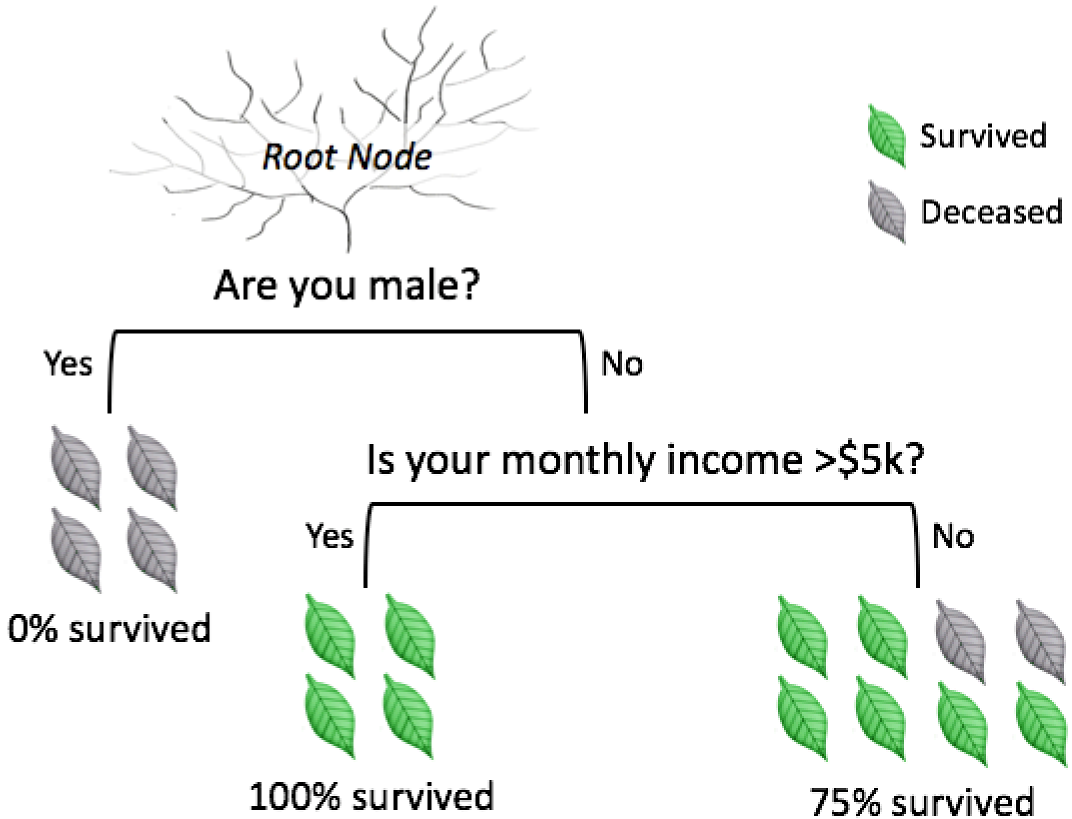
DECISION TREE

Week10

Decision Tree

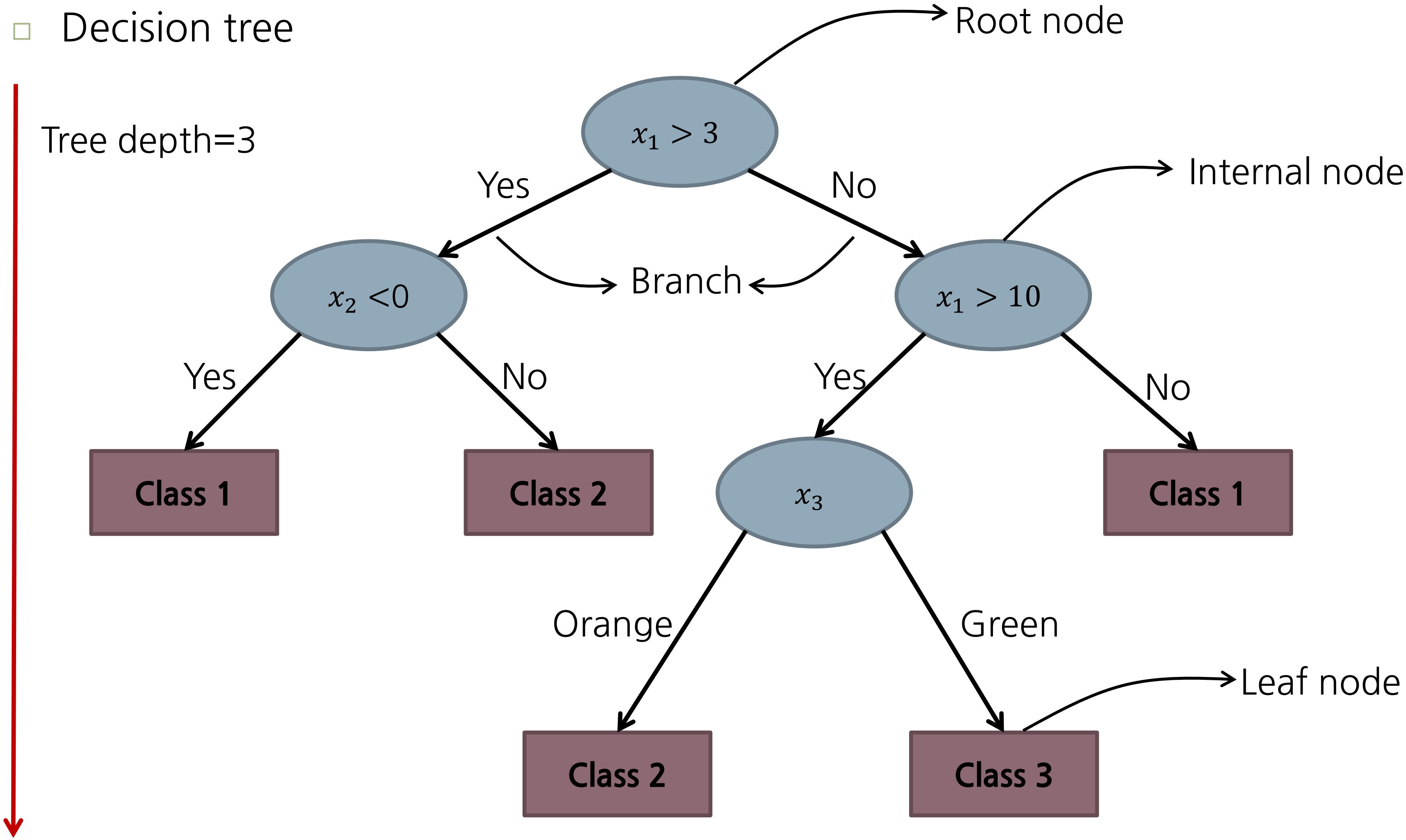
Decision Tree

- Decision tree



Decision Tree

- Decision tree



Decision Tree

- Each root node and internal node represent a specific input variable
  - Root and internal node tests each attribute
    - $x_1 > 1$
    - x_3 is orange

- Each branch corresponds to the result of the test of node
  - Yes/No
  - Values of attribute
    - Orang/Green
    - Long/Short

- Each leaf node assigns a class

In each node, how to choose attribute?
how to split branches?

