UNIVERSITI SAINS MALAYSIA

# MACHINE LEARNING CDS503

Topic 4: Decision Tree

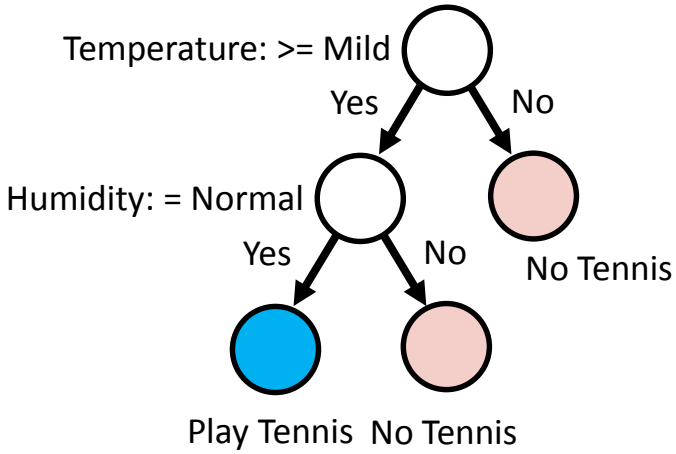
Mohd Halim Mohd Noor, PhD

# Outline

- Introduction
- Building Decision Trees
  - Classification Trees
  - Regression Trees
- Pruning Decision Trees
  - Pre-pruning
  - Post-pruning (Cost Complexity)

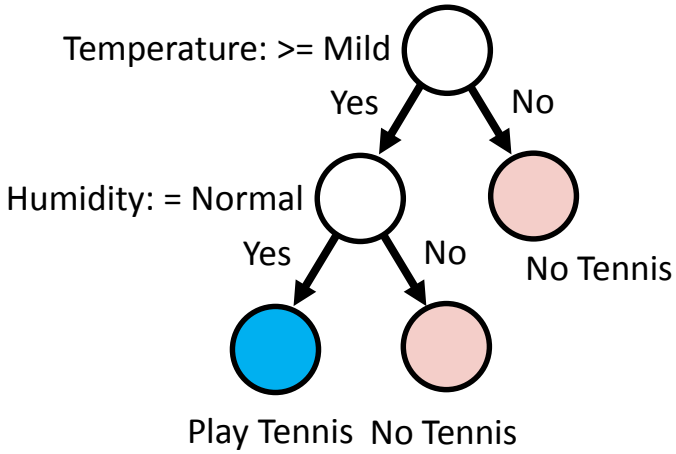
# Introduction

- Hierarchical model that composes of internal decision nodes, $m$ and leaf nodes
- Each decision node $m$ implements a test with discrete outcomes labelling the branches



