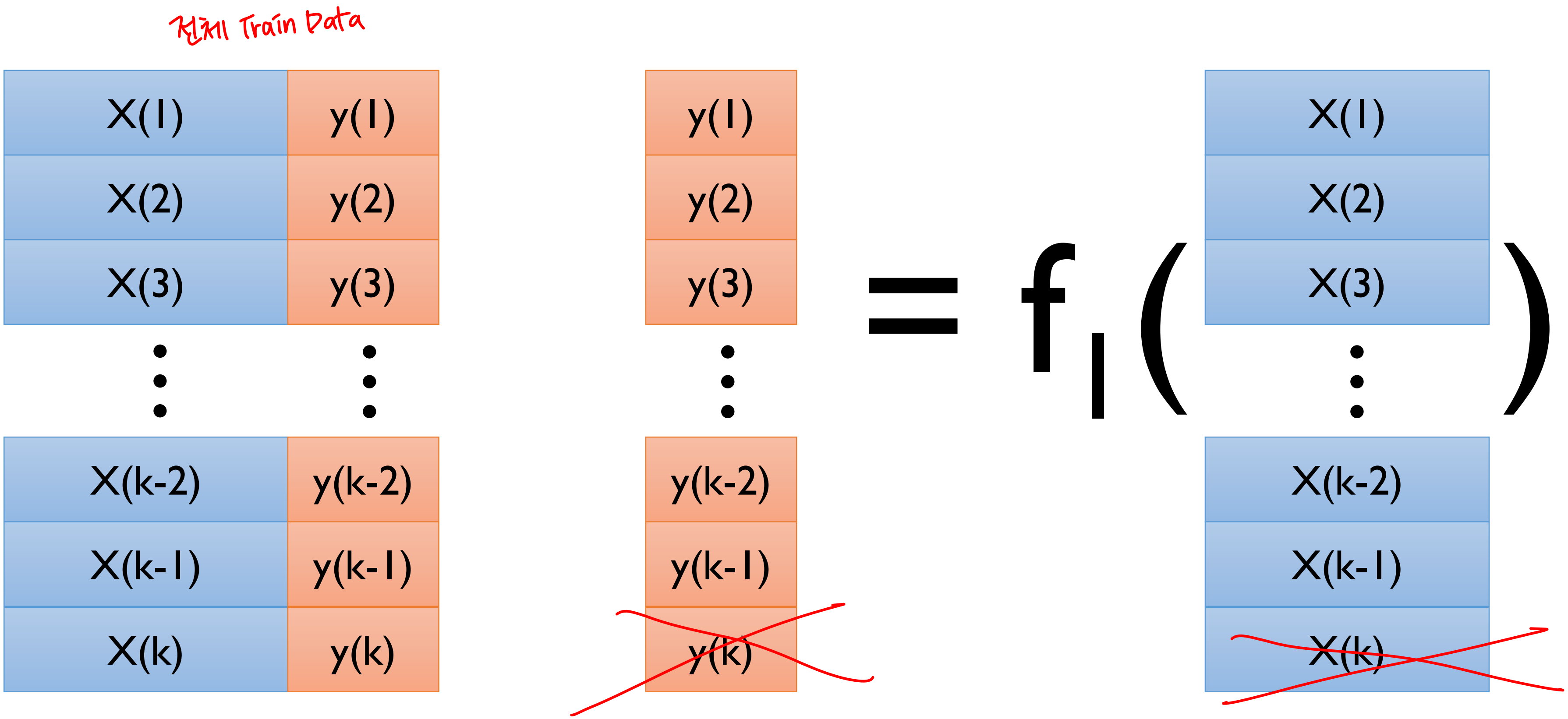
Lecture 7-2: Ensemble Learning
Bagging

Pilsung Kang
School of Industrial Management Engineering
Korea University

Sampling without Replacement

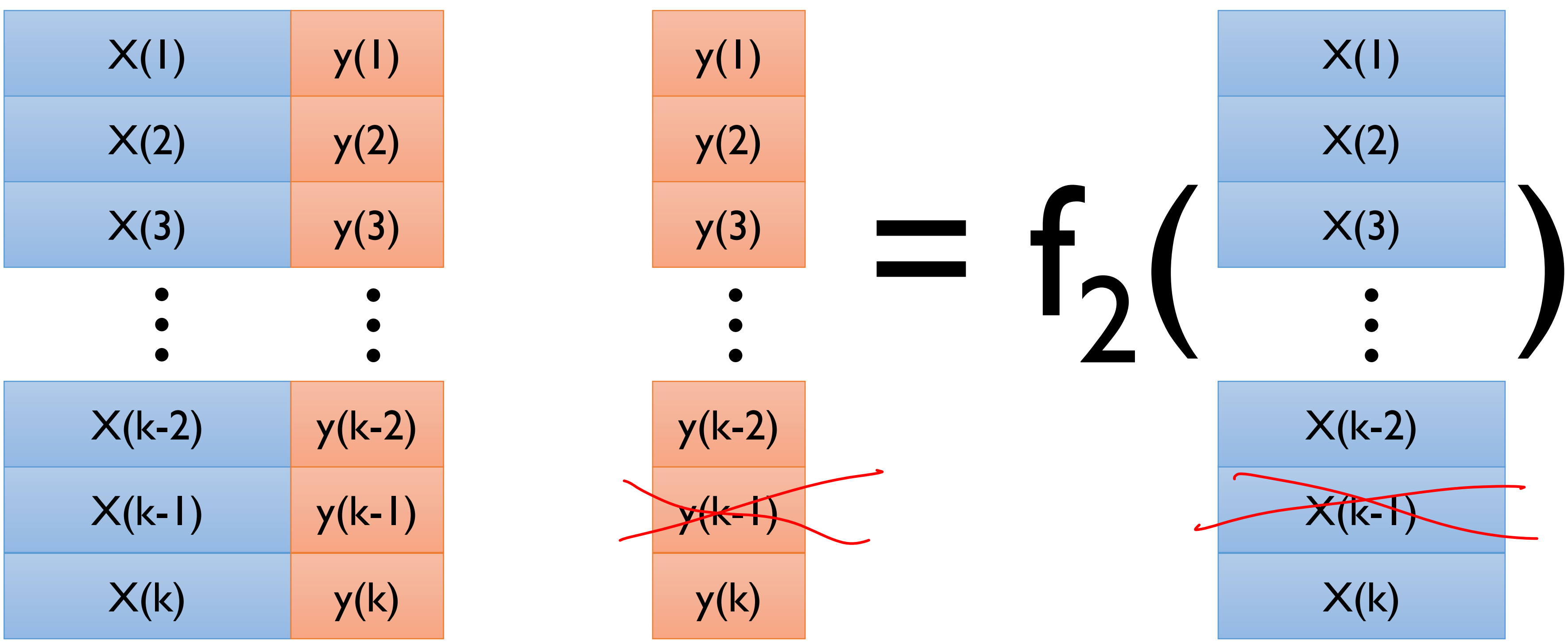
앙상블 : Diversity Data Bagging! Model

- K-fold data split
  - ✓ Entire data is split into k blocks; each classifier is trained only on different subset of (k-1) blocks