How to determine which one is the most effective?

⬇

Need some criteria for measure of effectiveness

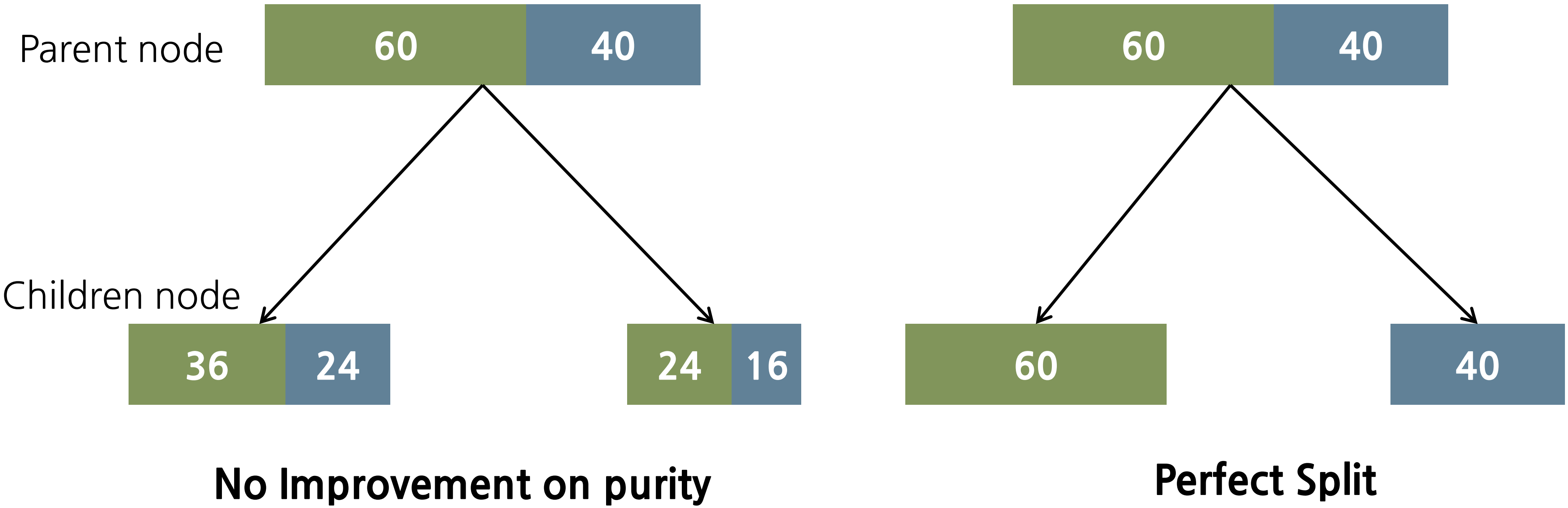
Splitting Criteria

- Categorical target - Classification
  - Entropy
  - Gini impurity

- Continuous target - Regression
  - MSE
  - Friedman MSE
  - MAE

Splitting Criteria: Purity

- Select each split of a node so that in each of the child nodes are purer or less impure than that in the parent node



- Entropy and Gini impurity are measures of impurity
  - Split a node toward decreasing impurity → Maximize reduction in impurity

Entropy

- Expected value of the information
  - The entropy quantifies how "informative" or "surprising" the entire random variable is, averaged on all its possible outcomes
- Entropy H of event X

Information content of X

$$H(X) = E[I(X)] = E[-\ln P(X)]$$

Expectation value Probability function

- For discrete random variable X

$$H(X) = \sum_i P(x_i) I(x_i) = -\sum_i P(x_i) \log_b P(x_i)$$

  - X with possible values of $\{x_1, x_2, \ldots, x_n\}$
  - Commonly b is 2 (10, e are also used)

Example: Entropy

- When you flip one coin(X)
  - Possible output of X: (H), (T)
  - $P(H) = 0.5, P(T) = 0.5$

$$
\begin{aligned}
H(X) &= -P(H)\log_2 P(H) - P(T)\log_2 P(T) \\
&= -0.5\log_2 0.5 - 0.5\log_2 0.5 \\
&= 1
\end{aligned}
$$

- When you flip two coins (X)
  - Possible output of X: (H,H), (H,T), (T,H), (T,T)
  - $P(H,H) = P(H,T) = P(T,H) = P(T,T) = 0.25$

$$
\begin{aligned}
H(X) &= -P(H,H)\log_2 P(H,H) - P(H,T)\log_2 P(H,T) - P(T,H)\log_2 P(T,H) - P(T,T)\log_2 P(T,T) \\
&= -4 \times 0.25\log_2 0.25 \\
&= 2
\end{aligned}
$$

How to Define Effectiveness of Split

- If split is effective, information gain is large
  - Information gain=reduction of uncertainty

$$IG(T, a) = H(T) - H(T|a)$$

Information gain with the split on attribute a

Entropy of original set

Entropy of new state after split

  - Entropy of new state after split=normalized sum of entropy of split sets

$$H(T|a) = \sum_{i}^{n} \frac{|T_i'|}{|T|} \cdot H(T_i')$$

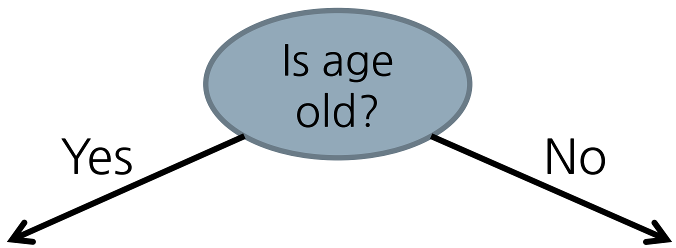
    - T is split to $T_1', T_2', \ldots, T_n'$

Example: Calculate Information Gain through Entropy

- The node is split by age to predict profit of company

Age(x)	old	old	old	mid	mid	mid	mid	new	new	new
Profit(y)	down	down	down	down	down	up	up	up	up	up

$$H(T) = -P(down)\log_2 P(down) - P(up)\log_2 P(up)$$
$$= -2 \times 0.5 \log_2 P(0.5) = 1$$



Age(x)	old	old	old
Profit(y)	down	down	down

$$H(T_1') = -1\log_2 1 = 0$$

Age(x)	mid	mid	mid	mid	new	new	new
Profit(y)	down	down	up	up	up	up	up

$$H(T_2') = -\frac{2}{7}\log_2\frac{2}{7} - \frac{5}{7}\log_2\frac{5}{7} \approx 0.86$$

