DATA MINING B(3)

11-2 Decision Tree (details)

Decision Tree

- In this material, the detail of Decision Tree is explained.
- The contents are,
  - Numeric Attributes
  - Missing Values
  - Pruning 枝刈り

Numeric Attributes

- Suppose a restriction that we separate only 2 branches, and we use “weather data” with numeric attributes.
- Consider separate by “temperature”.
  - Sort values, same values are collapsed.
  - Suppose that it is not allowed to separate items of the same class.
  - There are 8 possible break point.

Table: Weather Data

	outlook	temperature	humidity	windy	play
1	sunny	85	85	FALSE	no
2	sunny	80	90	TRUE	no
3	overcast	83	86	FALSE	yes
4	rainy	70	96	FALSE	yes
5	rainy	68	80	FALSE	yes
	rainy	65	70	TRUE	no
	overcast	64	65	TRUE	yes
	sunny	72	95	FALSE	no
	sunny	69	70	FALSE	yes
	rainy	75	80	FALSE	yes
	sunny	75	70	TRUE	yes
	overcast	72	90	TRUE	yes
	overcast	81	75	FALSE	yes
14	rainy	71	91	TRUE	no

64	65	68	69	70	71	72	75	80	81	83	85
Yes	no	yes	yes	yes	no	no yes	yes yes	no	yes	yes	no

Numeric Attributes (cont.)

64	65	68	69	70	71	72	75	80	81	83	85
Yes	no	yes	yes	yes	no	no	yes	no	yes	yes	no
						yes	yes				

- If breakpoint is 71.5, gains can be calculated as follows.
  - For below 71.5, we have 4 instances of yes and 2 instances of no.
    - $H_b = -\frac{4}{6}\log\frac{4}{6} - \frac{2}{6}\log\frac{2}{6}$
  - For over 71.5, we have 5 instances of yes and 3 instances of no.
    - $H_o = -\frac{5}{8}\log\frac{5}{8} - \frac{3}{8}\log\frac{3}{8}$
  - Totally, gain is
    - $I = \frac{4}{6}H_b + \frac{5}{8}H_o$

Numeric Attributes: Sort

- After separation, we consider another test in child nodes.
- Do we need to sort for another attributes?
  - No, if we sort all attributes at first.
  - Sort is required only once.

Consider about “temperature”, Values and IDs.

64	65	68	69	70	71	72	72	75	75	80	81	83	85
7	6	5	9	4	14	8	12	10	11	2	13	3	1

← Instance ID

If separated by outlook=sunny, which IDs are

9	8	11	2	1

Pick up above IDs in order of IDs in the sorted values.

64	65	68	69	70	71	72	72	75	75	80	81	83	85
7	6	5	9	4	14	8	12	10	11	2	13	3	1

The order of values are kept. No need to sort again.

69	72	75	80	85
9	8	11	2	1

Missing Values on Constructing Tree

- The instance with missing values will be…
  - Ignored.
    - It is the easiest way, but it has many drawbacks.
    - Usually, the instance has many information as well as the one without missing values.
  - Separated by the ratio of values in the training dataset.
    - Discussed in previous section.
    - Aggregate at leaves.
    - In the both case of the attribute is the test attribute or not, this concept is applicable.

Pruning

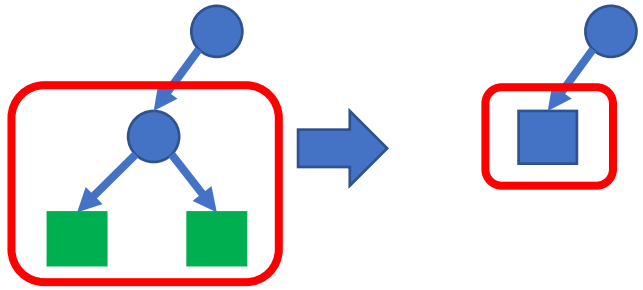
- Fully expanded tree is not always better.
  - It often contains unnecessary structure.
  - Pruning is required.