Sampling without Replacement

- K-fold data split
  - ✓ Entire data is split into k blocks; each classifier is trained only on different subset of (k-1) blocks



Sampling without Replacement

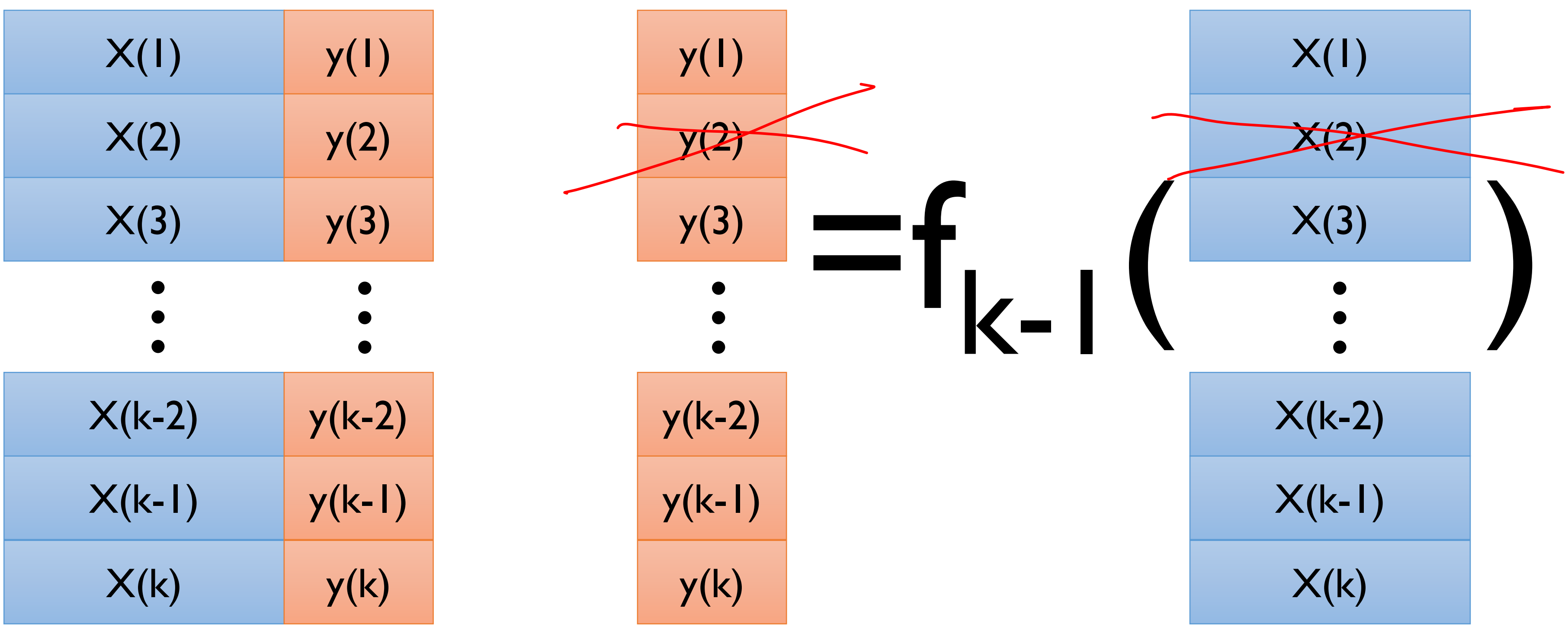
- K-fold data split
  - ✓ Entire data is split into k blocks; each classifier is trained only on different subset of (k-1) blocks

X(1)	y(1)
X(2)	y(2)
X(3)	y(3)
⋮	⋮
X(k-2)	y(k-2)
X(k-1)	y(k-1)
X(k)	y(k)

$$\begin{array}{c} y(1)\\ y(2)\\ y(3)\\ \vdots\\ \cancel{y(k-2)}\\ y(k-1)\\ y(k)\end{array} = f_3\left(\begin{array}{c} X(1)\\ X(2)\\ X(3)\\ \vdots\\ \cancel{X(k-2)}\\ X(k-1)\\ X(k)\end{array}\right)$$

Sampling without Replacement

- K-fold data split
  - ✓ Entire data is split into k blocks; each classifier is trained only on different subset of (k-1) blocks