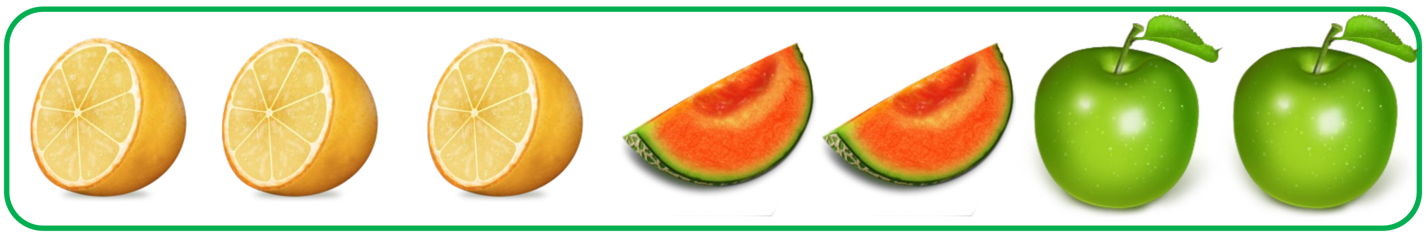
$$\boldsymbol{IG = H(\text{before}) - H(\text{after}) = 1 - 0.86 \times \frac{7}{10} = 1 - 0.602 = 0.398}$$

Gini Impurity

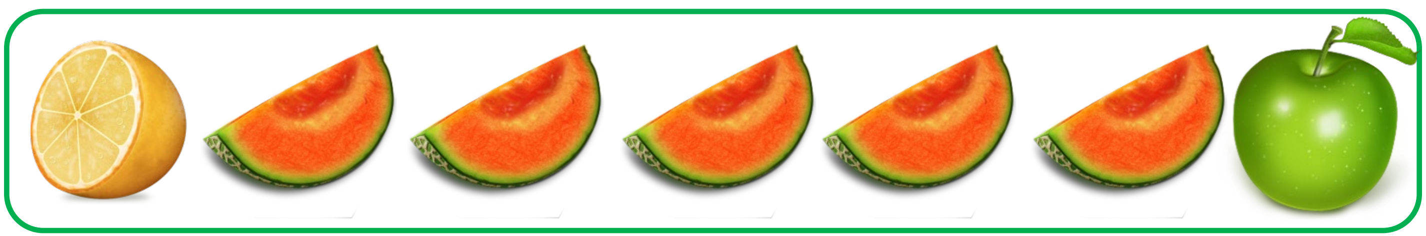
- Gini Impurity: measure of impurity

$$G(T) = \sum_{i \neq j} P(i|T)P(j|T) = 1 - \sum_j P(j|T)^2 = 1 - \sum_j \left(\frac{n_j(T)}{n(T)}\right)^2$$

  - $P(j|t)$ is the probability of output j in node T
  - $n(t)$ is the total number of samples in node T
  - $n_j(t)$ is the number of samples with output j in node T



$$G = 1 - \left(\frac{3}{7}\right)^2 - \left(\frac{2}{7}\right)^2 - \left(\frac{2}{7}\right)^2 = \frac{32}{49}$$



$$G = 1 - \left(\frac{1}{7}\right)^2 - \left(\frac{5}{7}\right)^2 - \left(\frac{1}{7}\right)^2 = \frac{22}{49}$$

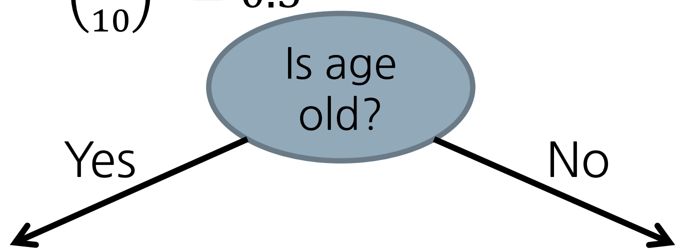
Example: Calculate Information Gain through Gini Impurity

- The node is split by age to predict profit of company

Age(x)	old	old	old	mid	mid	mid	mid	new	new	new
Profit(y)	down	down	down	down	down	up	up	up	up	up

$$G(T) = 1 - P^2(down) - P^2(up)$$

$$= 1 - \left(\frac{5}{10}\right)^2 - \left(\frac{5}{10}\right)^2 = 0.5$$



Age(x)	old	old	old
Profit(y)	down	down	down

$$G(T_1') = 1 - \left(\frac{3}{3}\right)^2 = 0$$

Age(x)	mid	mid	mid	mid	new	new	new
Profit(y)	down	down	up	up	up	up	up

$$G(T_2') = 1 - \left(\frac{2}{7}\right)^2 - \left(\frac{5}{7}\right)^2 \approx 0.41$$

$$\boldsymbol{IG = G(\text{before}) - G(\text{after}) = 0.5 - 0.41 \times \frac{7}{10} = 0.5 - 0.287 = 0.213}$$

Question

- Predict profit of companies based on age of company, type of company, and competition status

Age	Competition	Type	Profit
old	yes	S/W	down
old	no	S/W	down
old	no	H/W	down
mid	yes	S/W	down
mid	yes	H/W	down
mid	no	H/W	up
mid	no	S/W	up
new	yes	S/W	up
new	no	H/W	up
new	no	S/W	up

1) How much information gain based on Entropy is obtained with the splitting on competition?

2) How much information gain based on Gini impurity is obtained with the splitting on type of company?

Decision Tree Algorithm: C4.5

- Procedure of C4.5

1
- Check for terminal conditions

2-1
- (No fulfilled terminal conditions)For each input variable x, calculate the normalized information gain from splitting on x

2-2
- (Any terminal condition is fulfilled) Create leaf node and move onto step 1 with different internal node

3
- Create a decision node that splits on the input variable with the highest normalized information gain

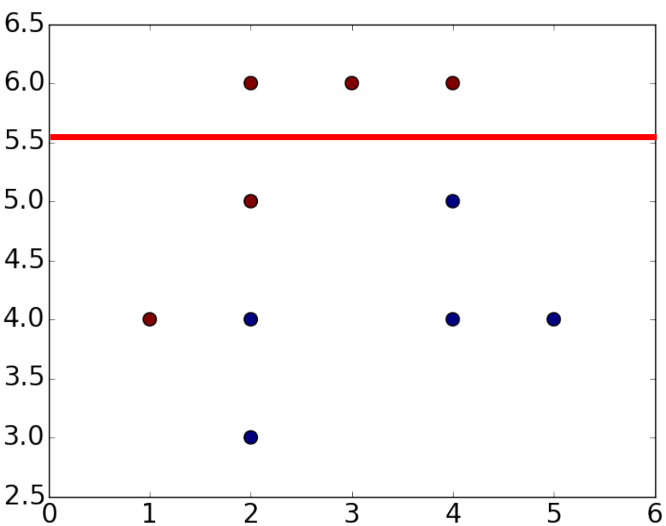
4
- Repeat this process until there is no internal node

