분석 3주차 세션

15기 분석 김동현/ 15기 분석 이윤정

CONTENTS

01 CV

Test error → **Model assessment & selection**

데이터 셋 크기가 충분히 큰 경우

- Train & Test set 으로 분리 후, Test error를 추정

데이터 셋 크기가 크지 않은 경우

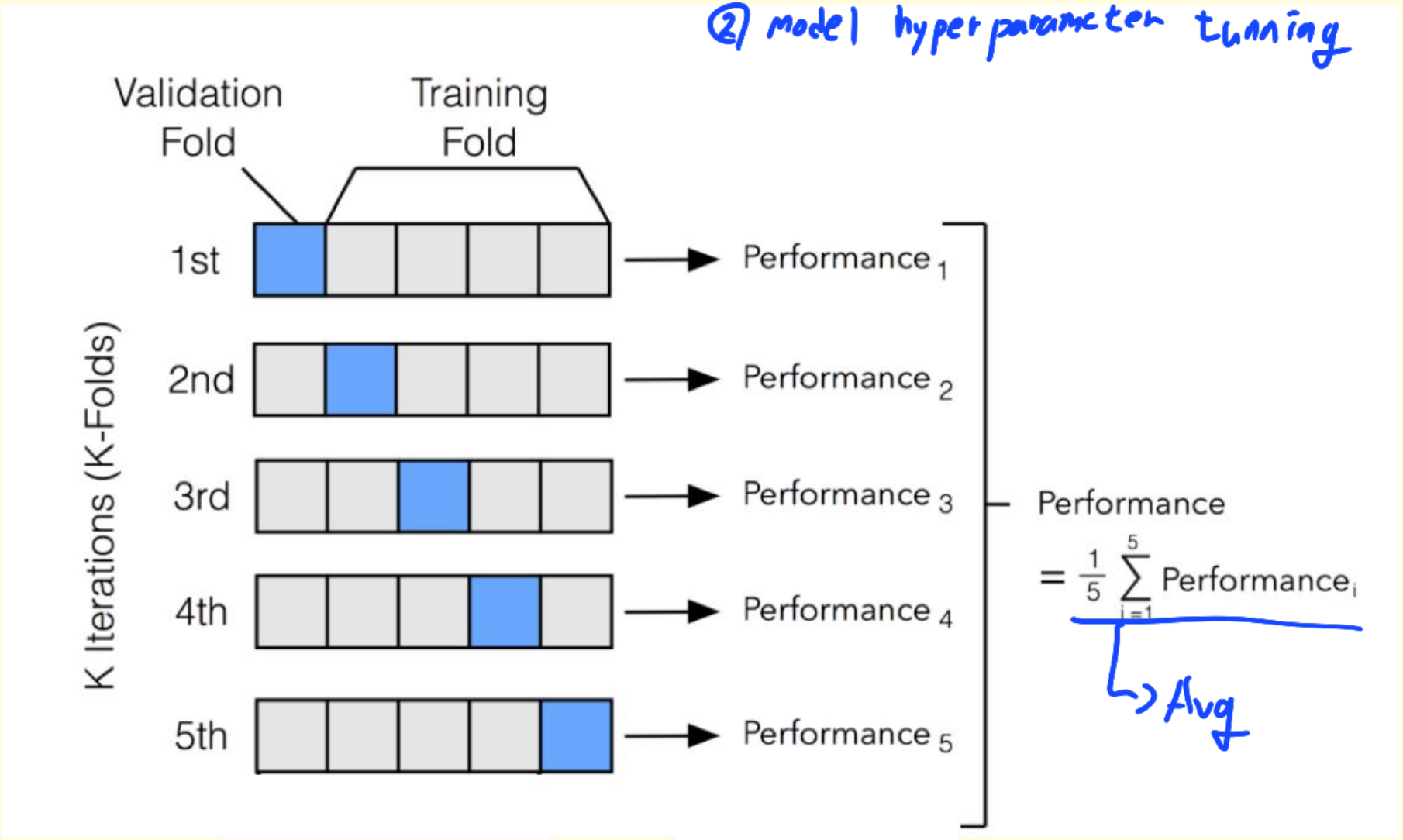
- Bootstrap, Cross Validation을 이용한 추정

- Indirect estimation: AIC, BIC 등 (Parametric한 모델의 경우)