Sampling without Replacement

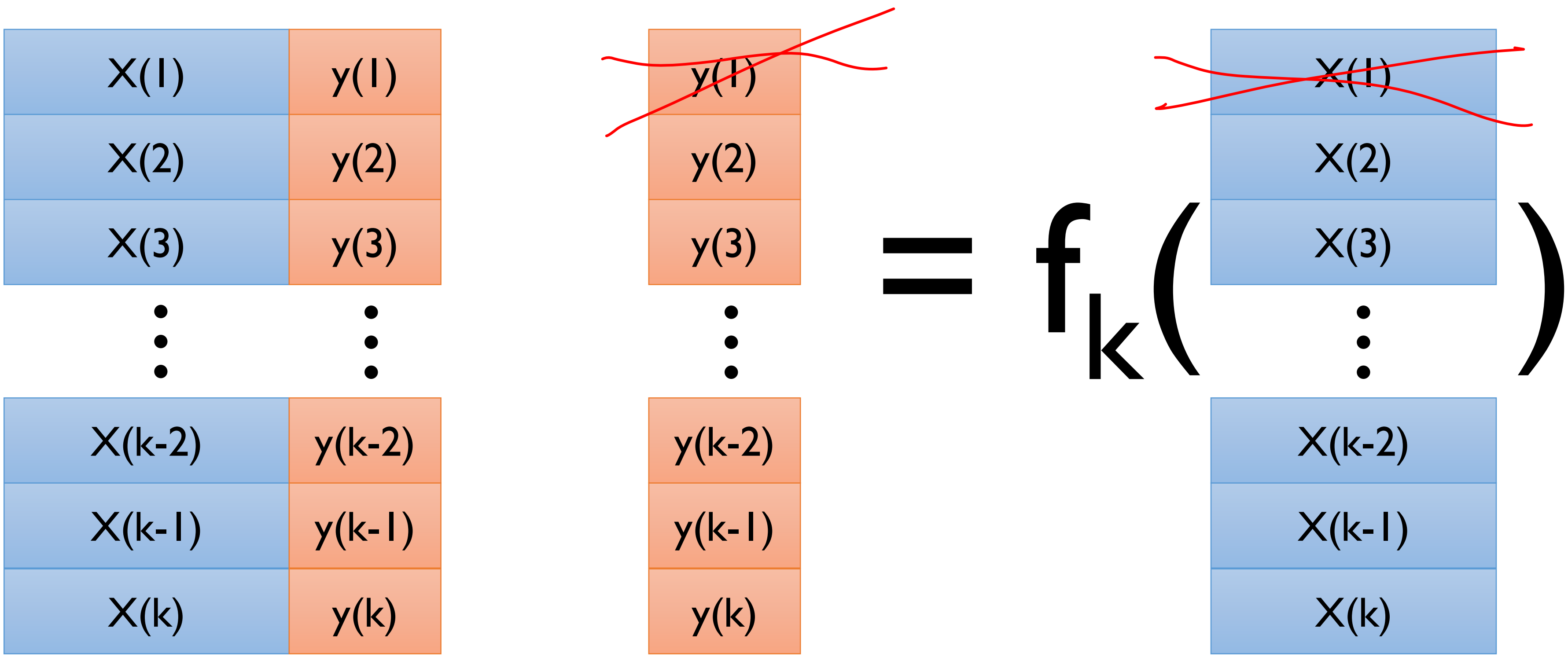
「모든 모델이 독립이라고하면 앙상블이 $\frac{1}{k}$ 만큼 줄어든다고 했음」 이상적인 성능

$f_1 \neq f_2 \neq \cdots \neq f_k$

f_i와 f_j를 선택 (k-2) folds는 공통으로 가지고 있음.

∴ 독립성이 보장이 안됨.

- K-fold data split
  - ✓ Entire data is split into k blocks; each classifier is trained only on different subset of (k-1) blocks



Sampling without Replacement

- K-fold data split
  - ✓ Entire data is split into k blocks; each classifier is trained only on different subset of (k-1) blocks
- Final output

delta
; aggeregating 함수

$$\hat{y} = \delta\Big(f_1(\mathbf{x}), f_2(\mathbf{x}), \cdots, f_{k-1}(\mathbf{x}), f_k(\mathbf{x})\Big)$$

  - ✓ $\delta(\cdot)$: An aggregation function of individual outputs (ex: simple average)

→ 잘 안쓰인다...

Bootstrap Aggregating: Bagging

Breiman (1996)

- Main Idea
  - ✓ Each member of the ensemble is constructed from a different training dataset
  - ✓ Each dataset is generated by sampling from the total N data examples, choosing N items uniformly at random with replacement 복원추출!
  - ✓ Each dataset sample is known as a bootstrap ⇒ 데이터의 분포를 왜곡시킴 (긍정적으로)

$y = f(x) + \epsilon$

ε이 가질수 있는 범위나 분포를 바꿔줌.
데이터를 왜곡시킴으로서 하나의 노이즈에 종속적일 위험을 방지할 수 있음.

우리가 원하는 만큼 무한정으로 만들수있음

Original Dataset

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
x^8	y^8
x^9	y^9
x^{10}	y^{10}

Bootstrap 1

x^3	y^3
x^6	y^6
x^2	y^2
x^{10}	y^{10}
x^8	y^8
x^7	y^7
x^7	y^7
x^3	y^3
x^2	y^2
x^7	y^7

복원 (

x^1, x^4, x^5, x^9 는 sampling 안됨

Bootstrap 2

x^7	y^7
x^1	y^1
x^{10}	y^{10}
x^1	y^1
x^8	y^8
x^6	y^6
x^2	y^2
x^6	y^6
x^4	y^4
x^9	y^9

⇒ 복원추출이기에 sampling 안되는 데이터가 있음

• • •

Bootstrap B

x^9	y^9
x^5	y^5
x^2	y^2
x^4	y^4
x^7	y^7
x^2	y^2
x^5	y^5
x^{10}	y^{10}
x^8	y^8
x^2	y^2

Bootstrap Aggregating : Bagging

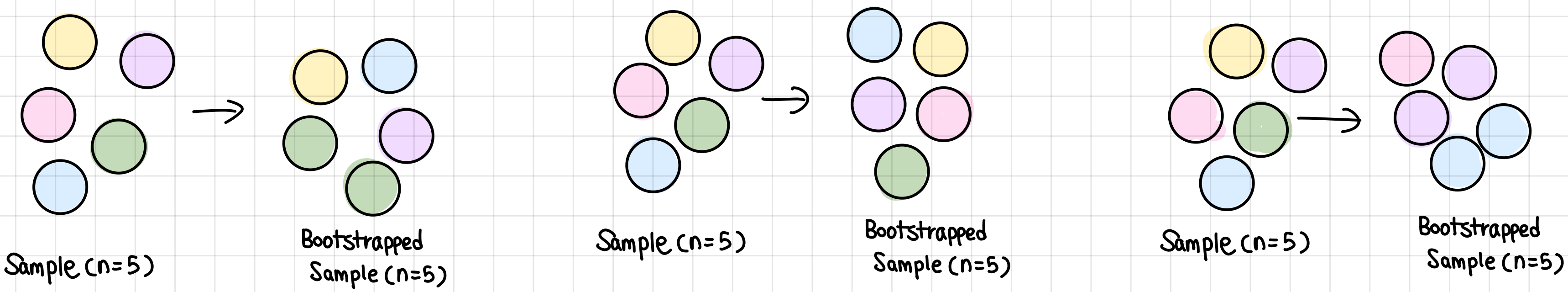
· Bagging : Bootstrap Aggregating

√ Probability that an instance is not included in a bootstrap ⇒

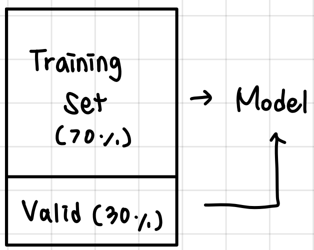
$$P = \left(1-\frac{1}{N}\right)^N \rightarrow \lim_{N\to\infty}\left(1-\frac{1}{N}\right)^N = e^{-1} = 0.368$$

