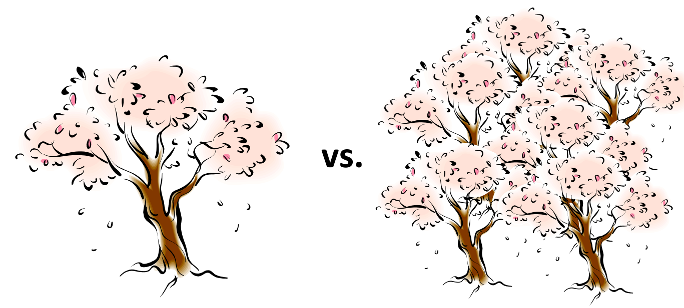
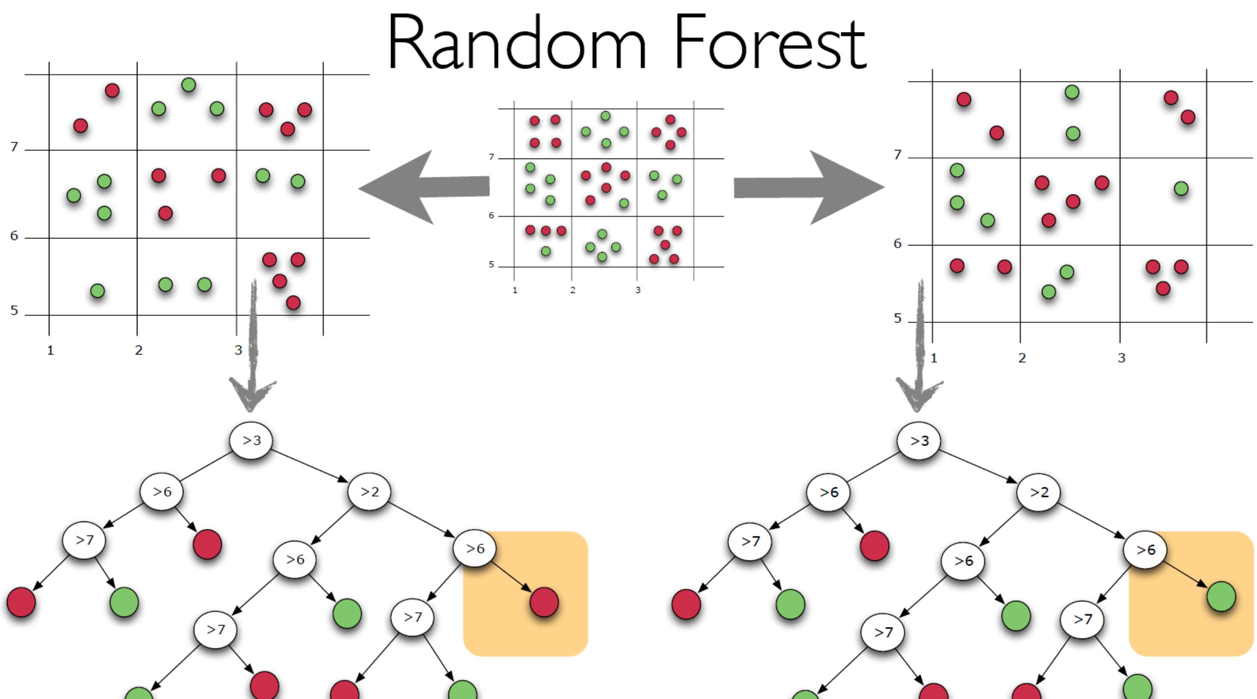
Ensemble Learning: Random Forests

Pilsung Kang
School of Industrial Management Engineering
Korea University

Random Forests

- A specialized bagging for decision tree algorithms
- Two ways to increase the diversity of ensemble
  - ✓ Bagging
  - ✓ Randomly chosen predictor variables





Random Forests

- Random Forests: Algorithm

1. For $b = 1$ to B:

   (a) Draw a $\boxed{\text{bootstrap sample}}$ $\mathbf{Z}^*$ of size N from the training data.

   (b) Grow a random-forest tree T_b to the bootstrapped data, by recursively repeating the following steps for each terminal node of the tree, until the minimum node size n_{min} is reached.

      i. Select $\boxed{m \text{ variables at random}}$ from the p variables.

      ii. Pick the best variable/split-point among the m.

      iii. Split the node into two daughter nodes.

