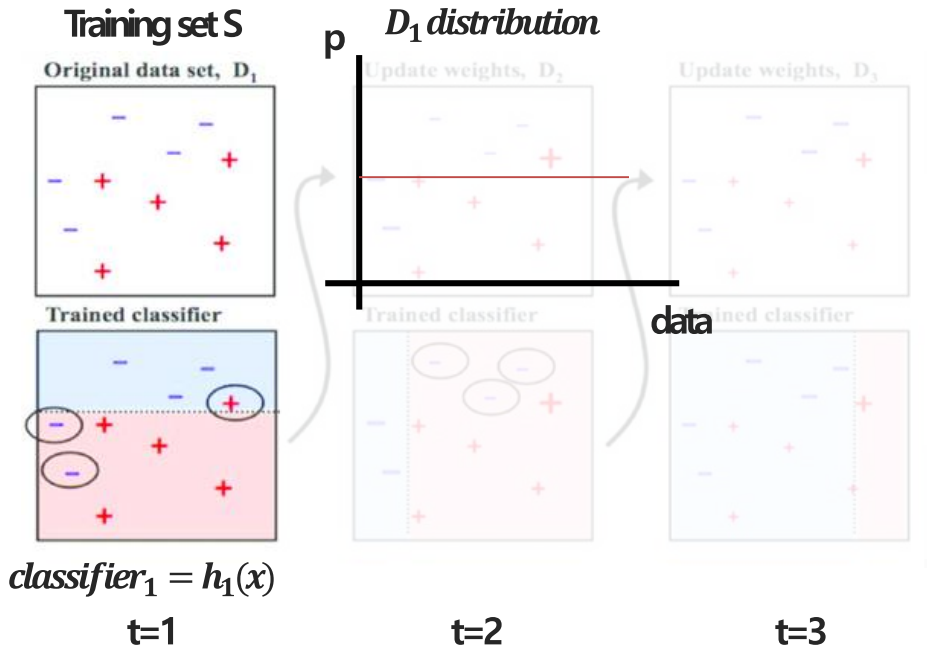
Update $D_{t+1}(i) = \frac{D_t(i) \exp(-\alpha_t y_i h_t(x_i))}{Z_t}$

where Z_t is a normalization factor so that D_{t+1} is a valid distribution.

end for

For a new testing point (x', y'),

$H(x') = sign\left(\sum_{t=1}^{T} \alpha_t h_t(x')\right)$



Boosting based Ensemble: AdaBoost

AdaBoost

❖ AdaBoost Algorithm

- 모델 복잡도가 낮은 weak learner에 적절한 가이드를 주면 strong learner가 될 수 있지 않을까?
- 어떻게 가이드를 줄까? Weak learner를 순차적으로 학습시키면서 이전 learner들이 어려워 했던 관측치에 가중치를 주자!!

Algorithm 2 Adaboost

Input: Required ensemble size T

Input: Training set $S = \{(x_1, y_1), (x_2, y_2), ..., (x_N, y_N)\}$, where $y_i \in \{-1, +1\}$

Define a uniform distribution $D_1(i)$ over elements of S.

for $t = 1$ to T **do**

Train a model h_t using distribution D_t.