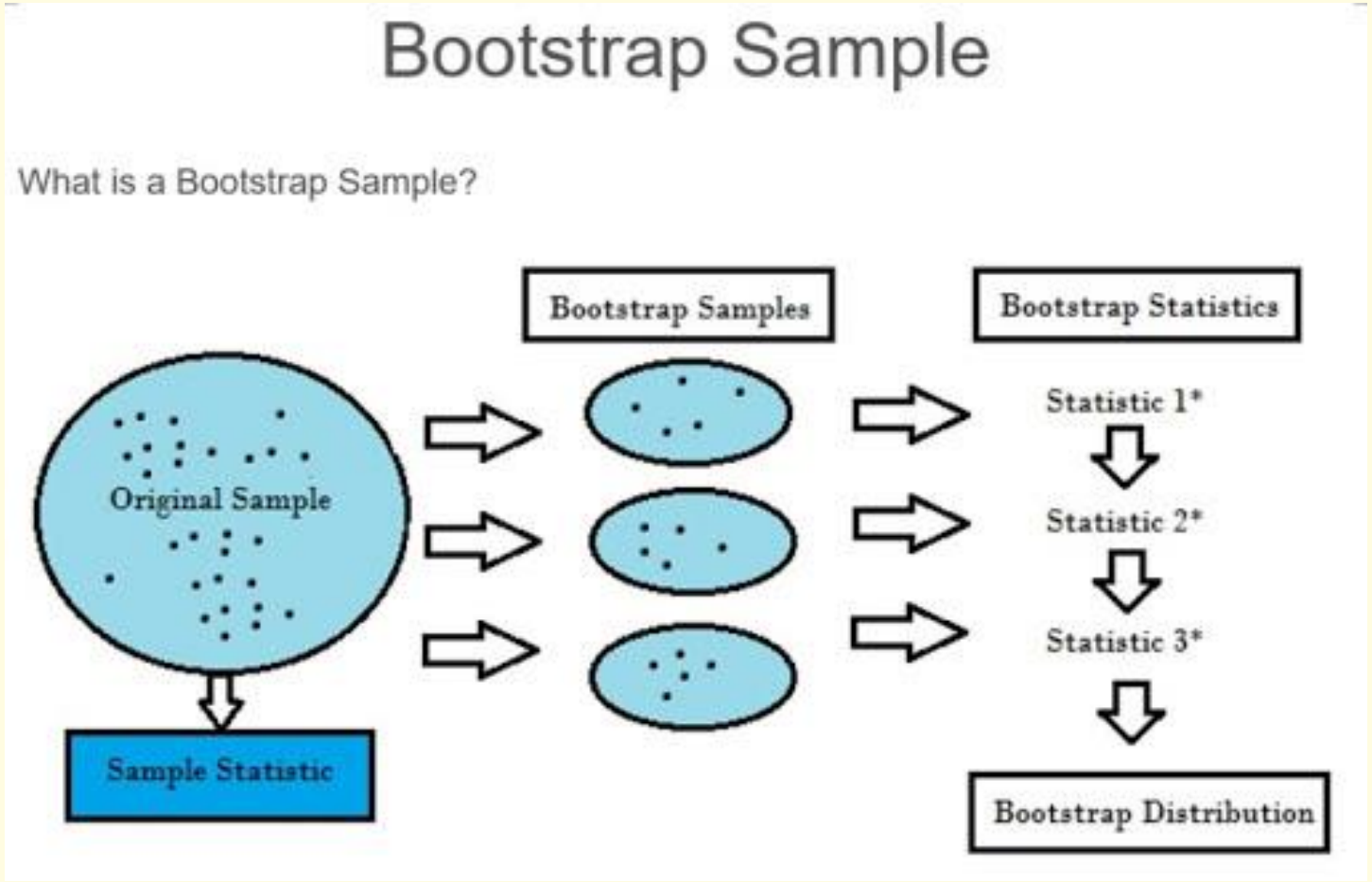
01 CV

① model selection

② model hyperparameter tuning



01 Bootstrap

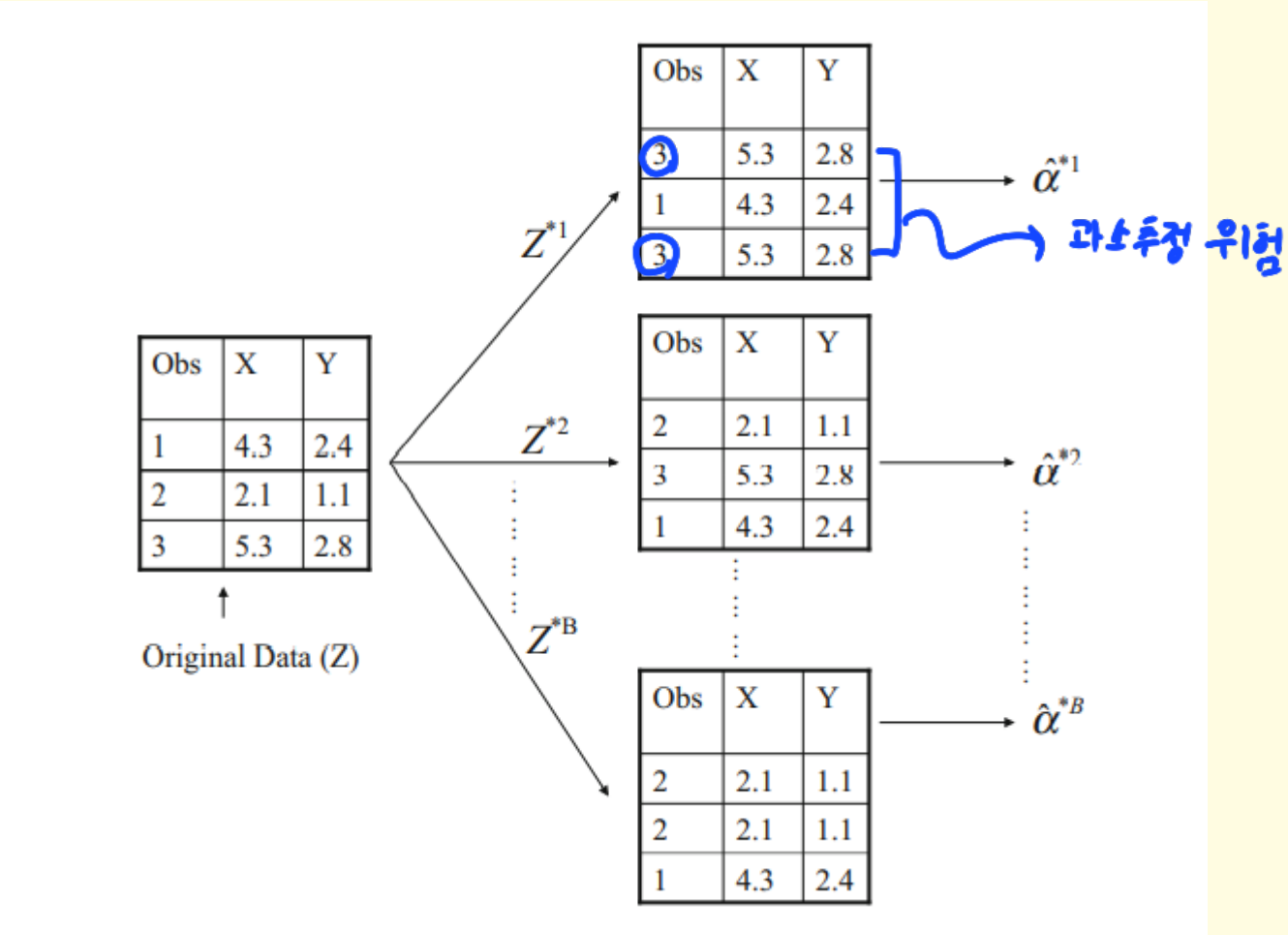


쉽게 말하자면, “복원추출 "

어렵게 말하자면,

데이터에 대해 확률 분포 가정을 하지 않고, 대신 주어진 데이터를 모집단을 대표하는 독립표본으로 가정하는 것

01 Bootstrap ⇒ 데이터가 많을 때 사용



01 Bootstrap

Bootstrap 에서 기억해야 할 특징 두 가지

한 번 뽑힌 샘플이 또 뽑힐 수 있다

한 번도 안 뽑힌 샘플이 존재할 수 있다

Decision Tree 의 치명적인 단점?

Model Variance가 아주 크다!