2. Output the ensemble of trees $\{T_b\}_1^B$.

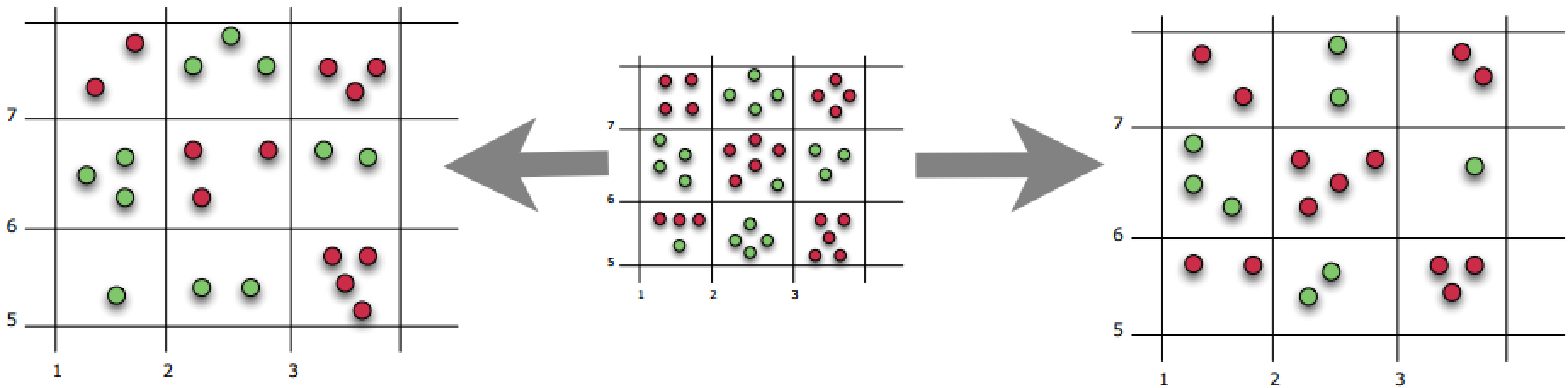
To make a prediction at a new point x:

Regression: $\hat{f}_{\text{rf}}^B(x) = \frac{1}{B}\sum_{b=1}^B T_b(x)$.

Classification: Let $\hat{C}_b(x)$ be the class prediction of the bth random-forest tree. Then $\hat{C}_{\text{rf}}^B(x) = \textit{majority vote } \{\hat{C}_b(x)\}_1^B$.

Random Forests

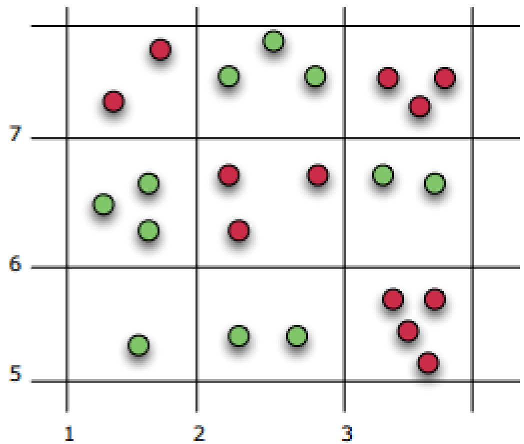
- Bagging
  - ✓ Sampling with replacement



Random Forests

- Bagging
  - ✓ Randomly selected variable

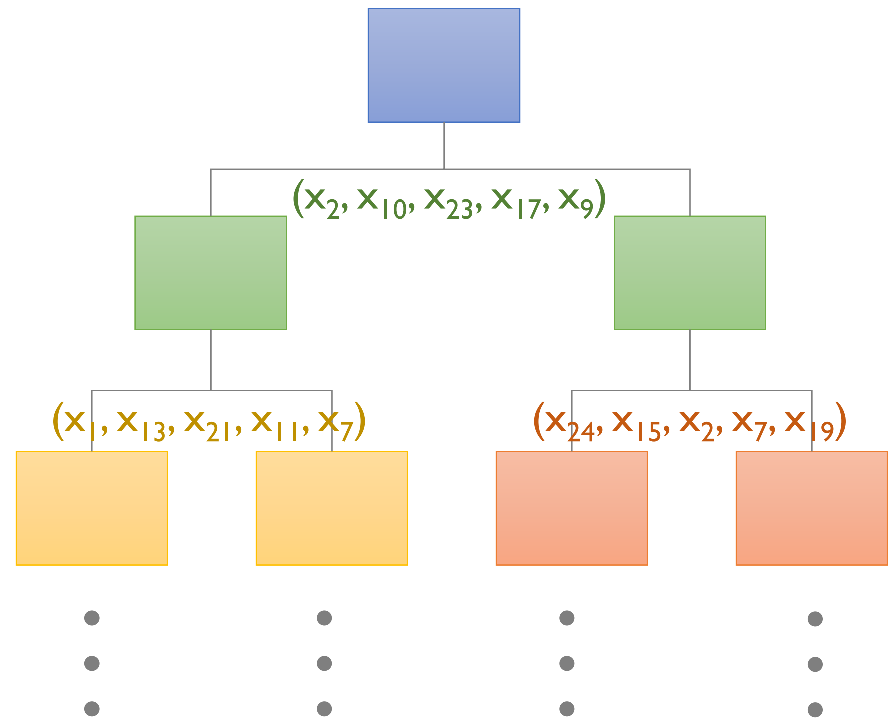
Bootstrap i (X in R^{25})