Terminal conditions

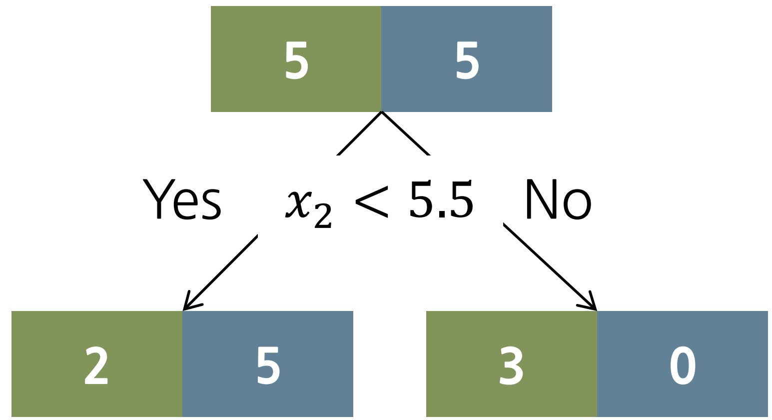
1. **All the samples in the list belong to the same class**
2. **None of the features provide any information gain**

Simple Example for Tree

Class	x_1	x_2
1	1	4
1	2	6
1	2	5
0	2	4
0	2	3
1	3	6
1	4	6
0	4	5
0	4	4
0	5	4



$$G = 1 - \left(\frac{5}{10}\right)^2 - \left(\frac{5}{10}\right)^2 = 0.5$$



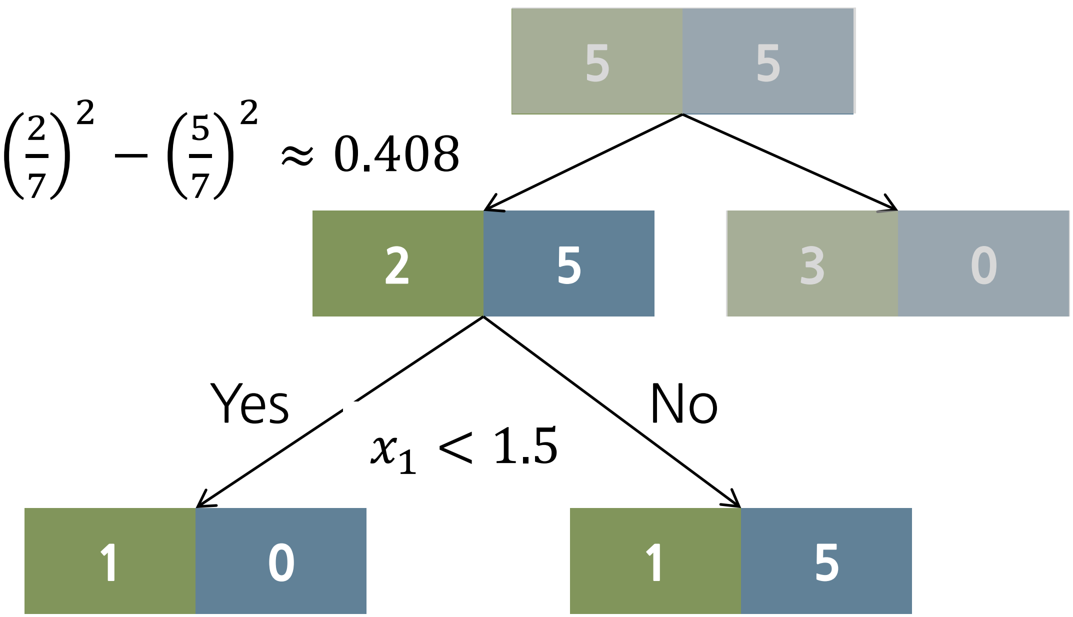
$$G = 1 - \left(\frac{2}{7}\right)^2 - \left(\frac{5}{7}\right)^2 \approx 0.4083 \qquad G = 1 - \left(\frac{3}{3}\right)^2 - \left(\frac{0}{3}\right)^2 = 0$$

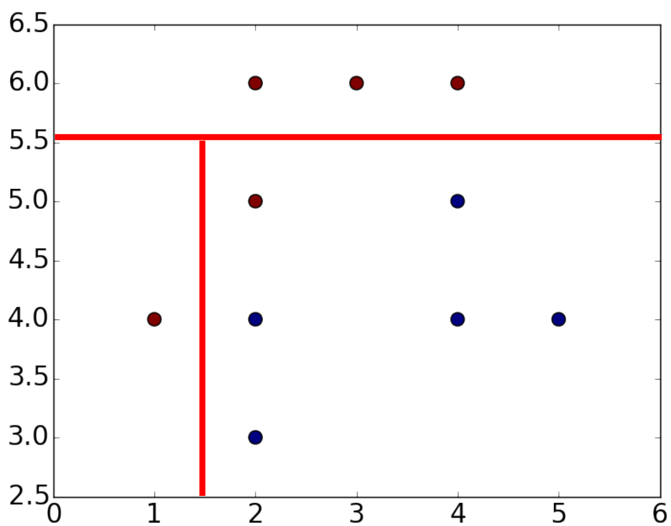
$$IG = 0.5 - 0.408 \times \frac{7}{10} - 0 \times \frac{3}{10} = 0.2144$$

Simple Example for Tree

Class	x_1	x_2
1	1	4
1	2	6
1	2	5
0	2	4
0	2	3
1	3	6
1	4	6
0	4	5
0	4	4
0	5	4

$$G = 1 - \left(\frac{2}{7}\right)^2 - \left(\frac{5}{7}\right)^2 \approx 0.408$$