- Pruning
  - Cut unnecessary structure

  - Pre-pruning: Pruning is involved during tree building process.
  - Post-pruning: Pruning is involved after tree building process.

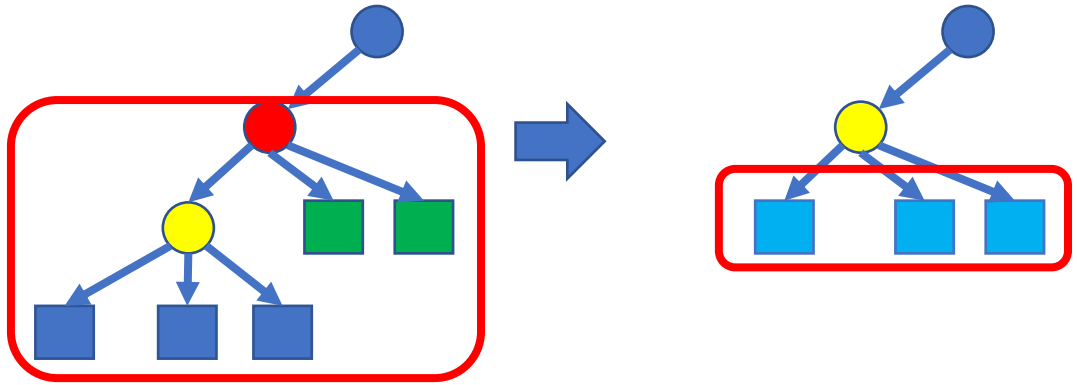
Post-Pruning

- 2 type of post-pruning
  1. Sub-tree replacement.



Cut subtree.
Instances in the green leaves will be gathered.

  2. Sub-tree raising.



Replace a test of red-node by yellow-node, and re-allocate instances in the green leaves.

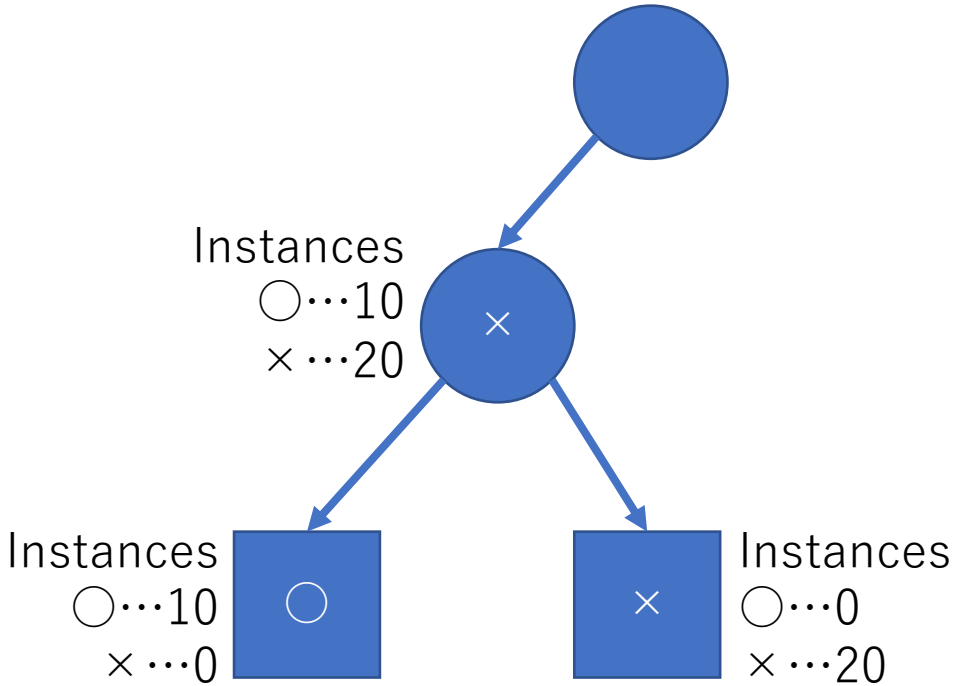
Unnecessary structure

- How to find the unnecessary structure.
  - Estimate error rate by validation data.