02 Bagging

Bias Variance Trade off

Overfitting, Underfitting

수식보다는, 간단한 예시를
들어 4가지 개념을 묶어서
설명

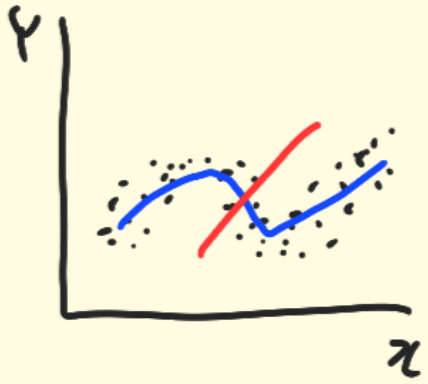
복잡한 모델, 단순한 모델

해석력과 예측력의 관계

02 Bagging

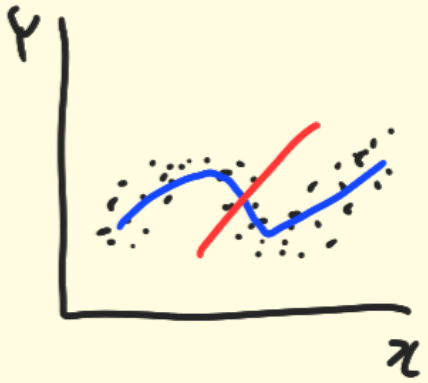
예시
제가 글씨를 잘 못써 갖고,,,
최대한 노력해보겠습니다 ㅠ

MSE = □ + bias↑ + variance↓



⇒ bias는 빨간색 ⇑

— : overfitting ⇑ ⇝ variance ⇑, bias ⇓
— : underfitting ⇑ ⇝ variance ⇓, bias ⇑



⇒ bias 빨간색 ⇑

02 Bagging

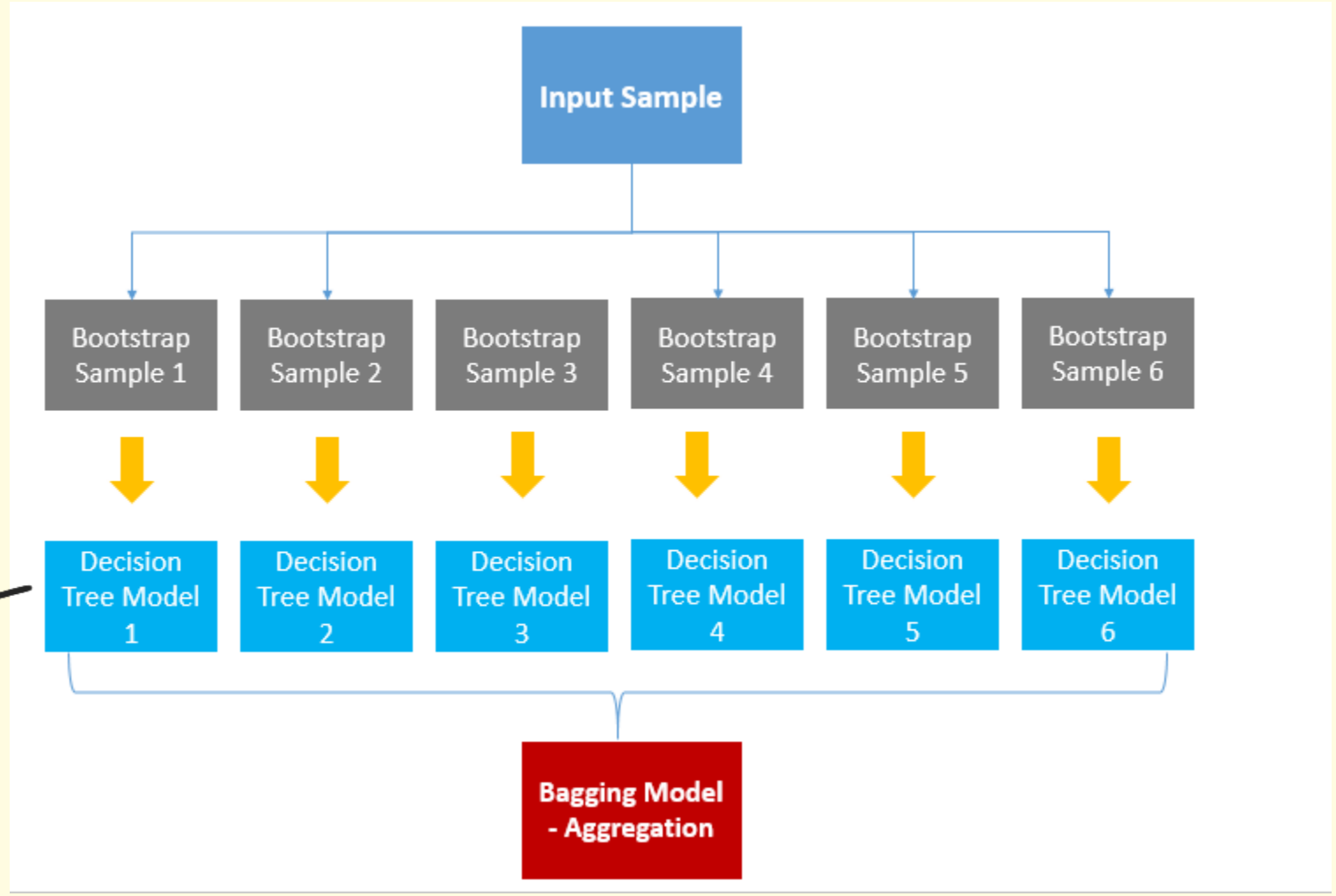
A.K.A
Bootstrap Aggregating

flexible Model => 해석력 ⇓, 예측력 ⇑

non flexible Model => 해석력 ⇑, 예측력 ⇓

02 Bagging

A.K.A
Bootstrap Aggregating



서로 다른 독립적인 모델을 만듦

02 Bagging

A.K.A
Bootstrap Aggregating

수식을 이용한 설명!!!!

단일트리 ⇒ $Var(\hat{f}) = \sigma_b^2$: b번째 bootstrap의 추정치분산

Bagging => $Var\left(\frac{1}{B}\sum^{B} f^b\right) = \frac{1}{B^2} \times Var(\Sigma f^b) = \frac{\sigma_b^2}{B}$

가정 : f가 모두 독립!!

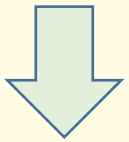
03 Random Forest

“가정”은 어디까지나 가정일 뿐!, 추정량끼리는 사실 상관관계가 있음,,

$$Var\left(\frac{1}{B}\sum^{B} f^{b}\right) = \frac{1}{B^2}\sum^{B} var(f^{b}) + 2 \times \frac{1}{B^2}\sum\sum cov(f^{b}, f^{b'})$$

03 Random Forest

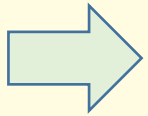
어떻게 이 상관관계를 줄이지?



Decorrelated Tree를 생성하기

03 Random Forest

기존 Decision Tree의 방식:
"Greedy Algorithm Search"