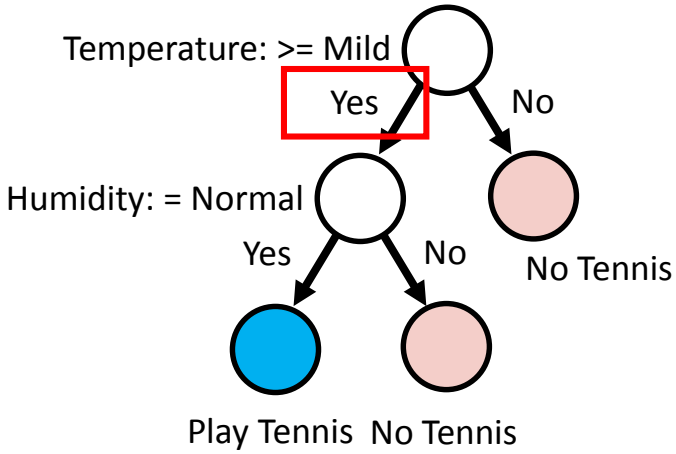
# Introduction

- Given an input
  - Input: Humidity = Normal, Temperature = Hot



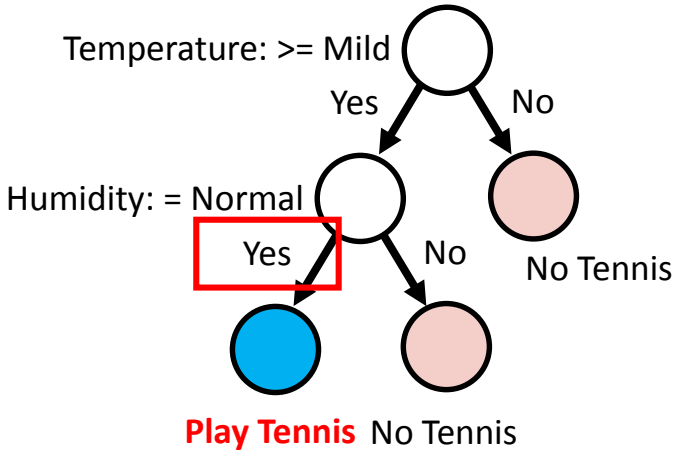
# Introduction

- Given an input
  - Input: Humidity = Normal, Temperature = Hot

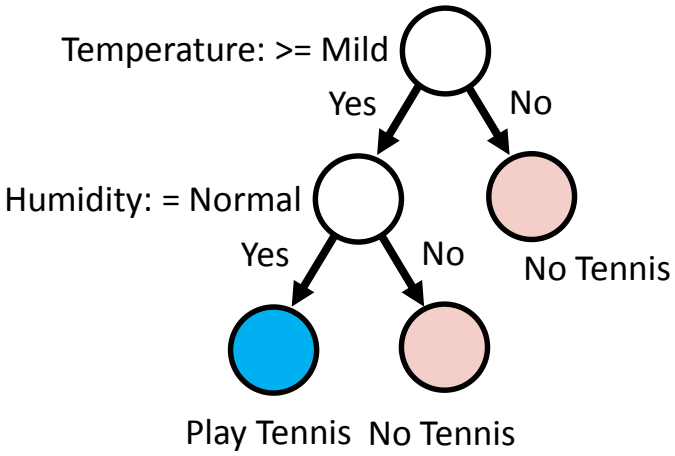


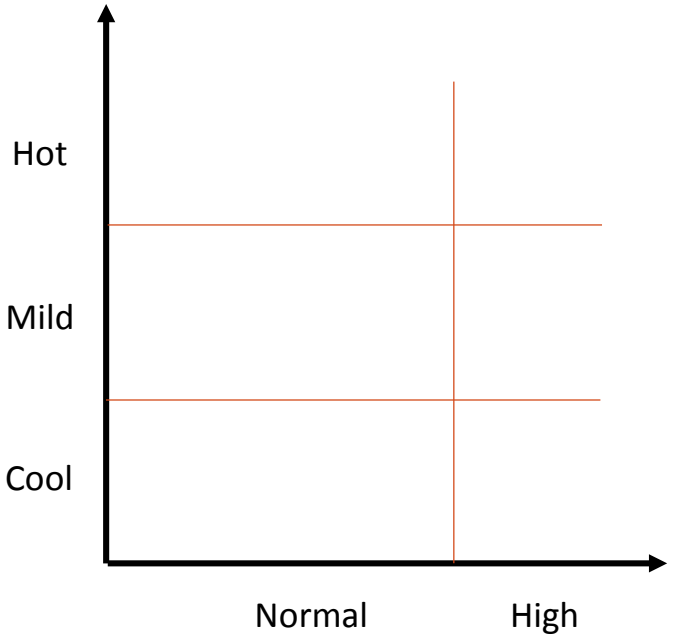
# Introduction

- Given an input
  - Input: Humidity = Normal, Temperature = Hot

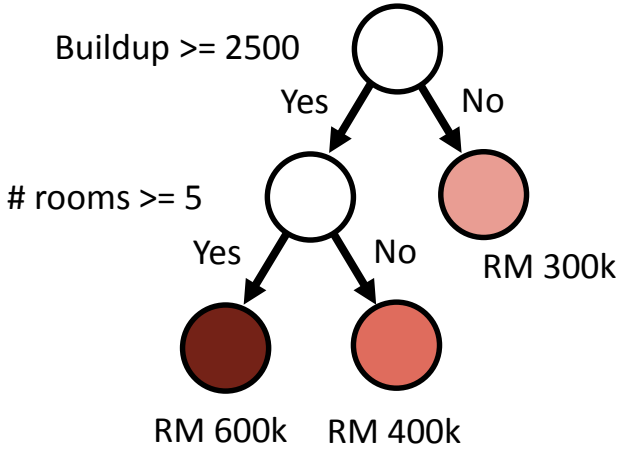


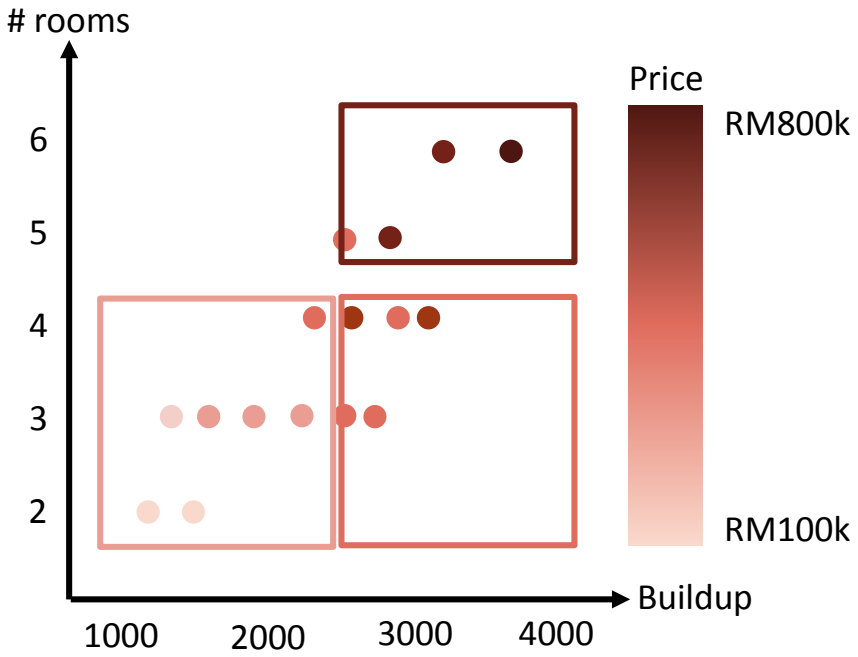
# Introduction (Classification)



# Introduction (Regression)



# Building Decision Trees

# Building a Decision Tree

- How do we obtain a good split?
- How do we quantify the goodness of a split?