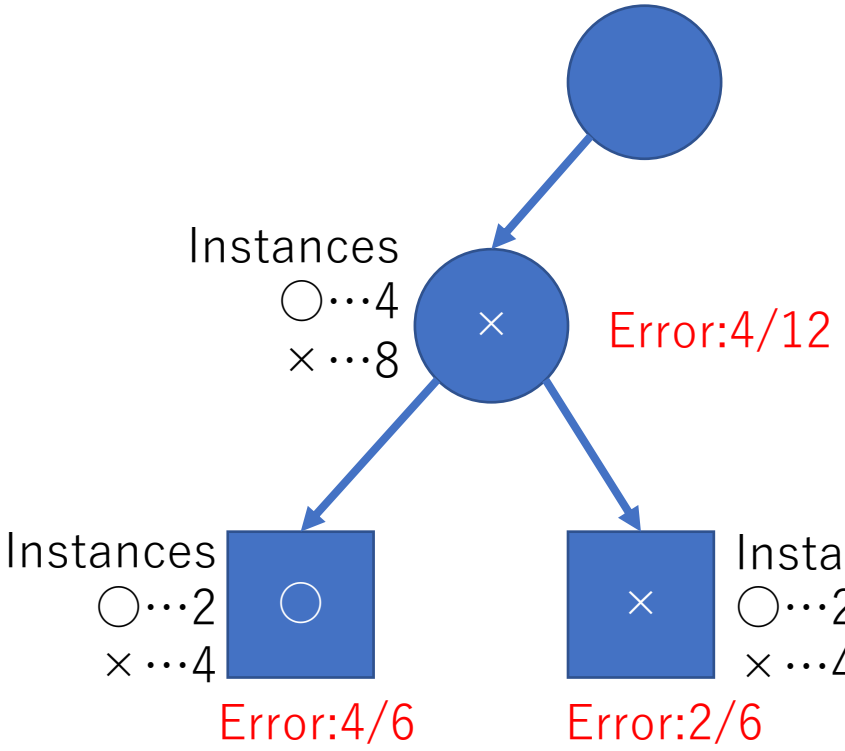
  - Training data: For construction, calculate gain.
  - Test data: For evaluation, calculate accuracy of the tree.
  - Validation data: For pruning or parameter search.

Example of Error Estimation

- Sub-tree replacement.