Random Forest의 방식:
M개의 변수를 랜덤하게 선택 후 트리 분할

Bootstrap 에서 기억해야 할 특징 두 가지

한 번 뽑힌 샘플이 또 뽑힐 수 있다

한 번도 안 뽑힌 샘플이 존재할 수 있다

Bootstrap 에서 기억해야 할 특징 두 가지

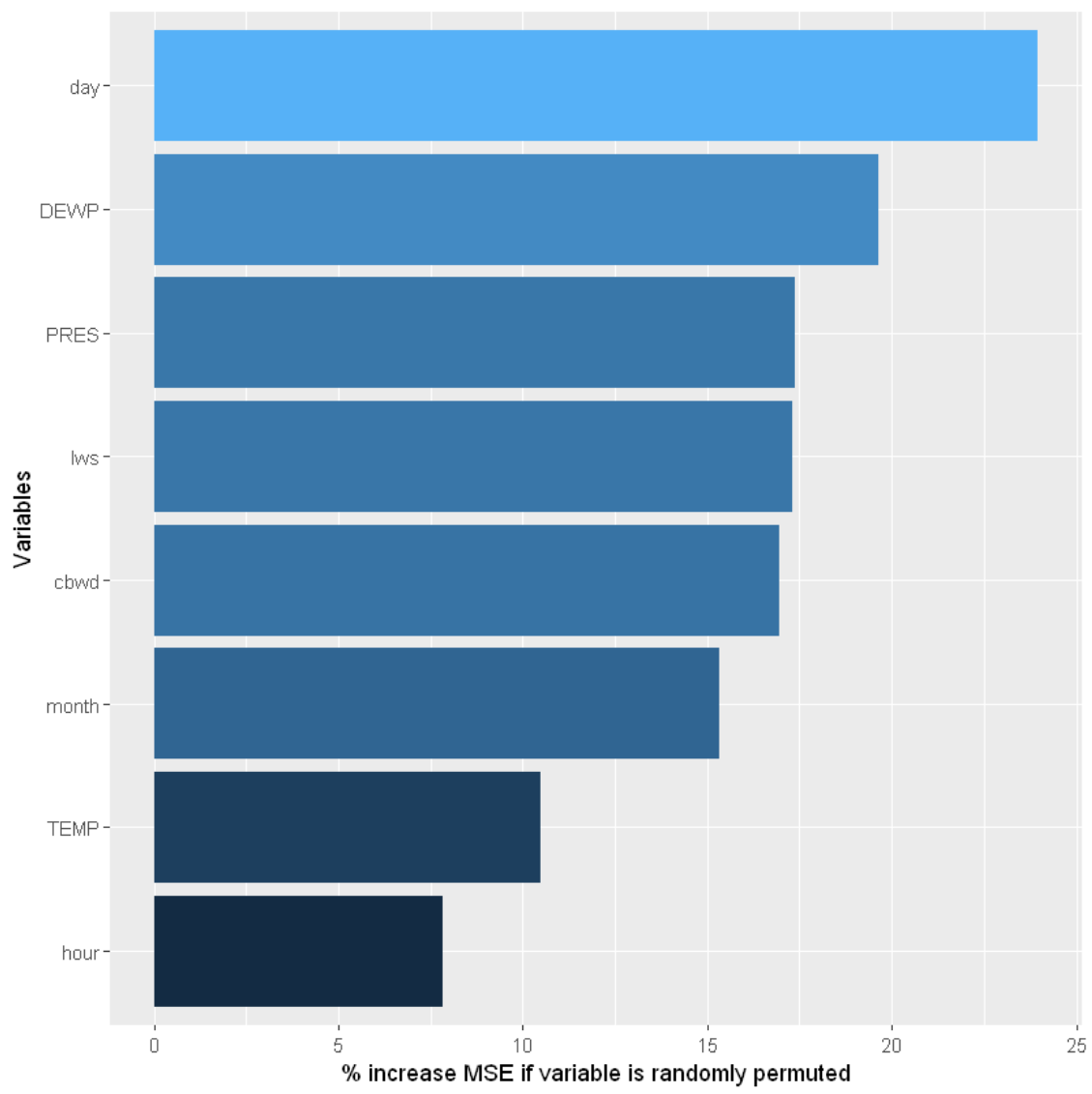
한 번 뽑힌 샘플이 또 뽑힐 수 있다

한 번도 안 뽑힌 샘플이 존재할 수 있다

한번도 뽑히지 않은 관측치들=
Out Of Bag samples
OOB Samples

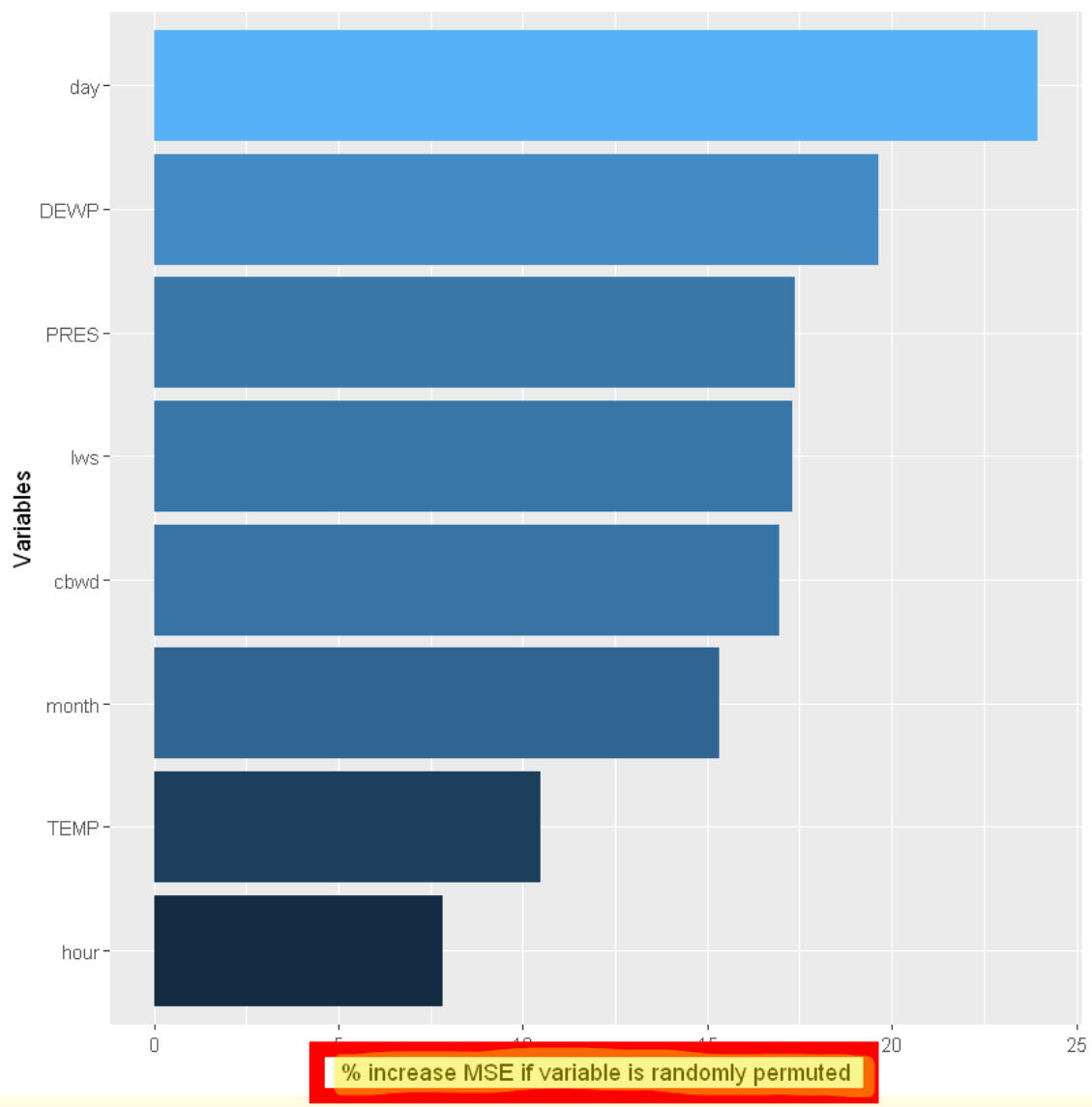
03 Random Forest

Feature importance plot in R



03 Random Forest