- Information gain

$$IG = I(D_p) - \frac{N_l}{N_p} I(D_l) - \frac{N_r}{N_p} I(D_r)$$

- $I$ is impurity – measure of homogeneity of the class at the node
- $N_i$ is the number of samples in node $i$

- Impurity:
  - Classification error
  - Entropy
  - Gini

# Classification Error

- $E = 1 - \max p_j$
- $p_j$ is probability of class $j$

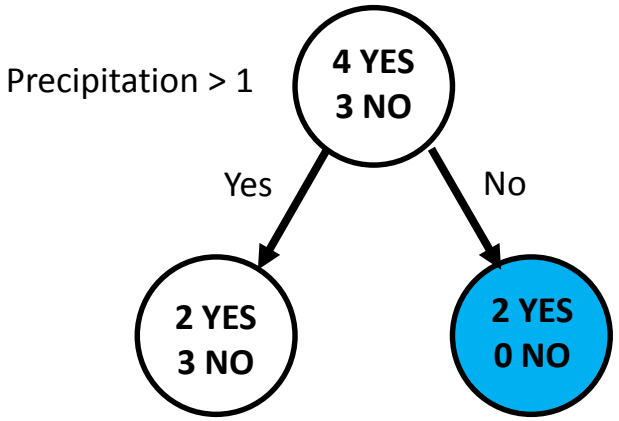
# Classification Error

- Precipitation, mm

| Precipitation | Temperature | Humidity | Play Tennis |
|---|---|---|---|
| 5 | 33 | 71 | No |
| 6 | 22 | 68 | No |
| 3 | 31 | 65 | Yes |
| 1 | 25 | 70 | Yes |
| 0 | 15 | 50 | Yes |
| 2 | 20 | 45 | Yes |
| 2 | 27 | 75 | No |