$(x_2, x_{10}, x_{23}, x_{17}, x_9)$

$(x_1, x_{13}, x_{21}, x_{11}, x_7)$

$(x_{24}, x_{15}, x_2, x_7, x_{19})$



Random Forests

- Generalization Error
  - ✓ Each tree in random forests may over-fit the data because pruning is not conducted.
  - ✓ If the population size is large enough, then the generalization error of random forests bounded by

$$Generalization\ Error \leq \frac{\bar{\rho}(1-s^2)}{s^2}$$

    - $\bar{\rho}$ is the mean value of the correlation coefficients between individual trees
    - s^2 is the margin function (for binary classification, it is simply the average difference proportions between the correct and incorrect trees over all training data.
  - ✓ The more accurate the individual classifiers, the larger the s^2 and the lower the generalization error
  - ✓ The less correlated among the classifiers, the lower the generalization error.

Random Forests

- Generalization Error: Example

Model A			
Label	P(y=1)	P(y=0)	Margin
1	0.90	0.10	0.80
1	0.80	0.20	0.60
1	0.75	0.25	0.50
1	0.78	0.22	0.56
1	0.51	0.49	0.02
0	0.24	0.76	0.52
0	0.12	0.88	0.76
0	0.14	0.86	0.72
0	0.01	0.99	0.98
0	0.14	0.86	0.72
Average Margin			0.62

Model B			
Label	P(y=1)	P(y=0)	Margin
1	0.58	0.42	0.16
1	0.65	0.35	0.30
1	0.94	0.06	0.88
1	0.99	0.01	0.98
1	0.98	0.02	0.96
0	0.06	0.94	0.88
0	0.05	0.95	0.90
0	0.04	0.96	0.92
0	0.18	0.82	0.64
0	0.08	0.92	0.84
Average Margin			0.75

Model C			
Label	P(y=1)	P(y=0)	Margin
1	0.88	0.12	0.76
1	0.98	0.02	0.96
1	0.97	0.03	0.94
1	0.89	0.11	0.78
1	0.92	0.08	0.84
0	0.08	0.92	0.84
0	0.02	0.98	0.96
0	0.05	0.95	0.90
0	0.08	0.92	0.84
0	0.04	0.96	0.92
Average Margin			0.87

✓ Average correlation = 0.9027 (A & B – 0.8229, A & C = 0.9413, B & C = 0.9438)

✓ Average margin = 0.7460

✓ Generalization error <= 0.3074

Random Forests

- Variable Importance
  - ✓ Step 1: Compute the OOB error for the original dataset (e_i)
  - ✓ Step 2: Compute the OOB error for the dataset in which the variable x_i is permuted (p_i)
  - ✓ Step 3: Compute the variable importance based on the mean and standard deviation of (p_i-e_i) over all trees in the population