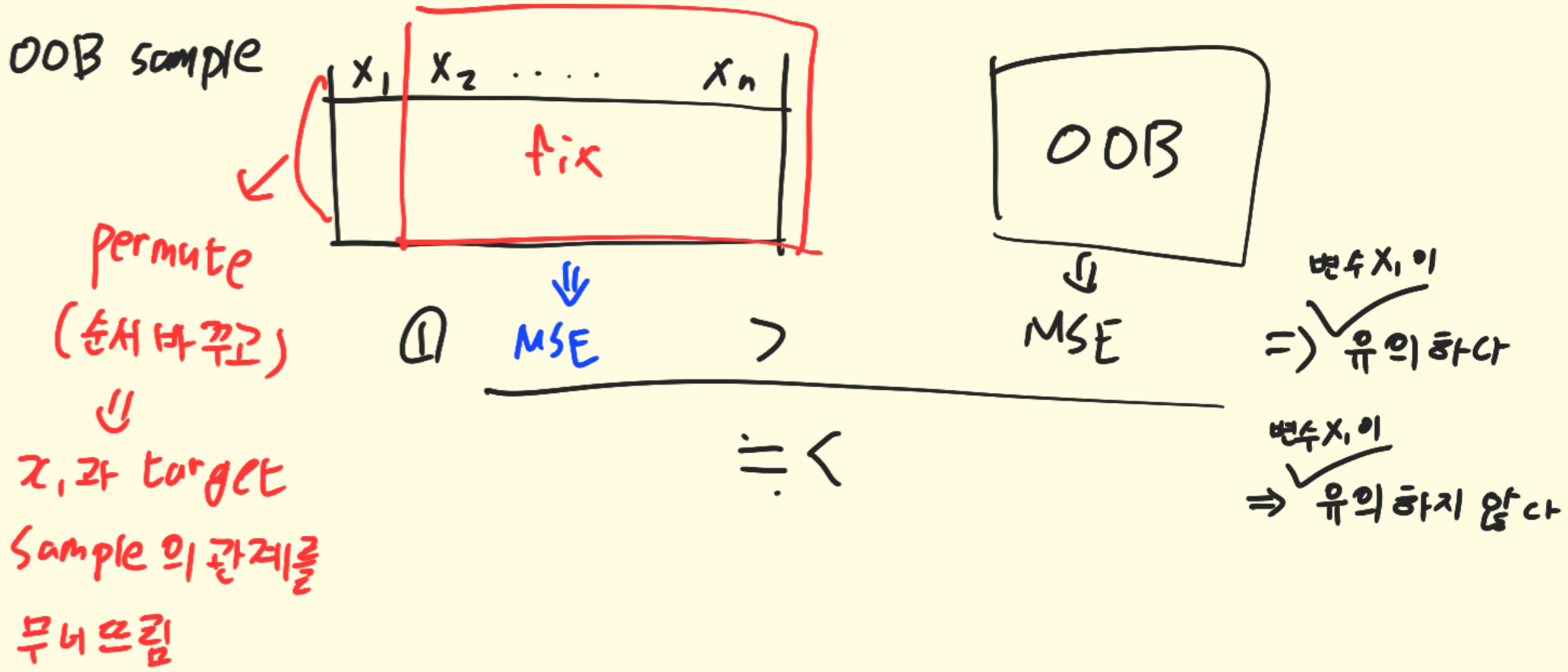
Feature importance plot in R



03 Random Forest

이를 계산하는 과정(간단하게)

feature Importance



03 Random Forest

단, Feature importance가 높다는 것이,
해당 변수가 타겟 변수와 함수적 관계가 있음
을 의미하지는 않는다

↳ 모델 안에서만 중요한것

03 Random Forest

타겟 변수와 함수적 관계를
확인하는 방법들

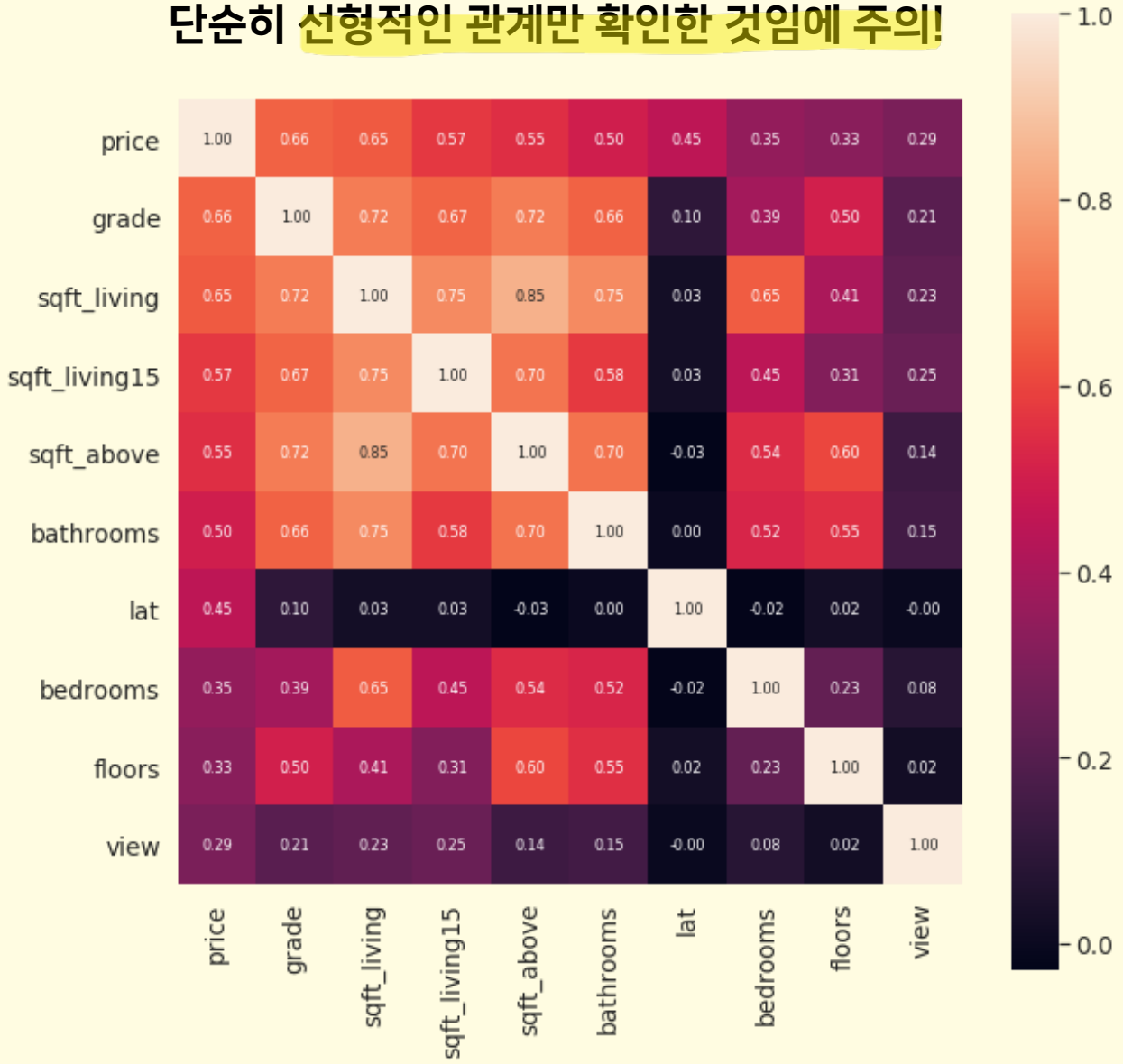
선형 관계 파악: R^2, 피어슨 상관계수(선형 관계), 스피어만 rank 상관계수(curve linear관계 혹은 nearly linear한 관계 파악)등 ….