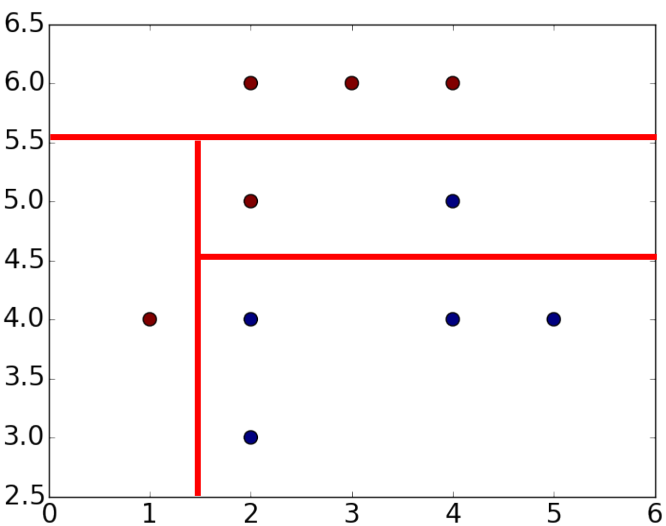


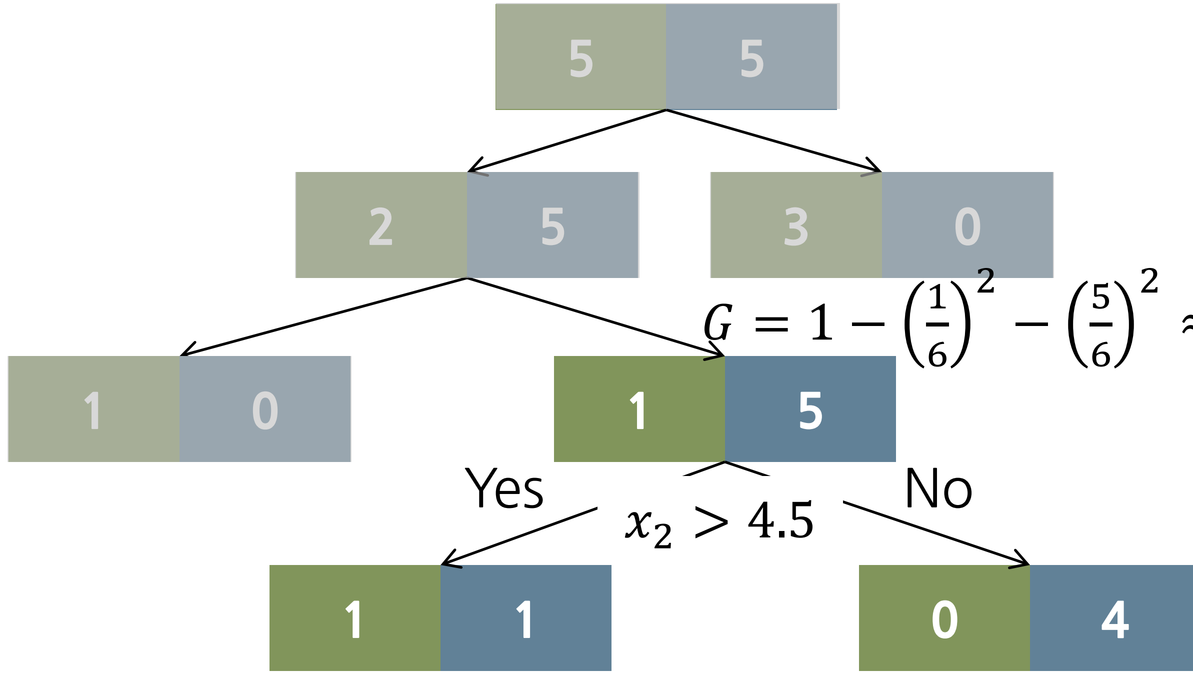
$$G = 1 - \left(\frac{1}{1}\right)^2 - \left(\frac{0}{1}\right)^2 = 0 \qquad G = 1 - \left(\frac{1}{6}\right)^2 - \left(\frac{5}{6}\right)^2 \approx 0.278$$

$$IG = \left(0.408 - 0 \times \frac{1}{7} - 0.278 \times \frac{6}{7}\right) \times \frac{7}{10} \approx 0.11$$

Simple Example for Tree

Class	x_1	x_2
1	1	4
1	2	6
1	2	5
0	2	4
0	2	3
1	3	6
1	4	6
0	4	5
0	4	4
0	5	4





$$G = 1 - \left(\frac{1}{6}\right)^2 - \left(\frac{5}{6}\right)^2 \approx 0.278$$

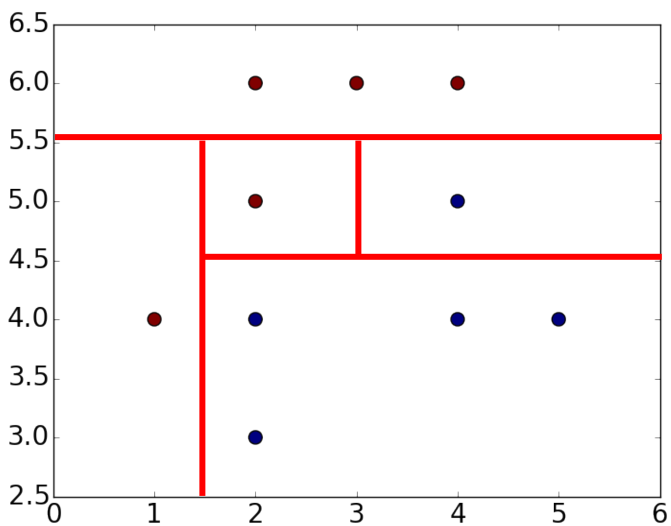
$$G = 1 - \left(\frac{1}{2}\right)^2 - \left(\frac{1}{2}\right)^2 = 0.5$$