# Classification Error

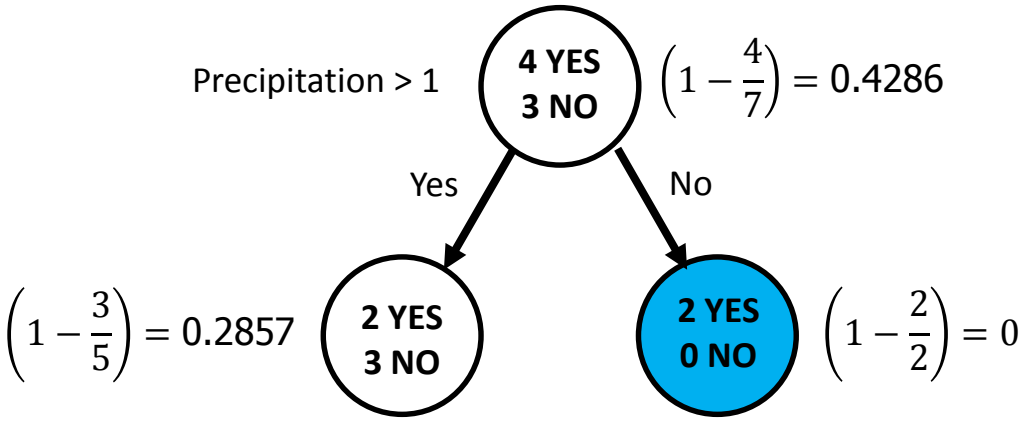
- Precipitation, mm




| Precipitation | Temperature | Humidity | Play Tennis |
|---|---|---|---|
| 5 | 33 | 71 | No |
| 6 | 22 | 68 | No |
| 3 | 31 | 65 | Yes |
| 1 | 25 | 70 | Yes |
| 0 | 15 | 50 | Yes |
| 2 | 20 | 45 | Yes |
| 2 | 27 | 75 | No |