비선형 관계 파악 : Pseudo R^2(local linear regression 을 이용한 비선형 관계 파악), MIC(maximal Information coefficient) 등….

타겟이 범주일 경우 : Relief, ReliefF 등 ….

03 Random Forest

단순히 선형적인 관계만 확인한 것임에 주의!



04 MissForest

- MCAR means that the probability if an information is missing does not depend on X_{mis} or on X_{obs} ;
- MAR means that the probability if an information is missing does not depend on X_{mis}, but may depend on X_{obs};
- MNAR means that the probability if an information is missing does depend on X_{mis}.

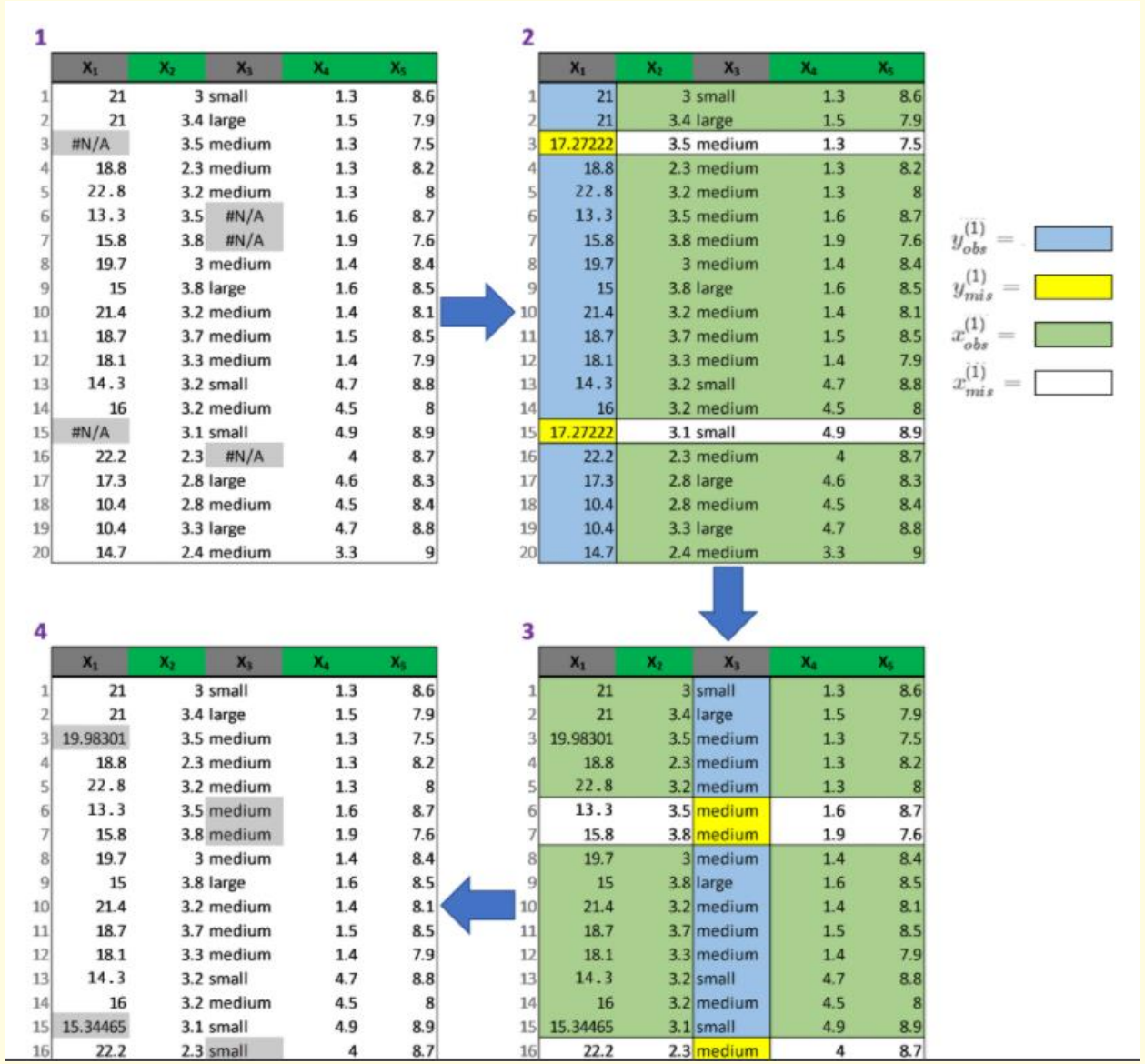
- 결측치에도 다양한 종류가 있으며, 그 다양한 종류에 맞는 다양한 처리 방법론들이 존재함.(단순 평균/중간값으로 보간하는 것부터 시작하여, MICE, KNN imputation, conditional mean, PMM 등)