Calculate $\epsilon_t = P_{D_t}(h_t(x) \neq y)$

If $\epsilon_t \geq 0.5$ break

Set $\alpha_t = \frac{1}{2} \ln\left(\frac{1-\epsilon_t}{\epsilon_t}\right)$

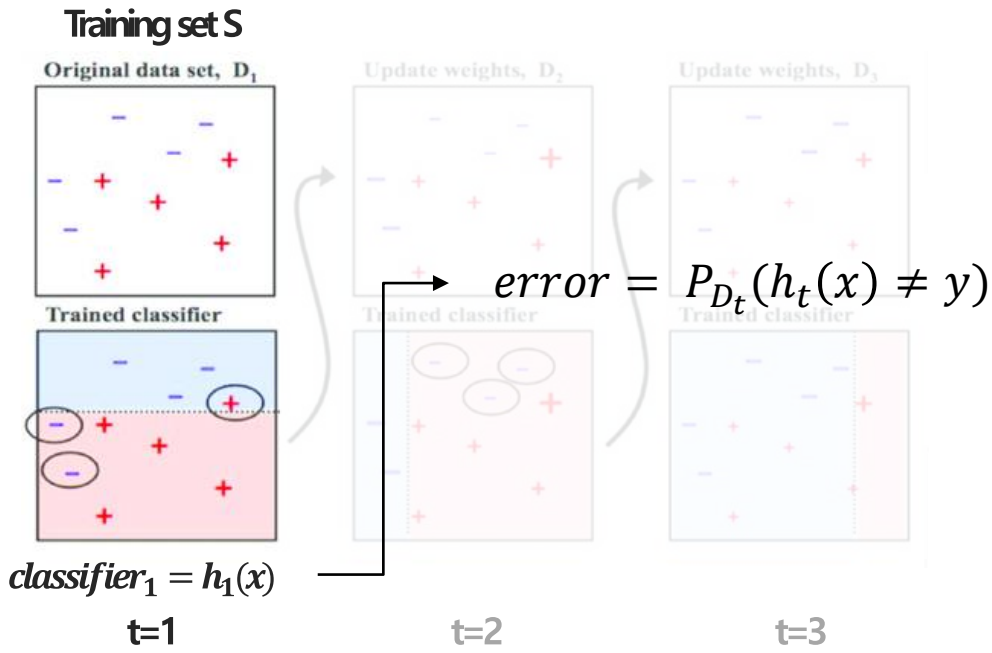
Update $D_{t+1}(i) = \frac{D_t(i)\exp(-\alpha_t y_i h_t(x_i))}{Z_t}$

where Z_t is a normalization factor so that D_{t+1} is a valid distribution.

end for

For a new testing point (x', y'),

$H(x') = sign\left(\sum_{t=1}^{T} \alpha_t h_t(x')\right)$



Boosting based Ensemble: AdaBoost

AdaBoost

❖ AdaBoost Algorithm

- 모델 복잡도가 낮은 weak learner에 적절한 가이드를 주면 strong learner가 될 수 있지 않을까?
- 어떻게 가이드를 줄까? Weak learner를 순차적으로 학습시키면서 이전 learner들이 어려워 했던 관측치에 가중치를 주자!!

Algorithm 2 Adaboost

Input: Required ensemble size T

Input: Training set $S = \{(x_1, y_1), (x_2, y_2), ..., (x_N, y_N)\}$, where $y_i \in \{-1, +1\}$

Define a uniform distribution $D_1(i)$ over elements of S.

for $t = 1$ to T **do**

Train a model h_t using distribution D_t.

Calculate $\epsilon_t = P_{D_t}(h_t(x) \neq y)$

If $\epsilon_t \geq 0.5$ break

Set $\alpha_t = \frac{1}{2} \ln\left(\frac{1-\epsilon_t}{\epsilon_t}\right)$

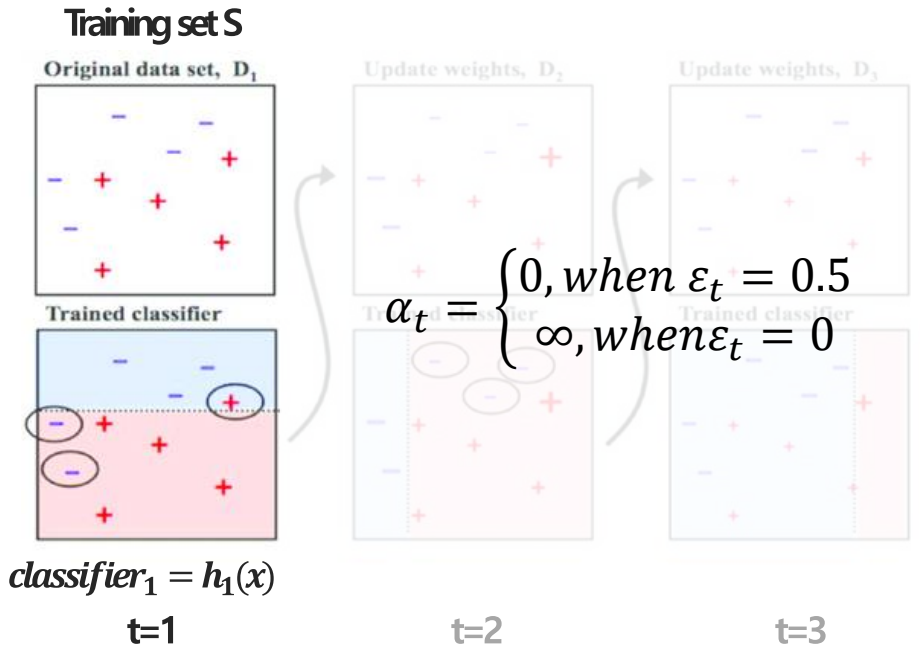
Update $D_{t+1}(i) = \frac{D_t(i) \exp(-\alpha_t y_i h_t(x_i))}{Z_t}$