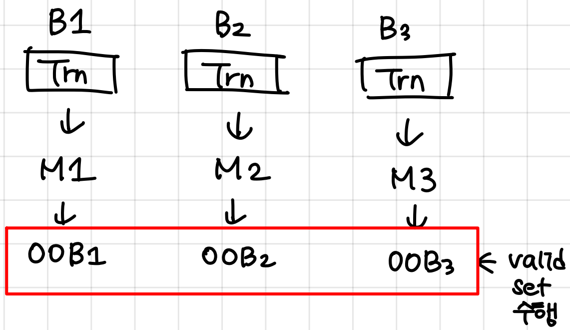
⇒ 선택되지 않을 확률

N이 일정수준 이상으로 큰 수이면
2/3는 Bootstrap에 1회이상 샘플링
1/3는 Bootstrap에 0회 샘플링
Out of Bag (OOB)



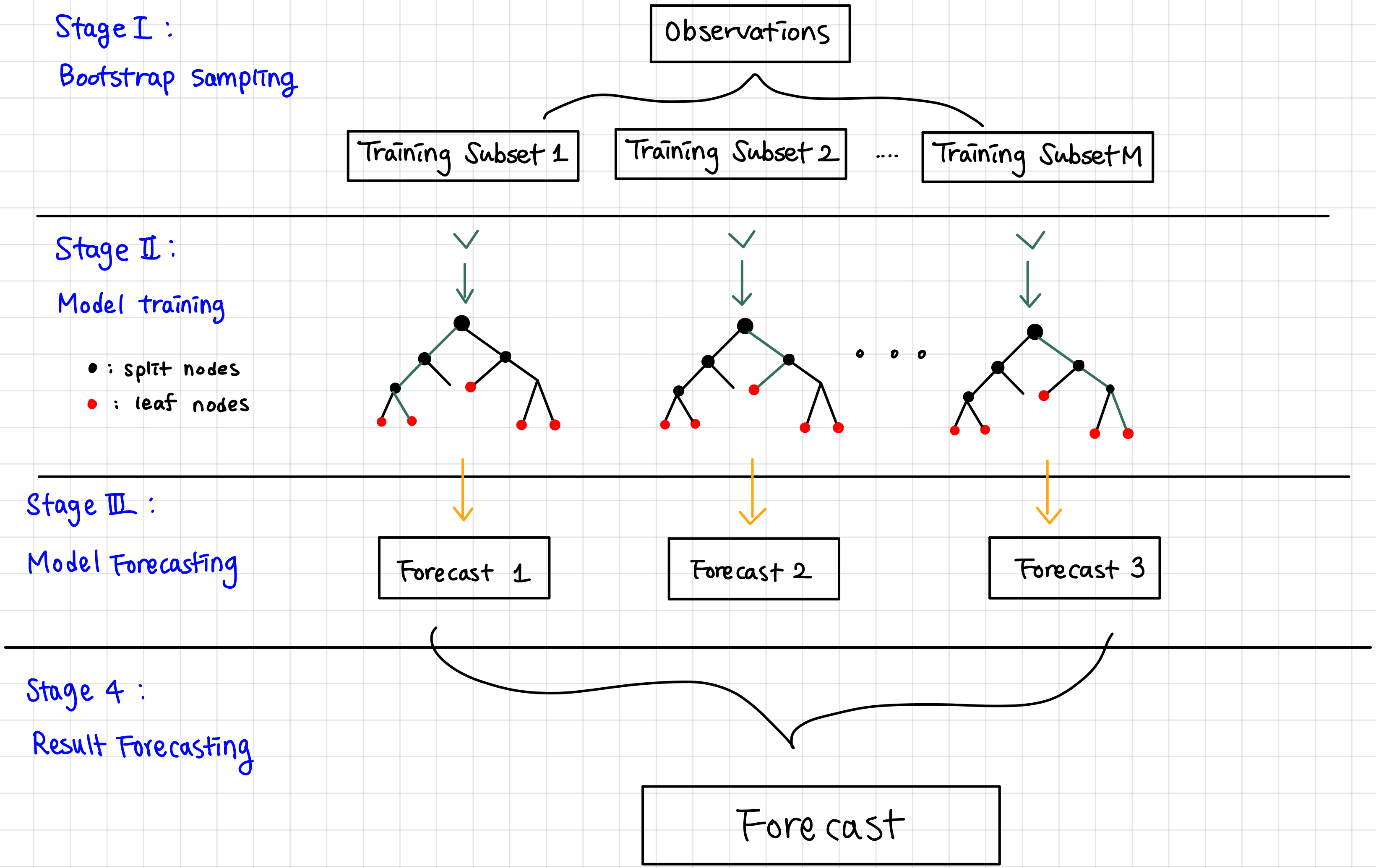
√ Fits well with the models with low bias and high variance
→ complex model
ex. DCT, SVM ...