- 결측치에 대한 보간은 FE 이전에 선제적으로 이루어져야 하는 Task임

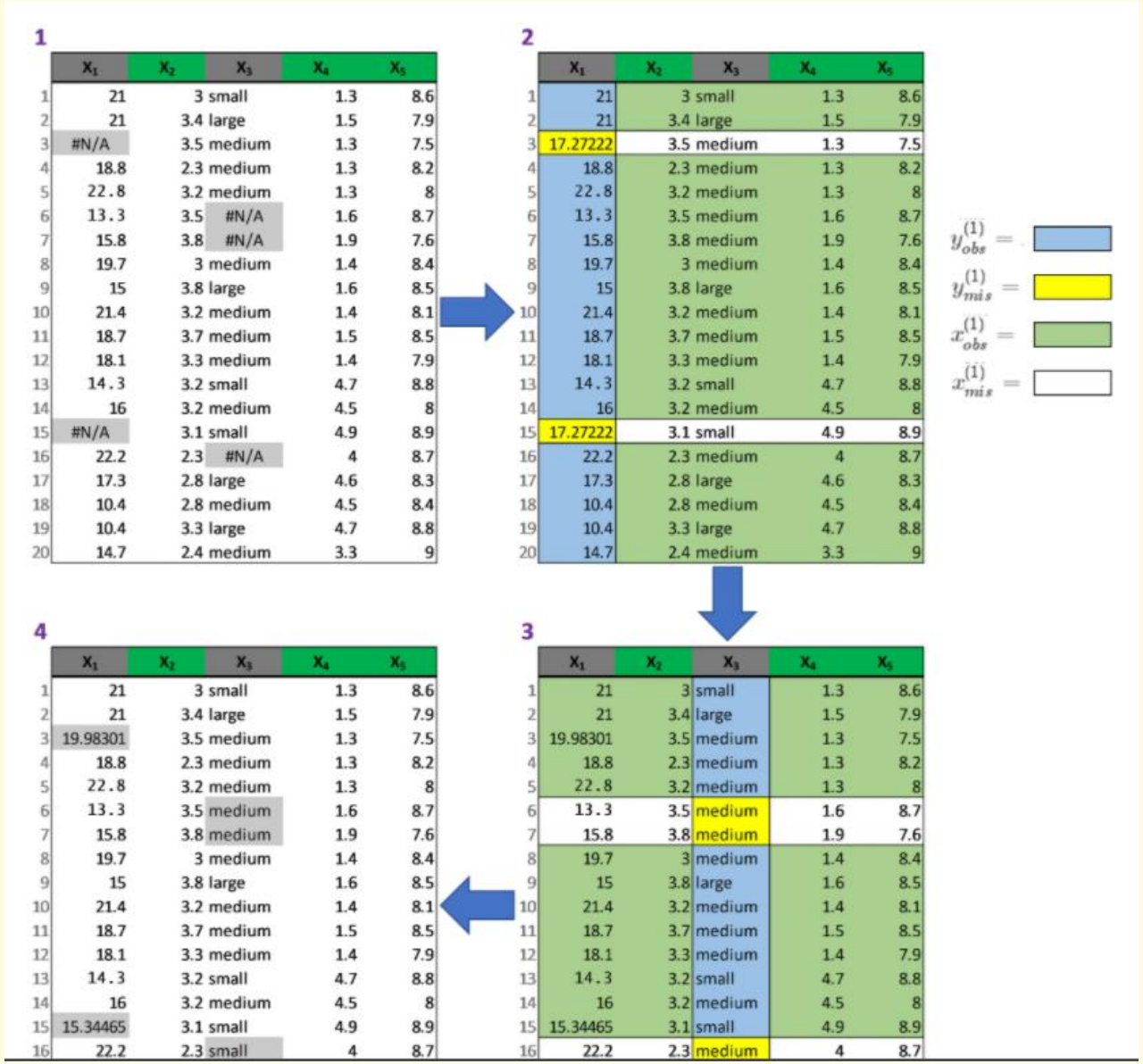
- 로버스트하며, 가정이 따로 필요 없는 비모수적 방법이며, 고차원 데이터에 적합한 Missforest 알고리즘에 대해서 소개

04 MissForest

1. 결측이 있는 행을 일단 평균이나 median으로 rough하게 채움
2. 결측의 정도에 따라(퍼센티지) 오름차순 정렬 (즉, 결측이 적은 열이 앞으로 온다)
3. 결측이 가장 적은 열에 대하여(예를 들어 X1이 결측이 제일 적었고, 3,15번째 행이 결측이었다 가정하면)
   - 3, 15번째 관측치를 제외하고 X1에 대하여 RF모델 FIT
   - 3, 15번째 관측치에 대해 FIT된 모델을 이용하여 predict
   - 이렇게 보간된 matrix를 업데이트
4. 그 다음 결측이 적은 행이 X3라고 가정하고, 결측치가 6,7,15였다고 생각하면
   - 6,7,15번째 값을 제외하고 제외하고 X3에 대해 RF FIT
   - FIT된 모델로 저 세개 값 predict 해서 imputed matrix update

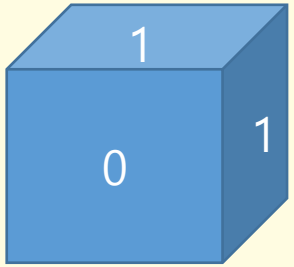


04 MissForest



05 Voting

확률변수= 확률에 따라서 결과 값이 바뀌는 변수



0과 1으로만 이루어진 주사위를 던진다고 가정하자.
이 주사위는 0이 4면, 1이 2면으로 이루어진 주사위 이다.
그렇다면 이 주사위의 기대값은?

$$\frac{2}{6} \times 1 + \frac{4}{6} \times 0 = \frac{1}{3}$$

05 Voting

대수의 강법칙

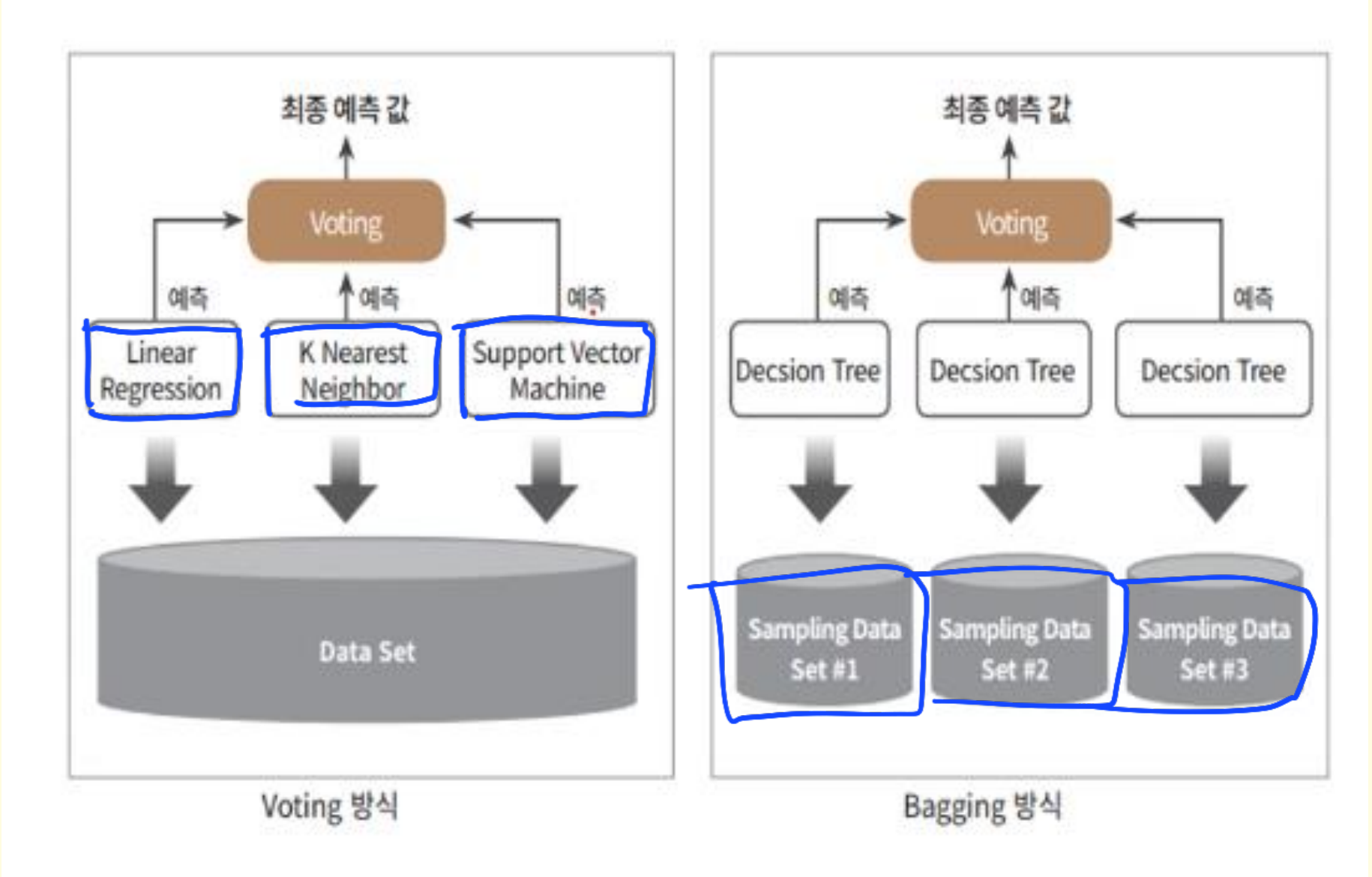
큰 수의 강한 법칙(또는 대수의 강법칙)은 확률 변수의 무한열 $X_1, X_2, X_3, \ldots$ 이 주어지고, 각 확률 변수가 $E(|X_i|) < \infty$ 이고 (기댓값 μ), 서로 독립이며 동일한 분포일 때,