where Z_t is a normalization factor so that D_{t+1} is a valid distribution.

end for

For a new testing point (x', y'),

$H(x') = sign\left(\sum_{t=1}^{T} \alpha_t h_t(x')\right)$



Boosting based Ensemble: AdaBoost

AdaBoost

❖ AdaBoost Algorithm

- 모델 복잡도가 낮은 weak learner에 적절한 가이드를 주면 strong learner가 될 수 있지 않을까?
- 어떻게 가이드를 줄까? Weak learner를 순차적으로 학습시키면서 이전 learner들이 어려워 했던 관측치에 가중치를 주자!!

Algorithm 2 Adaboost

Input: Required ensemble size T

Input: Training set $S = \{(x_1, y_1), (x_2, y_2), ..., (x_N, y_N)\}$, where $y_i \in \{-1, +1\}$

Define a uniform distribution $D_1(i)$ over elements of S.

for $t = 1$ to T **do**

Train a model h_t using distribution D_t.

Calculate $\epsilon_t = P_{D_t}(h_t(x) \neq y)$

If $\epsilon_t \geq 0.5$ break

Set $\alpha_t = \frac{1}{2} \ln\left(\frac{1-\epsilon_t}{\epsilon_t}\right)$

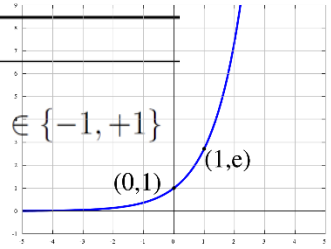
Update $D_{t+1}(i) = \frac{D_t(i) \exp(-\alpha_t y_i h_t(x_i))}{Z_t}$

where Z_t is a normalization factor so that D_{t+1} is a valid distribution.