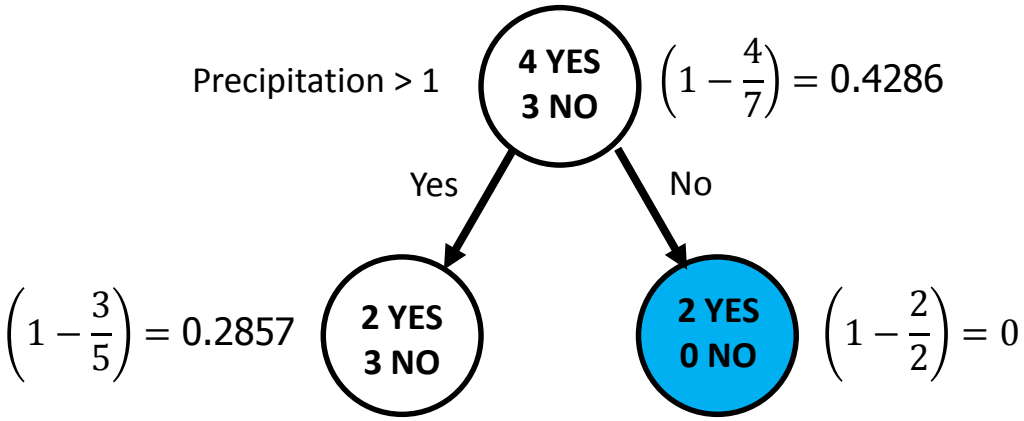
# Classification Error

- Precipitation, mm



# Classification Error

- Precipitation, mm

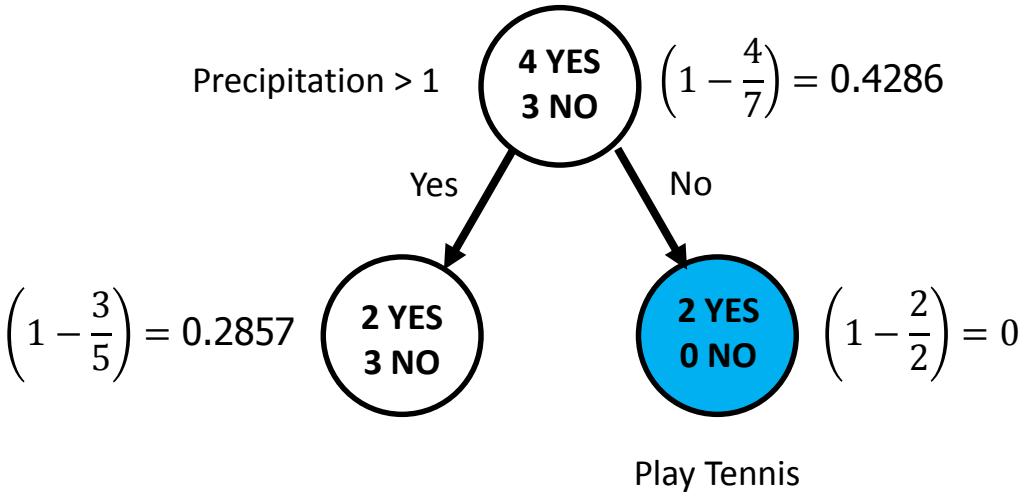



$$0.4286-\frac{2}{7}(0)-\frac{5}{7}(0.2857)=0.2245$$

We gain information by 0.2245.

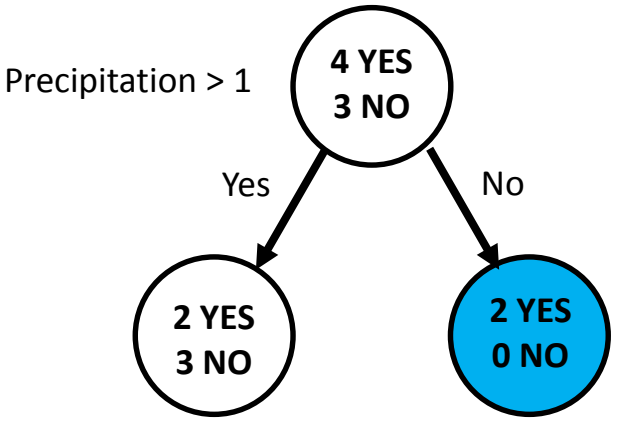
# Classification Error

- Precipitation, mm



# Classification Error

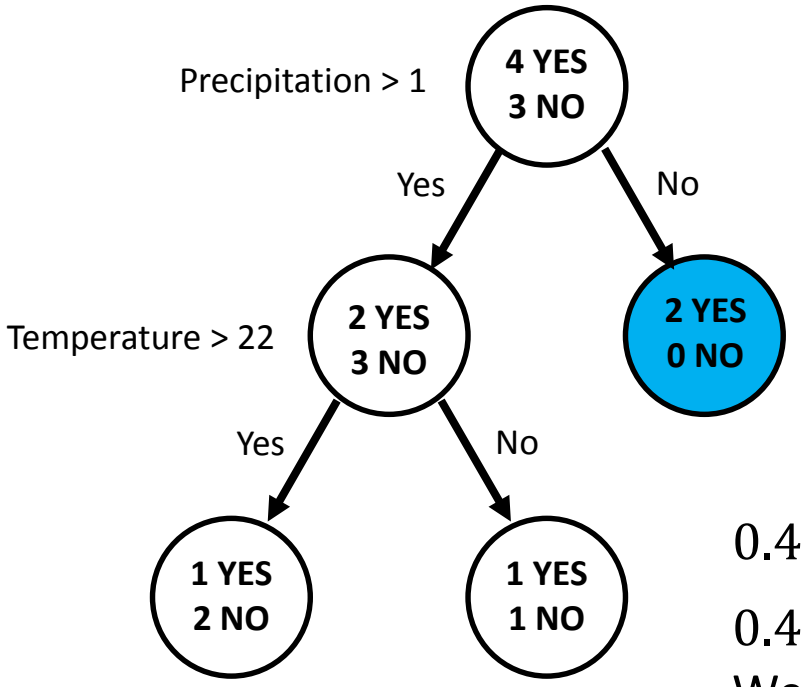
- Precipitation, mm




| Precipitation | Temperature | Humidity | Play Tennis |
|---|---|---|---|
| 5 | 33 | 71 | No |
| 6 | 22 | 68 | No |
| 3 | 31 | 65 | Yes |
| ~~1~~ | ~~25~~ | ~~70~~ | ~~Yes~~ |
| ~~0~~ | ~~15~~ | ~~50~~ | ~~Yes~~ |
| 2 | 20 | 45 | Yes |
| 2 | 27 | 75 | No |