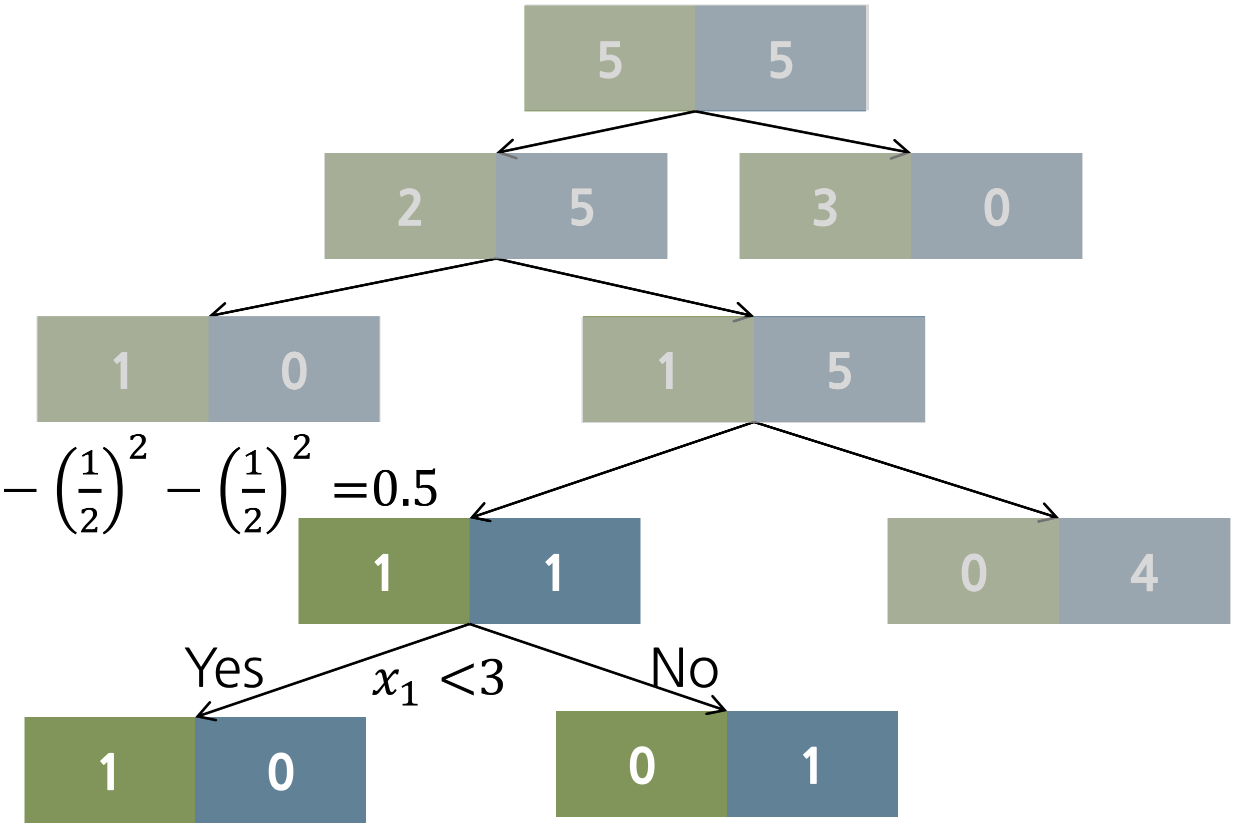
$$G = 1 - \left(\frac{0}{4}\right)^2 - \left(\frac{4}{4}\right)^2 = 0$$

$$IG = \left(0.278 - 0.5 \times \frac{2}{6} - 0 \times \frac{4}{6}\right) \times \frac{6}{10} \approx 0.067$$

Simple Example for Tree

Class	x_1	x_2
1	1	4
1	2	6
1	2	5
0	2	4
0	2	3
1	3	6
1	4	6
0	4	5
0	4	4
0	5	4



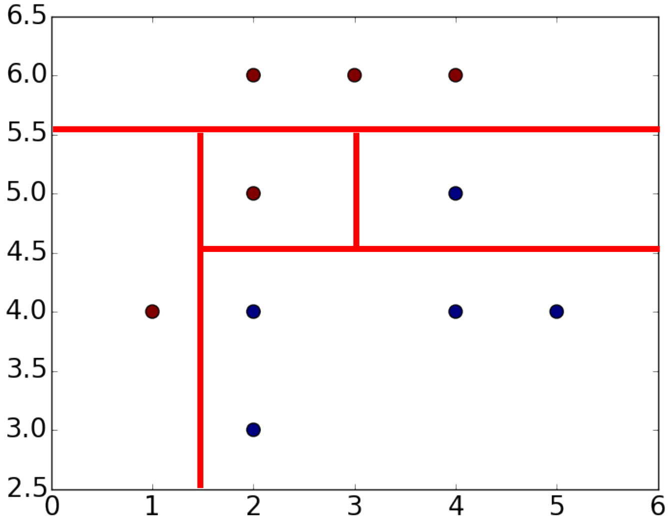


$$G = 1 - \left(\frac{1}{2}\right)^2 - \left(\frac{1}{2}\right)^2 = 0.5$$

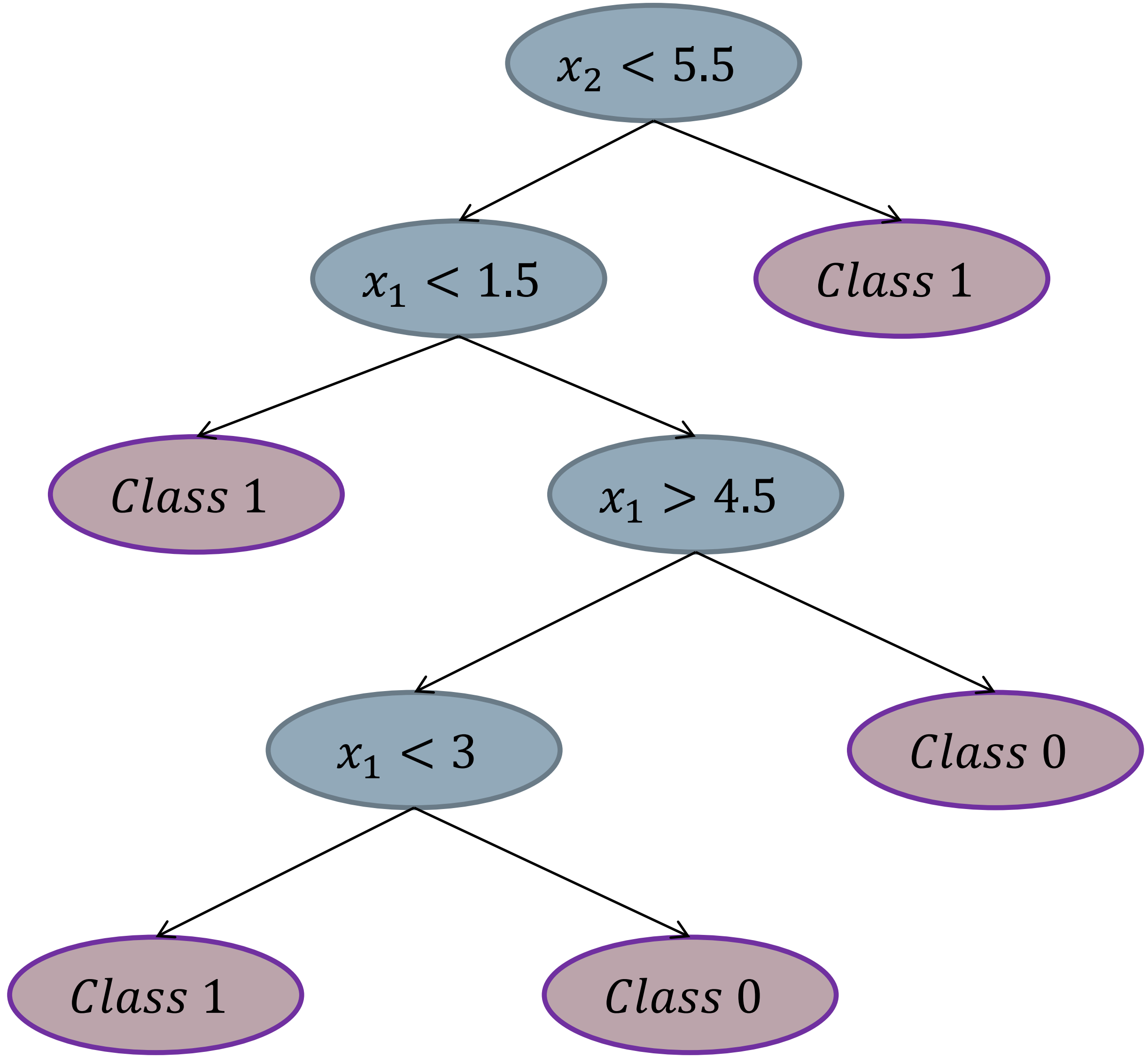
$$G = 1 - \left(\frac{1}{1}\right)^2 - \left(\frac{0}{1}\right)^2 = 0 \qquad G = 1 - \left(\frac{0}{1}\right)^2 - \left(\frac{1}{1}\right)^2 = 0$$

$$IG = \left(0.5 - 0 \times \frac{1}{2} - 0 \times \frac{1}{2}\right) \times \frac{2}{10} = 0.1$$