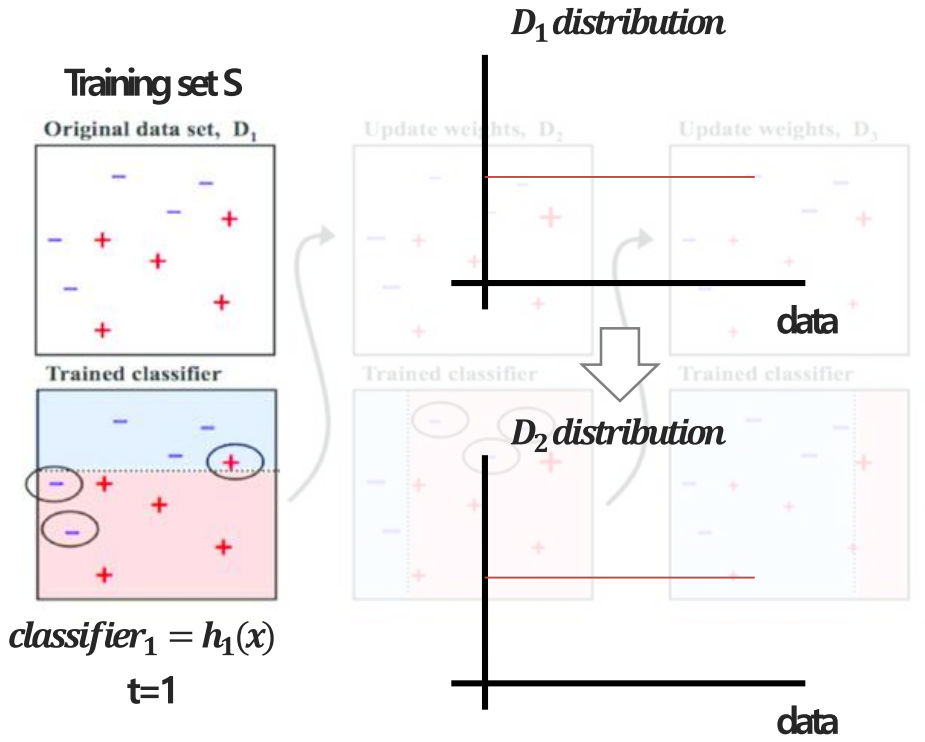
end for

For a new testing point (x', y'),

$H(x') = sign\left(\sum_{t=1}^{T} \alpha_t h_t(x')\right)$



$$D_{t+1}(i) = \frac{D_t(i) \exp\left(-\alpha_t y_i h_t(x_i)\right)}{Z_t}$$



Boosting based Ensemble: AdaBoost

AdaBoost

❖ AdaBoost Algorithm

- 모델 복잡도가 낮은 weak learner에 적절한 가이드를 주면 strong learner가 될 수 있지 않을까?
- 어떻게 가이드를 줄까? Weak learner를 순차적으로 학습시키면서 이전 learner들이 어려워 했던 관측치에 가중치를 주자!!

Algorithm 2 Adaboost

Input: Required ensemble size T

Input: Training set $S = \{(x_1, y_1), (x_2, y_2), ..., (x_N, y_N)\}$, where $y_i \in \{-1, +1\}$

Define a uniform distribution $D_1(i)$ over elements of S.

for $t = 1$ to T **do**

Train a model h_t using distribution D_t.

Calculate $\epsilon_t = P_{D_t}(h_t(x) \neq y)$

If $\epsilon_t \geq 0.5$ break

Set $\alpha_t = \frac{1}{2} \ln \left(\frac{1-\epsilon_t}{\epsilon_t} \right)$

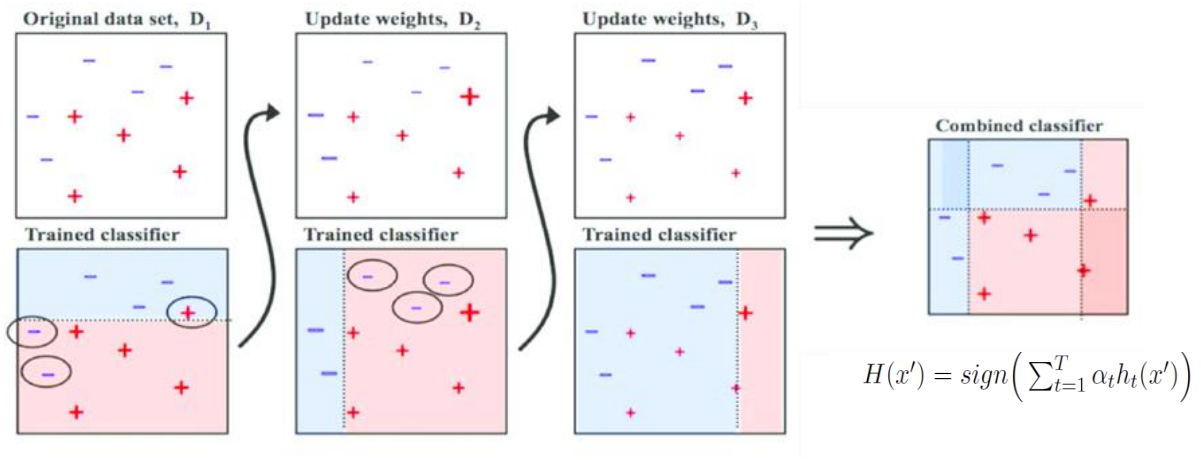
Update $D_{t+1}(i) = \frac{D_t(i) \exp(-\alpha_t y_i h_t(x_i))}{Z_t}$

where Z_t is a normalization factor so that D_{t+1} is a valid distribution.