# Classification Error

- Temperature



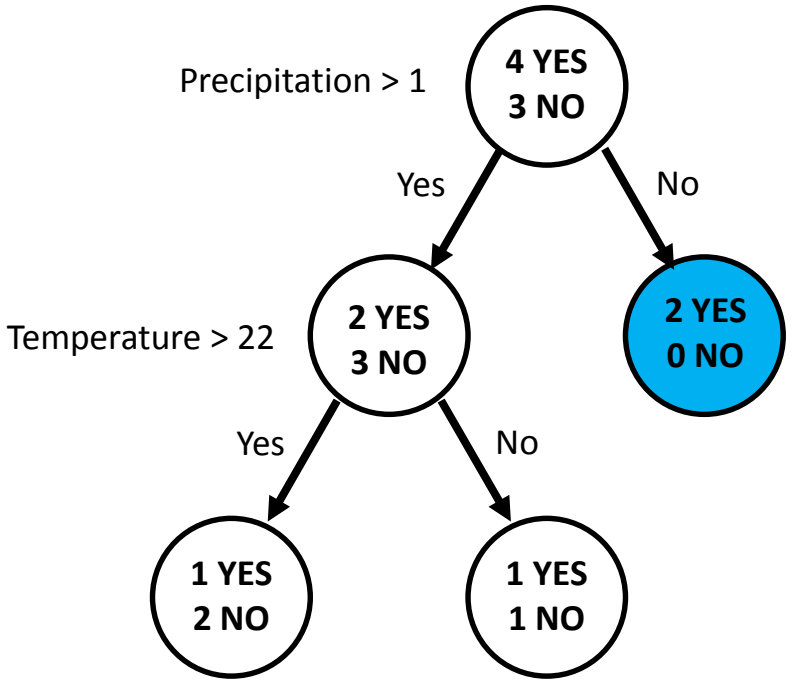

$$0.4 - \frac{3}{5}(0.333) - \frac{2}{5}(0.5)$$

$$0.4 - 0.2 - 0.2 = 0$$

We gain information by 0 (no change in classification error).

# Classification Error

- The tree algorithm will stop at this point



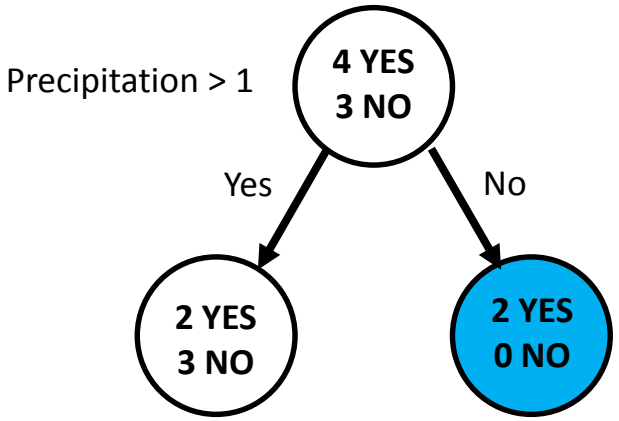

Stuck with non-pure leaves (the nodes are not homogeneous)

# Entropy

- Measure of randomness or uncertainty
- $H = -\sum_j p_j log_2 p_j$

# Entropy

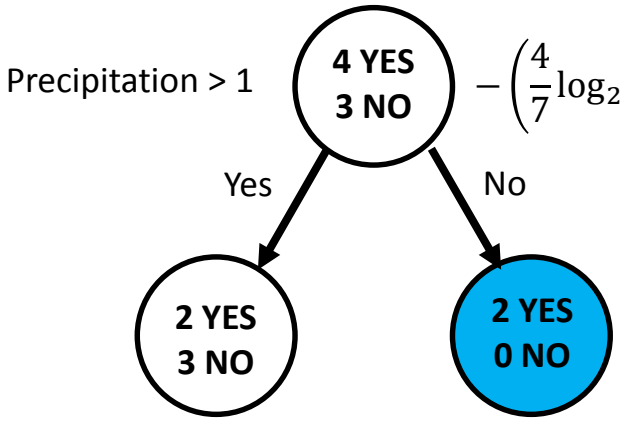
- Precipitation, mm




| Precipitation | Temperature | Humidity | Play Tennis |
|---|---|---|---|
| 5 | 33 | 71 | No |
| 6 | 22 | 68 | No |
| 3 | 31 | 65 | Yes |
| 1 | 25 | 70 | Yes |
| 0 | 15 | 50 | Yes |
| 2 | 20 | 45 | Yes |
| 2 | 27 | 75 | No |