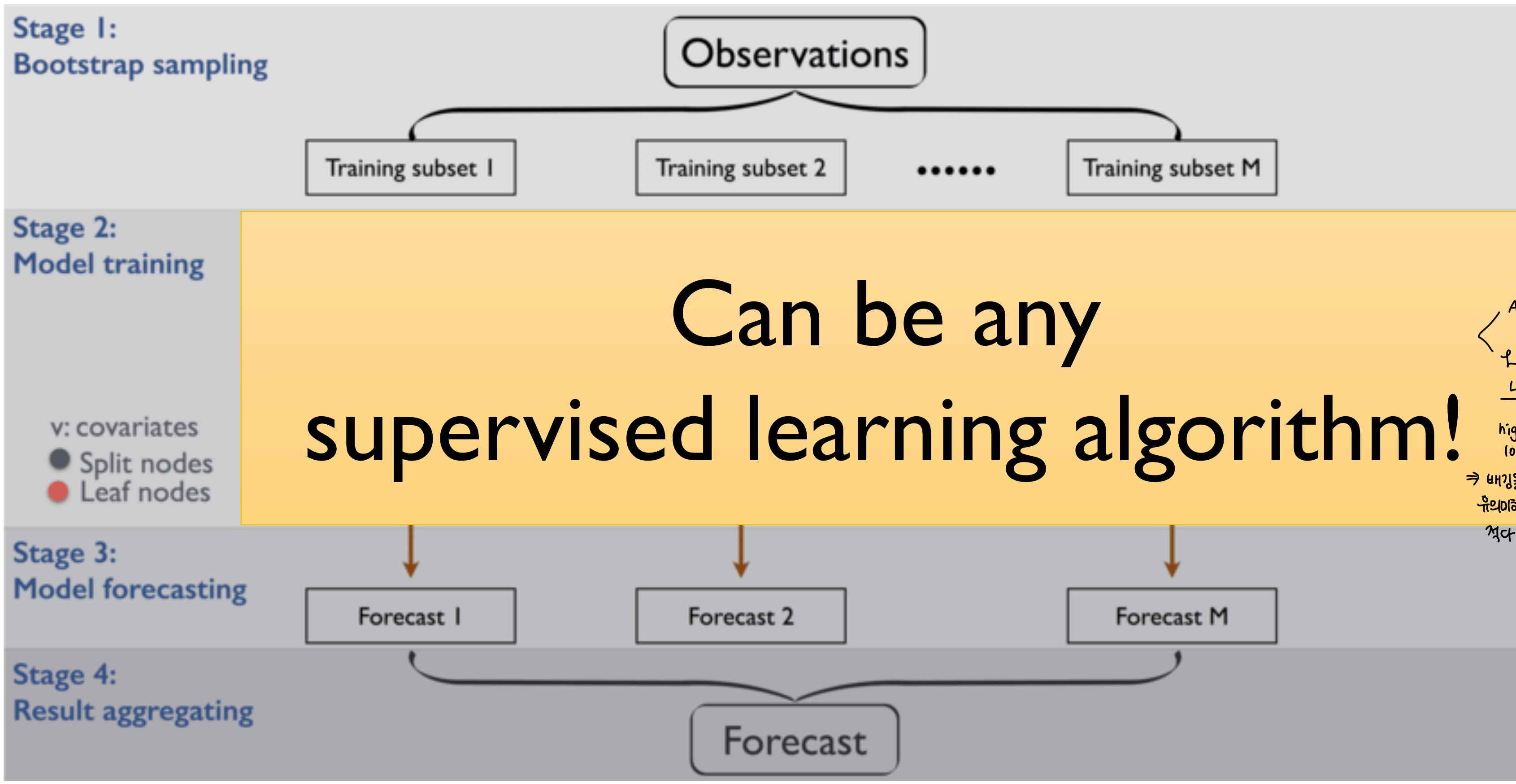
Bootstrap Aggregating : Bagging

- Bagging with Decision Tree (≠ Random Forest)



Bootstrap Aggregating: Bagging

- Bagging with Decision Tree



ANN, SVM, DT

Log, Reg
LDA

high bias
low variance

⇒ 배깅을 통해 성능이 유의미하게 향상될 학률이 적다.

Bootstrap Aggregating: Bagging

- Result Aggregating
  - ✓ For classification problem
    - Majority voting

Xnew

$$\hat{y}_{Ensemble} = arg\max_i \left(\sum_{j=1}^{n} \delta(\hat{y}_j = i), \quad i \in \{0, 1\} \right)$$

Out of Bag (OOB) ~~Training~~ Accuracy	Ensemble population	P(y=1) for a test instance	Predicted class label
0.80	Model 1	0.90	1
0.75	Model 2	0.92	1
0.88	Model 3	0.87	1
0.91	Model 4	0.34	0
0.77	Model 5	0.41	0
0.65	Model 6	0.84	1
0.95	Model 7	0.14	0
0.82	Model 8	0.32	0
0.78	Model 9	0.98	1
0.83	Model 10	0.57	1

$$\sum_{j=1}^{n} \delta(\hat{y}_j = 0) = 4$$

$$\sum_{j=1}^{n} \delta(\hat{y}_j = 1) = 6$$

$$\hat{y}_{Ensemble} = 1$$

Bootstrap Aggregating: Bagging

- Result Aggregating
  - ✓ For classification problem
    - ▪ Weighted voting (weight = training accuracy of individual models)

1번범주에 해당할 accuracy

Out of Bag
$$\hat{y}_{Ensemble} = \arg\max_i \left(\frac{\sum_{j=1}^{n} (\overset{OOB}{\cancel{Trn}}Acc_j) \cdot \delta(\hat{y}_j = i)}{\sum_{j=1}^{n} (\underset{OOB}{\cancel{Trn}}Acc_j)}, \quad i \in \{0, 1\} \right)$$

OOB ~~Training~~ Accuracy	Ensemble population	P(y=1) for a test instance	Predicted class label
0.80	Model 1	0.90	1
0.75	Model 2	0.92	1
0.88	Model 3	0.87	1
0.91	Model 4	0.34	0
0.77	Model 5	0.41	0
0.65	Model 6	0.84	1
0.95	Model 7	0.14	0
0.82	Model 8	0.32	0
0.78	Model 9	0.98	1
0.83	Model 10	0.57	1

$\Sigma = A$

$P(Y=0 \mid X_{new}) = \frac{0.91+0.77+0.95+0.82}{A}$

$$\frac{\sum_{j=1}^{n} (TrnAcc_j) \cdot \delta(\hat{y}_j = 0)}{\sum_{j=1}^{n} (\underset{OOB}{\cancel{Trn}}Acc_j)} = 0.424$$

$$\frac{\sum_{j=1}^{n} (TrnAcc_j) \cdot \delta(\hat{y}_j = 1)}{\sum_{j=1}^{n} (\underset{OOB}{\cancel{Trn}}Acc_j)} = 0.576$$