Random Forests

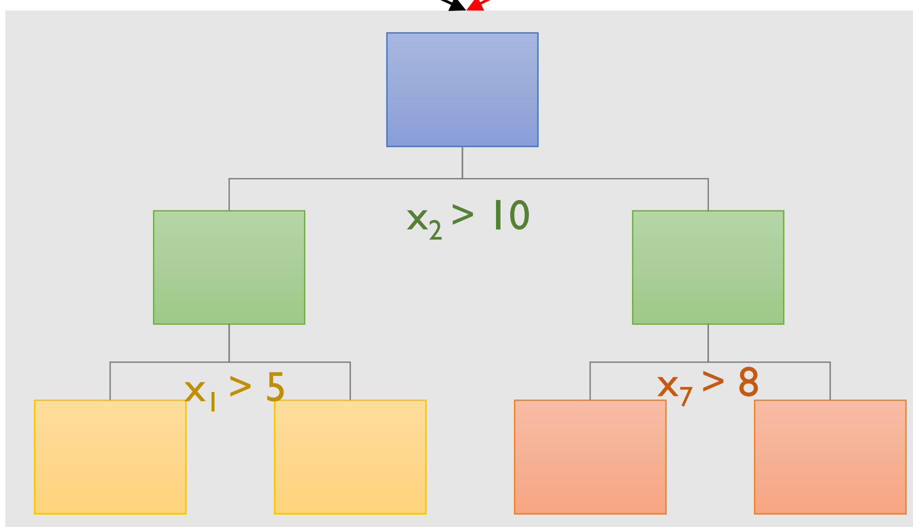
Original OOB Data

ID	X1	...	**Xi**	...	Xd	Y
1			**0.1**			
2			**0.5**			
3			**1.1**			
4			**1.2**			
5			**0.4**			
6			**0.2**			
7			**0.7**			
8			**0.8**			
9			**1.4**			
10			**1.6**			

i번째 변수에 대한 random permutation이 수행된 OOB Data

ID	X1	...	**Xi**	...	Xd	Y
1			**1.1**			
2			**0.2**			
3			**0.1**			
4			**1.4**			
5			**1.2**			
6			**0.5**			
7			**1.6**			
8			**0.8**			
9			**0.7**			
10			**0.4**			

변수 i가 Tree를 split하는데
한번도 사용되지 않았다면



OOB Error of the Original Data e_i = OOB Error of the Permuted Data p_i

Random Forests

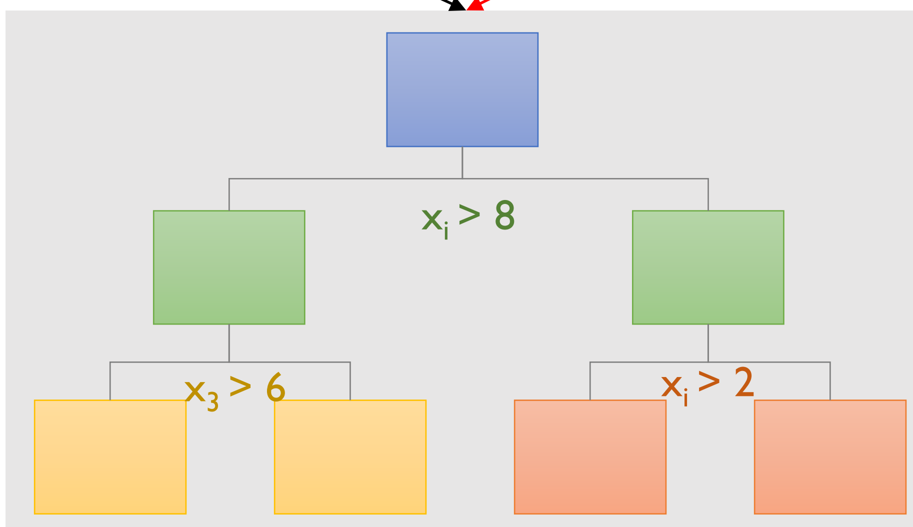
Original OOB Data

ID	X1	...	**Xi**	...	Xd	Y
1			**0.1**			
2			**0.5**			
3			**1.1**			
4			**1.2**			
5			**0.4**			
6			**0.2**			
7			**0.7**			
8			**0.8**			
9			**1.4**			
10			**1.6**			

i번째 변수에 대한 random permutation이 수행된 OOB Data

ID	X1	...	**Xi**	...	Xd	Y
1			**1.1**			
2			**0.2**			
3			**0.1**			
4			**1.4**			
5			**1.2**			
6			**0.5**			
7			**1.6**			
8			**0.8**			
9			**0.7**			
10			**0.4**			

변수 i가 Tree를 split하는데 중요하게 사용되었다면



OOB Error of the Original Data e_i

OOB Error of the Permuted Data p_i

Random Forests

- 변수의 중요도
  - ✓ 랜덤 포레스트에서 변수의 중요도가 높다면
    - 1) Random permutation 전-후의 OOB Error 차이가 크게 나타나야 하며,
    - 2) 그 차이의 편차가 적어야 함