$$\mathrm{P}\left(\lim_{n\to\infty} \overline{X}_n = \mu\right) = 1 \qquad \overline{X}_n = (X_1 + \cdots + X_n)/n$$

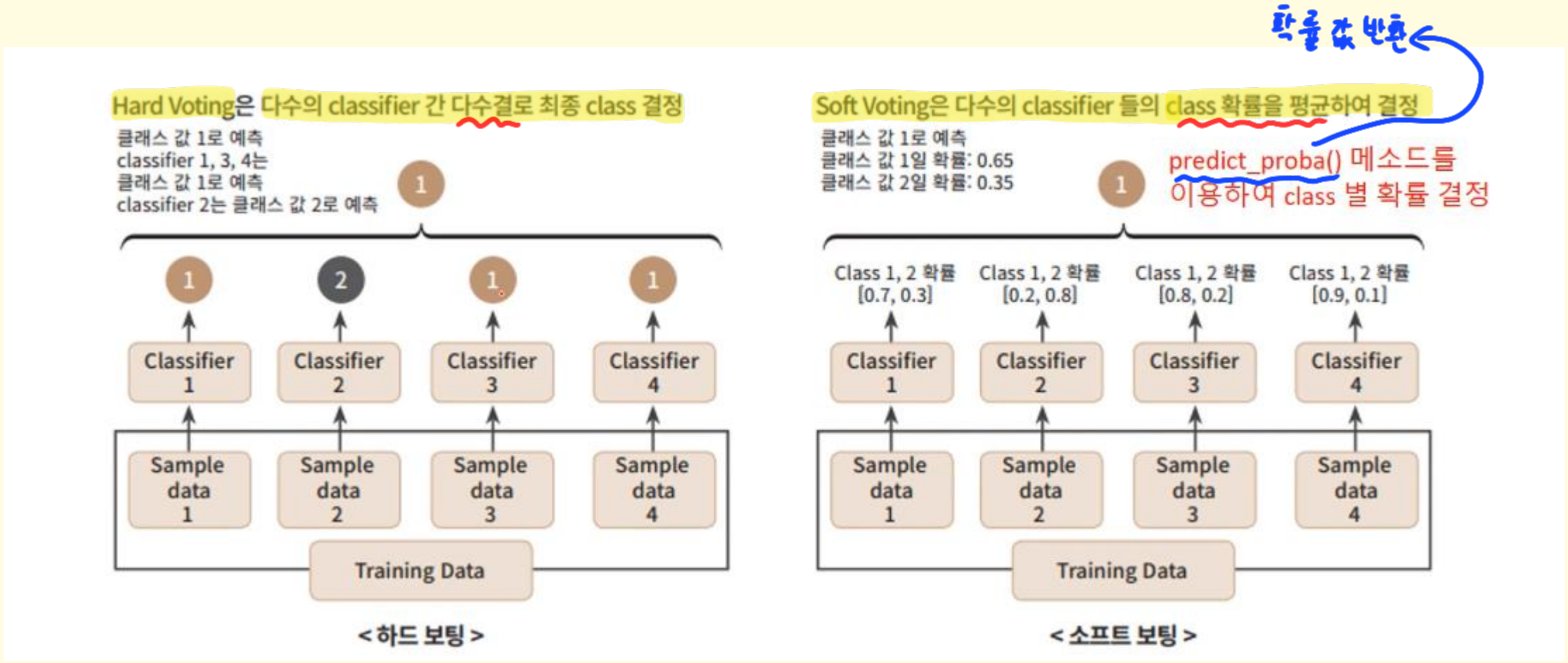
이 성립한다. 즉 표본의 평균은 거의 확실하게 μ로 수렴한다.

앙상블 성능 향상 ⇒ 2~3%, 과적방지

05 Voting


최종 예측 값
Voting
예측
예측
예측
Linear Regression
K Nearest Neighbor
Support Vector Machine
Data Set
Voting 방식
최종 예측 값
Voting
예측
예측
예측
Decsion Tree
Decsion Tree
Decsion Tree
Sampling Data Set #1
Sampling Data Set #2
Sampling Data Set #3
Bagging 방식

05 Voting


확률값 반환
Hard Voting은 다수의 classifier 간 다수결로 최종 class 결정
클래스 값 1로 예측
classifier 1, 3, 4는
클래스 값 1로 예측
classifier 2는 클래스 값 2로 예측
1
1
2
1
1
Classifier 1
Classifier 2
Classifier 3
Classifier 4
Sample data 1
Sample data 2
Sample data 3
Sample data 4
Training Data
< 하드 보팅 >
Soft Voting은 다수의 classifier 들의 class 확률을 평균하여 결정
클래스 값 1로 예측
클래스 값 1일 확률: 0.65
클래스 값 2일 확률: 0.35
predict_proba() 메소드를
이용하여 class 별 확률 결정
1
Class 1, 2 확률 [0.7, 0.3]
Class 1, 2 확률 [0.2, 0.8]
Class 1, 2 확률 [0.8, 0.2]
Class 1, 2 확률 [0.9, 0.1]
Classifier 1
Classifier 2
Classifier 3
Classifier 4
Sample data 1
Sample data 2
Sample data 3
Sample data 4
Training Data
< 소프트 보팅 >


끝!