Simple Example for Tree



Step	Impurity Change	Total impurity
0	0.000	0.500
1	0.214	0.286
2	0.119	0.167
3	0.067	0.100
4	0.100	0.000



When Does Tree Stop Growing?

- Growing full-size tree can cause overfitting
  - Low classification accuracy on test set

- Introduce pruning step after growing tree
  - Pruning simplifies the tree by trimming some branches of the fully grown tree
  - Generate several pruned trees and select best tree

- There are several popular pruning algorithms
  - Reduced error pruning: Starting at leaf node, each node is replaced with its most popular class and if the prediction accuracy is not affected then the change is kept
  - Cost complexity pruning: Cost complexity pruning removes subtree based on cost complexity measure at each step

Cost Complexity Pruning

- Misclassification cost at node t

$$r(t) = \min_i \sum_{k=1}^{K} C(i|k)p(k|t) \;\rightarrow\; r(t) = 1 - \max_k p(k|t)$$

  - $C(i|k) = \begin{cases} 1, & if\ i \neq k \\ 0, & if\ i = k \end{cases}$
  - $p(k|t)$ is probability that class of data point is k given that it is in node t

- Misclassification cost at tree T

$$R(T) = \sum_{t \in \tilde{T}} r(t)p(t) = \sum_{t \in \tilde{T}} R(t)$$

  - $\tilde{T}$ is set of terminal nodes of tree T
  - $p(t)$ is probability that data point is in node t
  - Set $R(t) = r(t)p(t)$

Cost Complexity Pruning

- Cost complexity measure

Complexity parameter

$$R_\alpha(T) = R(T) + \alpha|T|$$

$|T|$ is tree complexity=the number of terminal nodes

- For a terminal node t

$$R_\alpha(t) = R(t) + \alpha$$

- For a subtree at t

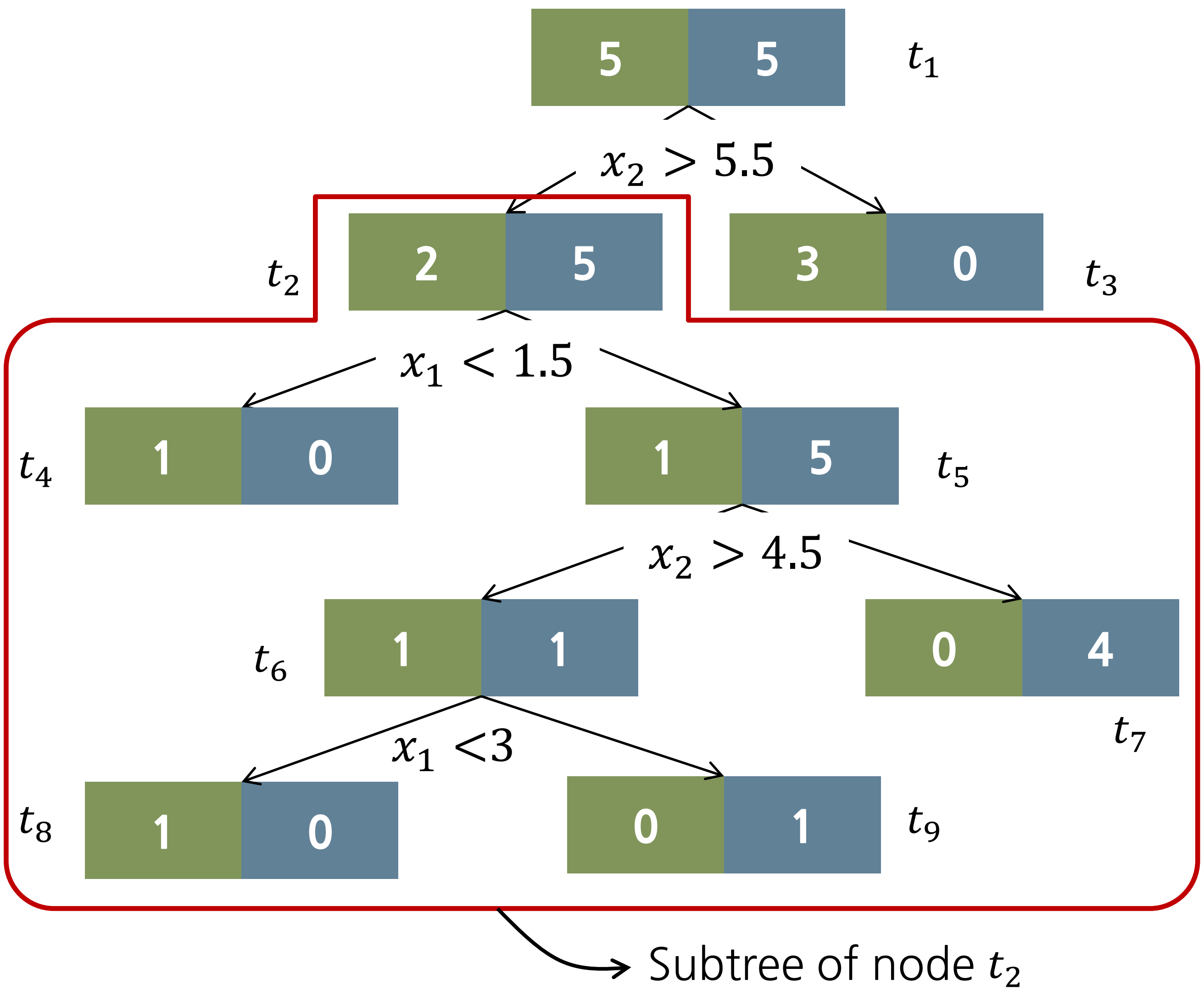
$$R_\alpha(T_t) = R(T_t) + \alpha|T_t|$$