During building tree

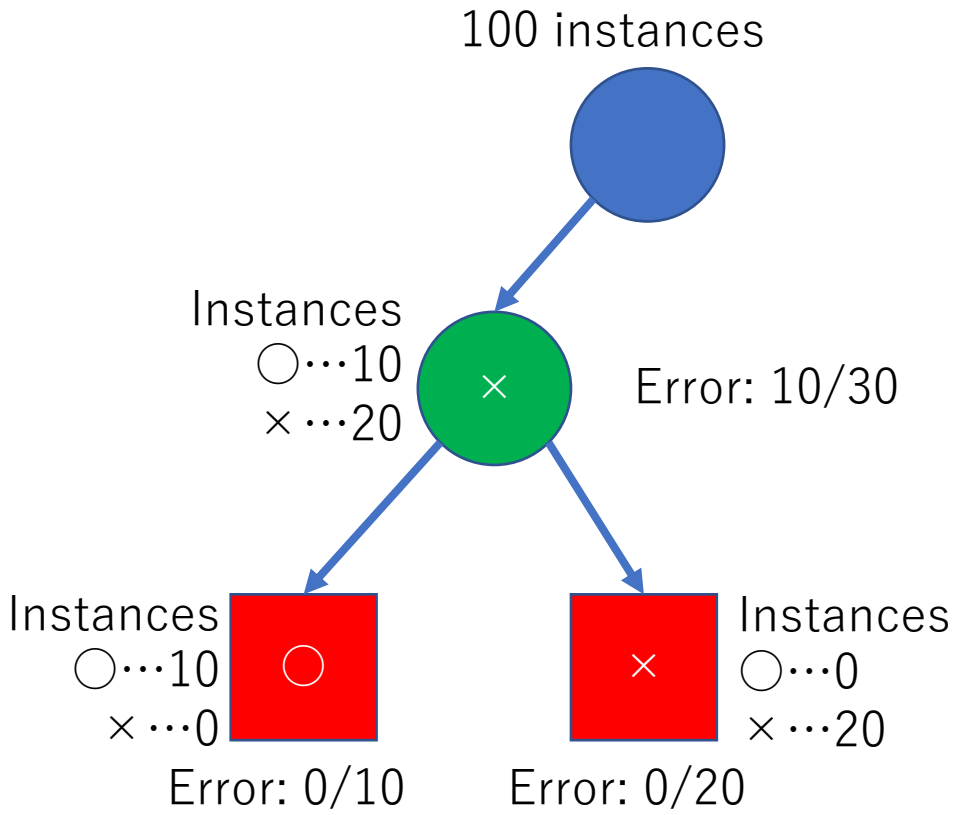


For validation data

Error
Parent 4/12
Children (4/6)*(6/12)
+(2/6)*(6/12)
=6/12
It should be pruned.

Cost Complexity Pruning

- Another post-pruning method.
- Consider the number of child nodes.
  - Evaluation
    - Error + α *number of nodes.
    - Error = error rate on the node
      * probability of arrival to this node
  - Error of Parent(green)
    - 10/30 * 30/100 = 0.1
    - Number of node = 1
  - Error of Child nodes(red)
    - 0/10 * 10/100 + 0/20 * 20/100 = 0
    - Number of node = 2
  - Compare
    - Parent 0.1 + α
    - Child 2α
    - If parent < child (α >0.1) then replace the subtree.
    - Suitable α is decided by validation data.



Pre-pruning

- Involve pruning during building the tree.
- Can’t use validation dataset.
  - Error estimation is difficult.

- Estimate error statistically.
  - Pessimistic Pruning.

Pessimistic Pruning

- Suppose that the occurrence of an error is binary event.
  - Consider the error distribution follows a binary distribution.
  - Let N: number of instances, f: obserbed error rate, Nf: number of error E, q: true error rate(unknown).
  - Binary distribution $B(N, q)$, $\sigma = \sqrt{Nq(1-q)}$
  - Normalize as normal distribution: $N(Nq, Nq(1-q))$
  - the range of q at the c% confidence interval is
    - $z < \frac{Nf-Nq}{\sqrt{Nq(1-q)}} = \frac{f-q}{\sqrt{q(1-q)/N}}$ $z = P(\frac{1-\frac{c}{100}}{2})$
    - The pessimistic case, maximize error rate, $q = \frac{f+\frac{z^2}{2} \pm z\sqrt{\frac{f}{N}+\frac{f^2}{N}+\frac{z^2}{4N}}}{1+\frac{z^2}{N}}$
  - Ex: $c = 50\%\ then\ z = 0.69, and\ if\ E = 2\ and\ N = 6\ then\ q = 0.47$
    - Observed error rate is 1/3, however, estimated rate is 0.47. We use 0.47 for evaluation.

C4.5

- Open source decision tree construction program.
  - Source code is also open to the public.
  - The successor C5.0 (See5.0) is commercial software.
    - Source and algorithms are closed.
- Many techniques are introduced.
  - Pruning: Post and Pre pruning.
    - In pessimistic pruning, C=25%
    - Cost-Complexity Pruning
  - Minimum instance restriction at leaf. (Default: 2)
  - Balance of separation.
  - Information Gain Ratio

Summary

- More detail of decision tree algorithm.
  - For implementation.
  - Numerical Attribute.
    - Separation and sort.
  - Missing Values
  - Pruning
    - Post-pruning
      - Sub-tree replacement and Sub-tree raising.
      - Cost-Complexity pruning
    - Pre-pruning
      - Pessimistic Pruning