$P(Y=1 \mid X_{new}) = \frac{0.80+0.75+0.88+0.65+0.78+0.83}{A}$

$$\hat{y}_{Ensemble} = 1$$

Bootstrap Aggregating: Bagging

- Result Aggregating
  - ✓ For classification problem
    - ▪ Weighted voting (weight = predicted probability for each class)

$$\hat{y}_{Ensemble} = arg\max_i \left(\frac{1}{n} \sum_{j=1}^{n} P(y=i), \quad i \in \{0,1\} \right)$$

Training Accuracy	Ensemble population	P(y=1) for a test instance	Predicted class label
0.80	Model 1	0.90	1
0.75	Model 2	0.92	1
0.88	Model 3	0.87	1
0.91	Model 4	0.34	0
0.77	Model 5	0.41	0
0.65	Model 6	0.84	1
0.95	Model 7	0.14	0
0.82	Model 8	0.32	0
0.78	Model 9	0.98 강하게 판정	1 ⇒ 그렇다면 모델10보다
0.83	Model 10	0.57 약하게 판정	1 모델9가 더 영향력을 끼쳐야함.

$$\frac{1}{n} \sum_{j=1}^{n} P(y=0) = 0.375$$

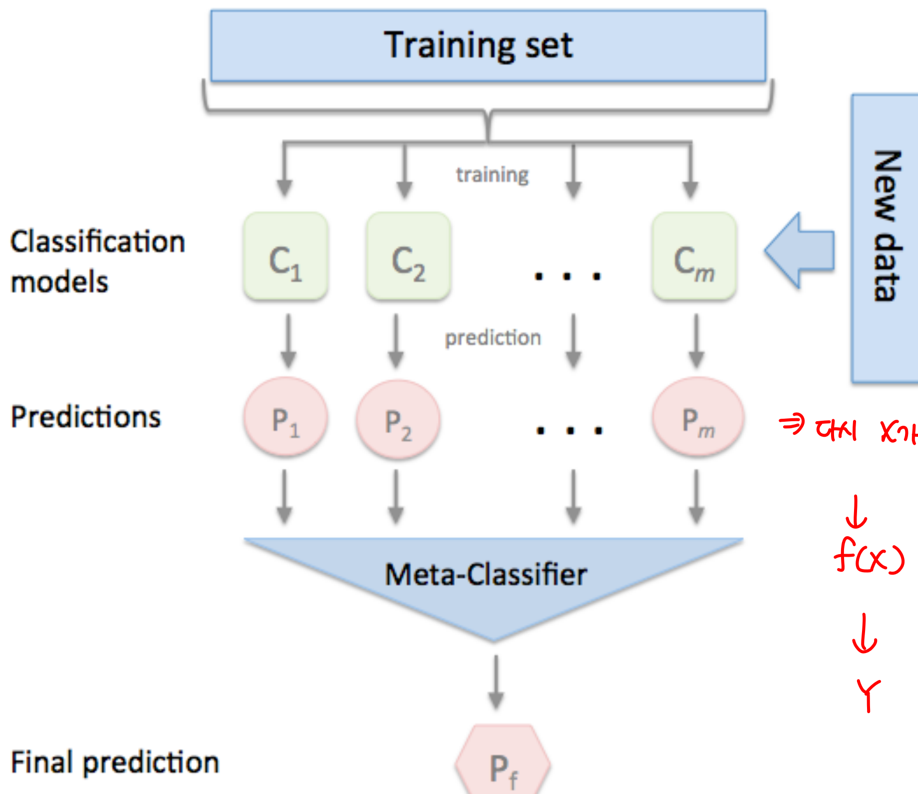
$$\frac{1}{n} \sum_{j=1}^{n} P(y=1) = 0.625$$

$$\hat{y}_{Ensemble} = 1$$

Bootstrap Aggregating: Bagging

부트스트랩 $\begin{cases} X \to f_1(x) \\ X \to f_2(x) \\ \vdots \quad f_B(x) \end{cases}$ $g(f_1(x), \cdots f_B(x)) = y$ "meta learner"

- Result Aggregating: Stacking ★★★
  - ✓ Use another prediction model to aggregate the results
    - ▪ Input: Predictions made by ensemble members
    - ▪ Target: Actual true label



⇒ 다시 X가 됨.

↓

$f(x)$

↓

Y

https://rasbt.github.io/mlxtend/user_guide/classifier/StackingClassifier/

Bootstrap Aggregating: Bagging

- Result Aggregating: Stacking
  - ✓ The winner of KDD-cup 2015
    - MOOC dropout prediction