end for

For a new testing point (x', y'),

$H(x') = sign\left(\sum_{t=1}^{T} \alpha_t h_t(x') \right)$



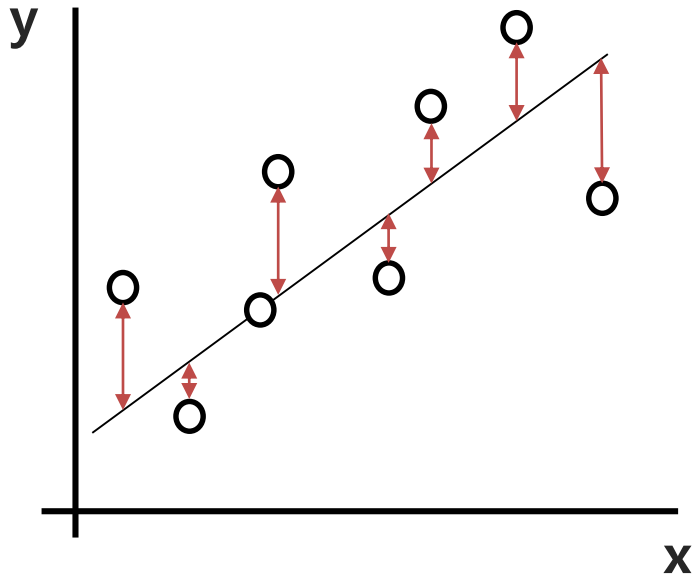
Boosting based Ensemble: Gradient Boosting Machine

GMB

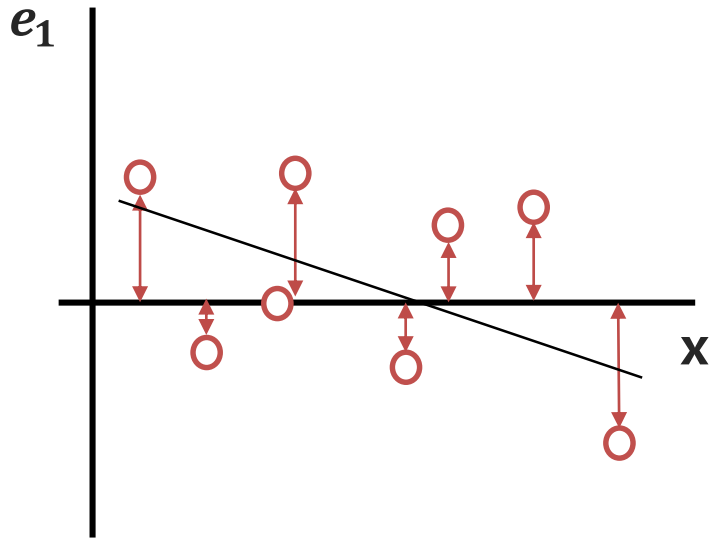
❖ Gradient Boosting Idea

- 이전 모델의 에러를 표현할 수 있는 모델을 학습하면 성능이 좋아지지 않을까?
- 못 맞췄던 만큼만 다음 모델이 맞출 수 있게 하자!

$\hat{y} = f_1(x)$



$y - f_1(x) = f_2(x)$



Boosting based Ensemble: Gradient Boosting Machine

GMB

❖ Why "**Gradient**" boosting?