    - m번째 tree에서 변수 i에 대한 Random permutation 전후 OOB error의 차이

$$d_i^m = p_i^m - e_i^m$$

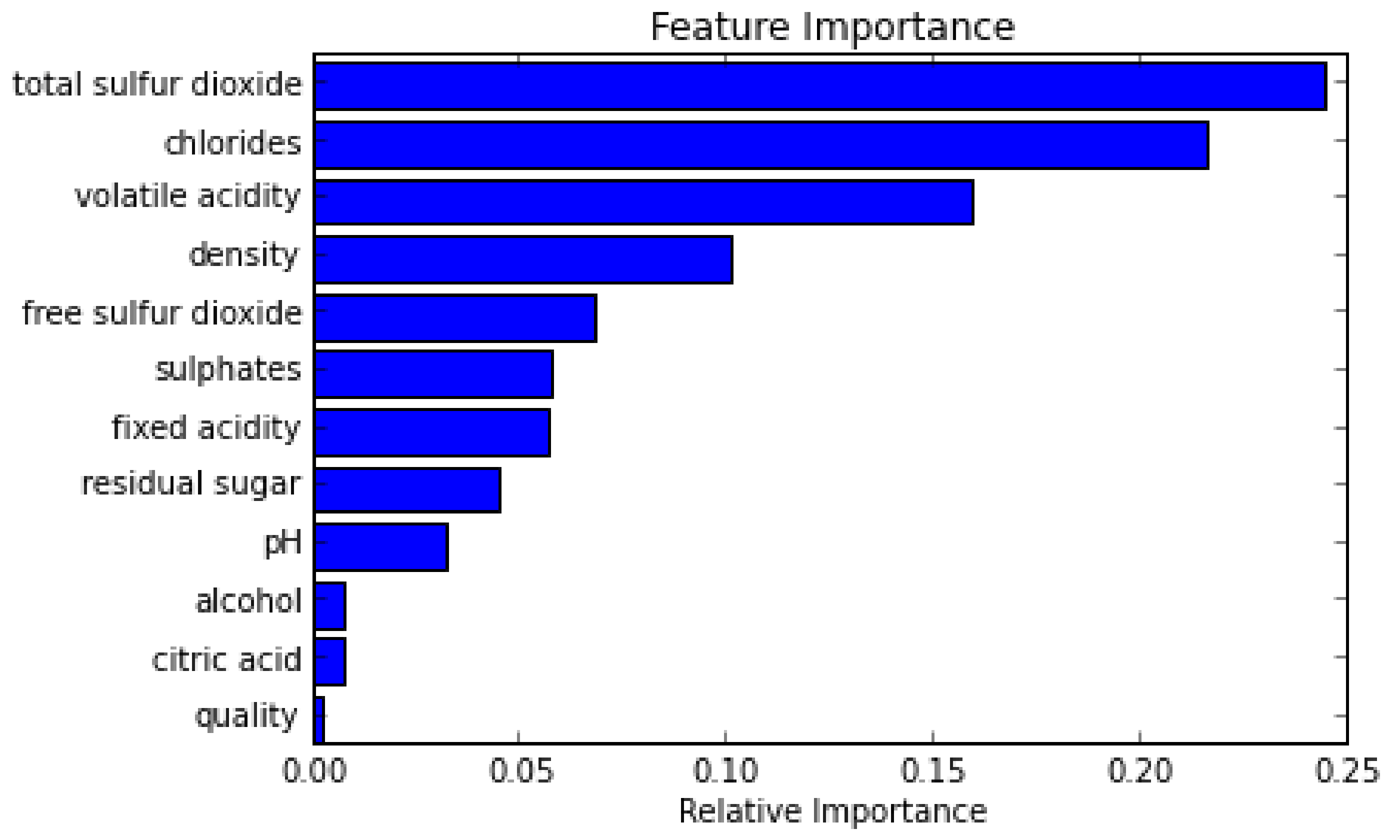
    - 전체 Tree들에 대한 OOB error 차이의 평균 및 분산

$$\overline{d}_i = \frac{1}{m}\sum_{i=1}^{m} d_i^m, \quad s_i^2 = \frac{1}{m-1}\sum_{i=1}^{m}(d_i^m - \overline{d}_i)^2$$

    - i번째 변수의 중요도: $v_i = \dfrac{\overline{d}_i}{s_i}$

Random Forests

- 변수 중요도 산출 결과



고려대학교 KOREA UNIVERSITY

DSBA Data Science & Business Analytics


Q & A