•Jeong-Yoon Lee, Winning Data Science Competitions

Bootstrap Aggregating: Bagging

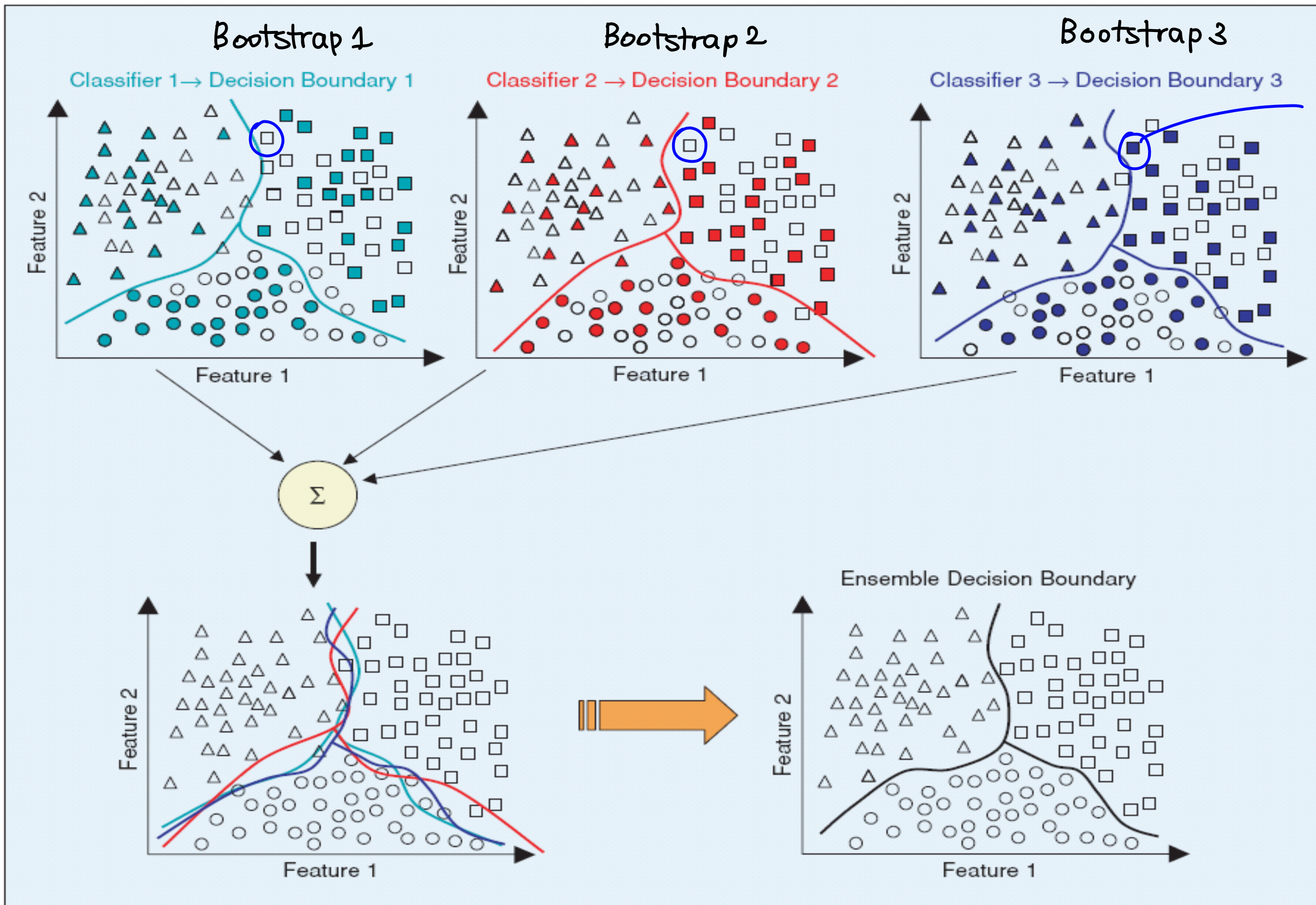
- Bagging: Algorithm

Algorithm 1 Bagging

Input: Required ensemble size T
Input: Training set $S = \{(x_1, y_1), (x_2, y_2), ..., (x_N, y_N)\}$
for $t = 1$ to T **do**
  Build a dataset S_t, by sampling N items, randomly *with replacement* from S.
  Train a model h_t using S_t, and add it to the ensemble.
end for
For a new testing point (x', y'),
If model outputs are continuous, combine them by averaging.
If model outputs are class labels, combine them by voting.

Bootstrap Aggregating: Bagging

- Bagging: Illustration



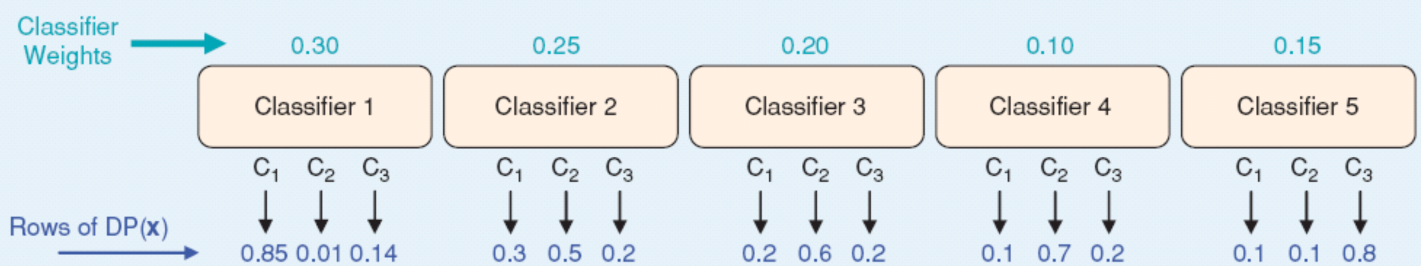
Bootstrap 3
에서만 샘플링됨
(train으로 쓰임)
B1, B2 에서는
아무래도 validset

Figure 3. Combining classifiers that are trained on different subsets of the training data.

(참고)