- Cost complexity pruning prunes subtree at t which minimizes

$$\alpha(t) = \frac{R(t) - R(T_t)}{|T_t| - 1}$$

- Prune subtree if $R_\alpha(t) \leq R_\alpha(T_t)$

Cost Complexity Pruning



At node t_2

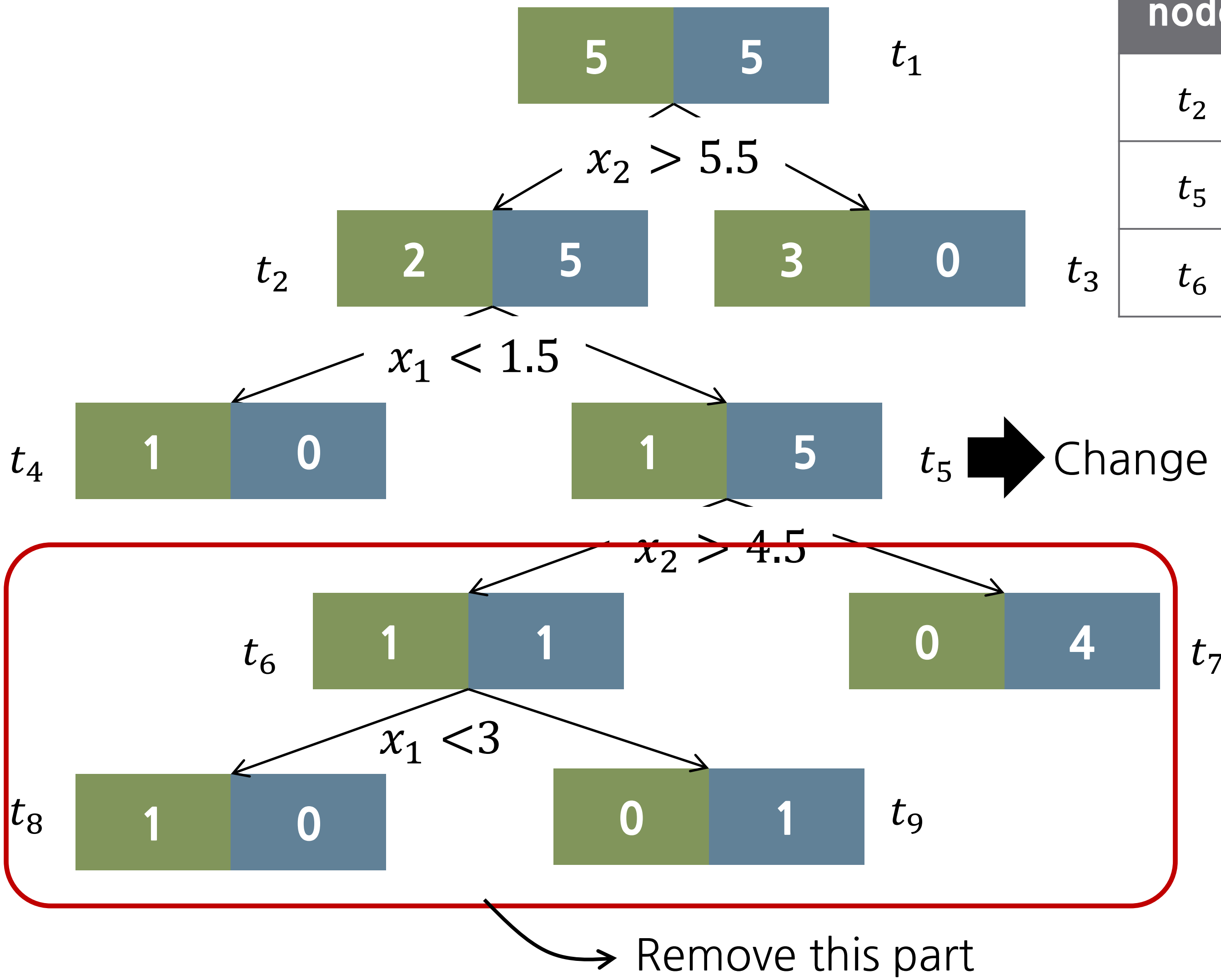
$$R(t_2) = r(t_2)p(t_2)$$
$$= \left(1 - \frac{5}{7}\right) \times \frac{7}{10} = 0.2$$

$$|T_{t_2}| = 4$$

$$R(T_{t_2})$$
$$= (1-1) \times \frac{1}{10} + (1-1) \times \frac{1}{10}$$
$$+ (1-1) \times \frac{1}{10} + (1-1) \times \frac{4}{10}$$
$$= 0$$

$$\alpha(t_2) = \frac{0.2 - 0}{4 - 1} \approx 0.067$$

Cost Complexity Pruning



node	$R(t)$	$R(T_t)$	$\lvert T_t \rvert$	$\alpha(t)$
t_2	0.2	0	4	0.067
t_5	0.1	0	3	**0.050**
t_6	0.1	0	2	0.100

Pros and Cons of Decision Tree

- Pros
  - Easy interpretation
  - Non-parametric approach
  - Inherently non-linear
  - Easy to handle categorical variables
  - Implicitly perform feature selection

- Cons
  - Large computing cost
  - Lack of linearity or main effects
  - Each node only considers single variable
    - Many algorithms has been proposed to overcome this problem

Regression Tree

- CART
  - Classification And Regression Tree (CART) is one of decision tree algorithms
  - Decision tree can be also used for regression analysis

How?

Regression Tree

- New split rule
  - Entropy and Gini impurity are not appropriate split measure for regression analysis
  - MSE (Mean Squared Error)
    - The split that most decreases the MSE is selected

$$\hat{y}_i = \frac{\sum_{j \in t_i} y_j}{|t_i|}$$

$$R(t_i) = \frac{1}{N_{t_i}} \sum_{j \in t_i} \left(y_j - \hat{y}_i\right)^2$$

$$IG = p(t_p)R\left(t_p\right) - p(t_r)R(t_r) - p(t_l)R(t_l)$$

  - Friedman MSE

$$\hat{y}_i = \frac{\sum_{j \in t_i} y_j}{|t_i|}$$

$$IG = \frac{N_{t_r} N_{t_l}}{N_{t_r} + N_{t_l}} \left(\hat{y}_{t_r} - \hat{y}_{t_l}\right)^2$$

Regression Tree

- New split rule
  - MAE (Mean Absolute Error)
    - The split that minimizes the L1 loss using the median of each terminal node is selected

$$\hat{y}_i = the\ median\ of\ each\ terminal\ node$$

$$R(t_i) = \frac{1}{N_{t_i}} \sum_{j \in t_i} \left| y_j - \hat{y}_i \right|$$

$$IG = p(t_p)R\left(t_p\right) - p(t_r)R(t_r) - p(t_l)R(t_l)$$

Regression Tree

- Drawback of regression tree