- Residual을 손실함수의 미분값으로 표현할 수 있음

$$Loss\ function\ L = \frac{1}{2}\sum_{i=1}^{n}\left(y_i - f(x_i)\right)^2$$

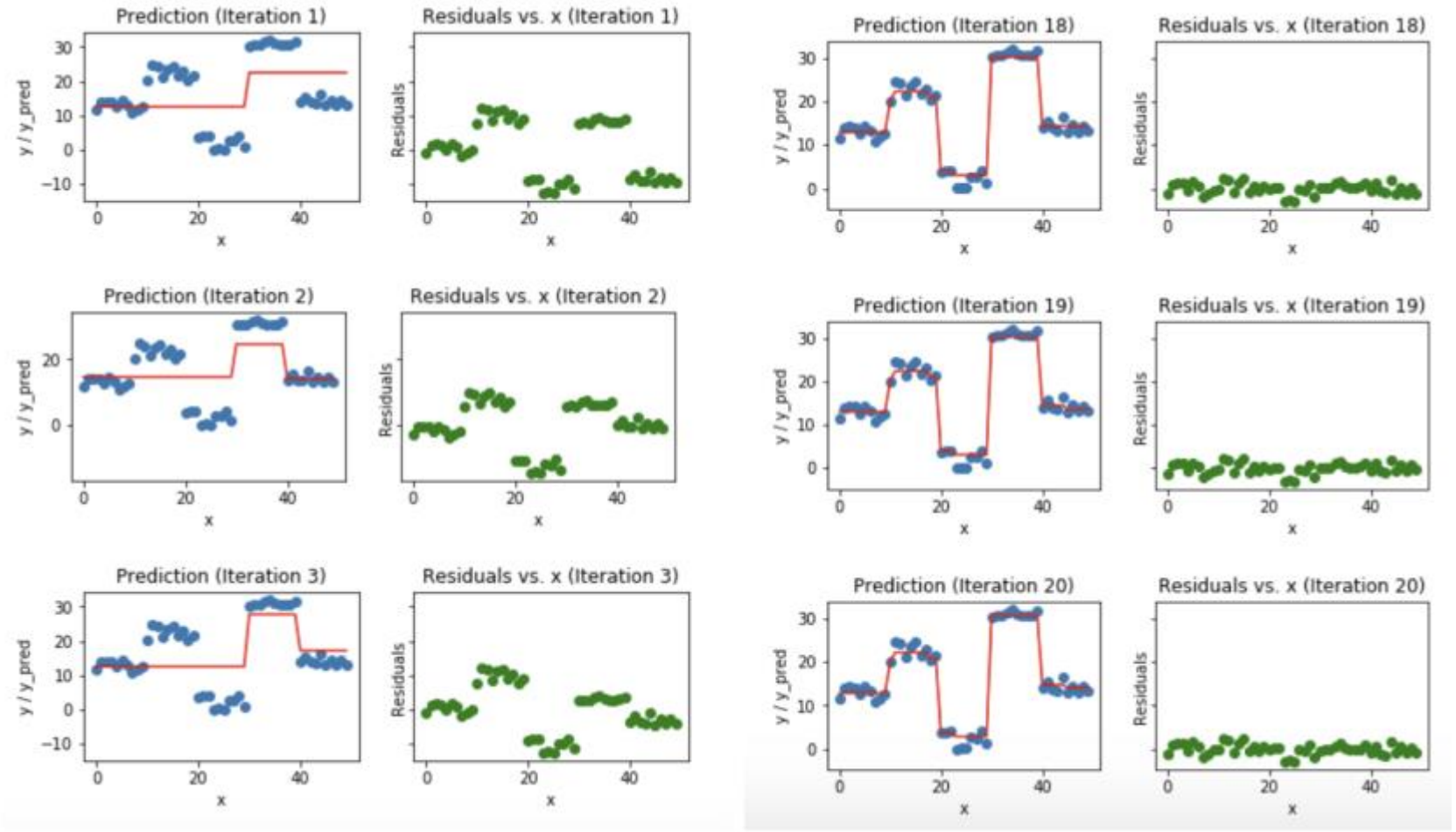
$$gradient\ of\ L \rightarrow \frac{\partial L}{\partial f(x_i)} = f(x_i) - y_i$$

$$residual = y_i - f(x_i) = -(f(x_i) - y_i) = -\frac{\partial L}{\partial f(x_i)}$$

Boosting based Ensemble: Gradient Boosting Machine

GMB

❖ Example of GBM



감사합니다