Bootstrap Aggregating: Bagging

- Aggregation examples 결합예시

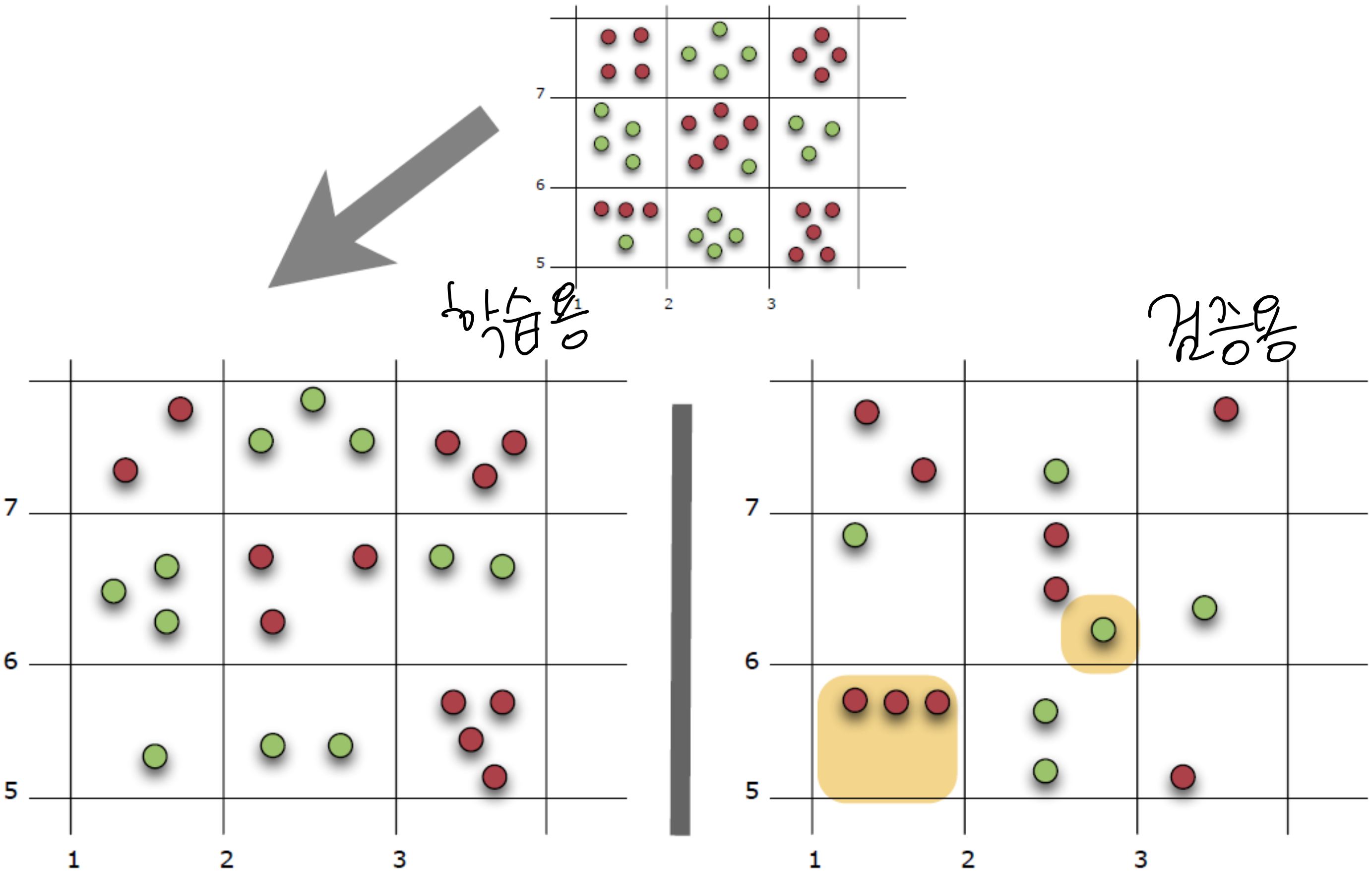


Product Rule:	**C_1: 0.0179,**	C_2: 0.00021,	C_3: 0.0009
Sum Rule:	C_1: 1.55,	**C_2: 1.91,**	C_3: 1.54
Mean Rule:	C_1: 0.310,	**C_2: 0.382,**	C_3: 0.308
Max Rule:	**C_1: 0.85,**	C_2: 0.7,	C_3: 0.8
Median Rule:	C_1: 0.2,	**C_2: 0.5,**	C_3: 0.2
Minimum Rule:	C_1: 0.1,	C_2: 0.01,	**C_3: 0.14**
Weighted Average	**C_1: 0.395,**	C_2: 0.333,	C_3: 0.272
Majority Voting:	C_1: 1,	**C_2: 3,**	C_3: 1 Vote
Weighted Majority Voting:	C_1: 0.3,	**C_2: 0.55,**	C_3: 0.15 Votes
Borda Count:	C_1: 5	**C_2: 6,**	C_3: 4 Votes

Figure 15. Example on various combination rules.

Bootstrap Aggregating: Bagging

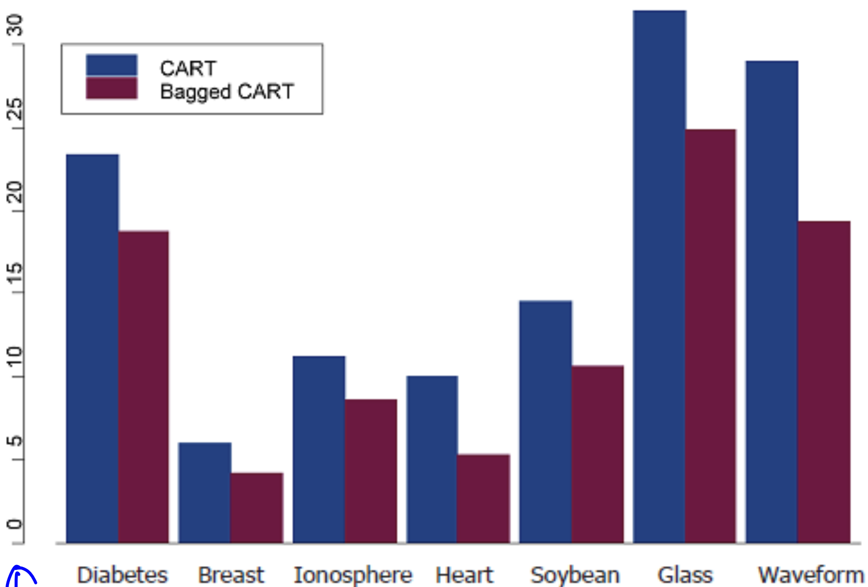
- Out of bag error (OOB Error)
  - ✓ Use the training instances that are not sampled for validation



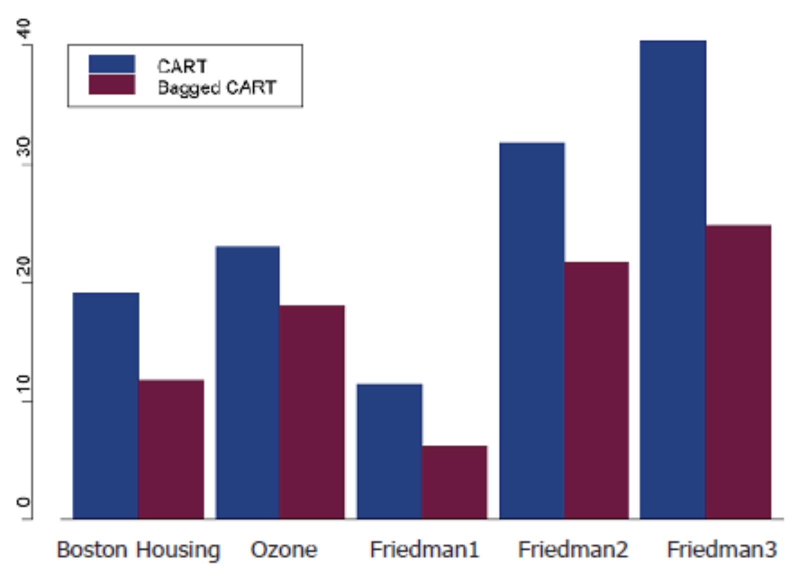
Bootstrap Aggregating: Bagging

- Bagged Trees vs. Single Tree

Classification



Regression



Error율
낮을수록 좋음


Q & A