# Entropy

- Precipitation, mm



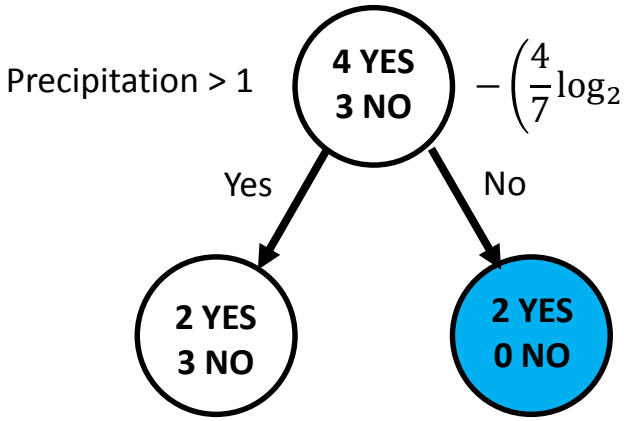

$$-\left(\frac{4}{7}\log_2\frac{4}{7}+\frac{3}{7}\log_2\frac{3}{7}\right)=0.9852$$

$$-\left(\frac{2}{5}\log_2\frac{2}{5}+\frac{3}{5}\log_2\frac{3}{5}\right)=0.9710$$

$$-\left(\frac{2}{2}\log_2\frac{2}{2}+\frac{0}{2}\log_2\frac{0}{2}\right)=0$$

# Entropy

- Precipitation, mm




$$-\left(\frac{4}{7}\log_2\frac{4}{7}+\frac{3}{7}\log_2\frac{3}{7}\right)=0.9852$$

$$-\left(\frac{2}{5}\log_2\frac{2}{5}+\frac{3}{5}\log_2\frac{3}{5}\right)=0.9710$$

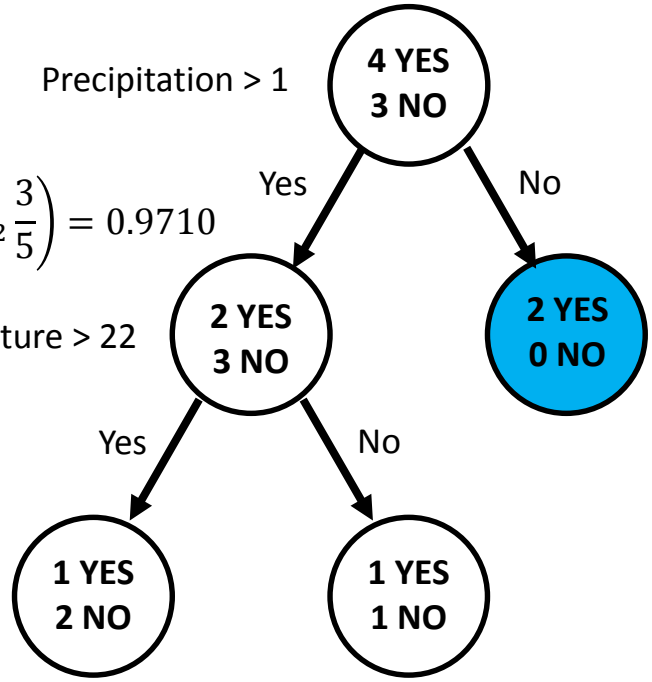
$$-\left(\frac{2}{2}\log_2\frac{2}{2}+\frac{0}{2}\log_2\frac{0}{2}\right)=0$$

$$0.9852-\frac{5}{7}(0.9710)-\frac{2}{7}(0)=0.2916$$

We gain information by 0.2916.

# Entropy

- Temperature




$$-\left(\frac{2}{5}\log_2\frac{2}{5}+\frac{3}{5}\log_2\frac{3}{5}\right)=0.9710$$

$$-\left(\frac{1}{2}\log_2\frac{1}{2}+\frac{2}{3}\log_2\frac{2}{3}\right)=0.8900$$

$$-\left(\frac{1}{2}\log_2\frac{1}{2}+\frac{1}{2}\log_2\frac{1}{2}\right)=1$$

$$0.9710-\frac{3}{5}(0.9710)-\frac{2}{5}(1)$$

$$0.9710-0.534-0.400=0.0370$$

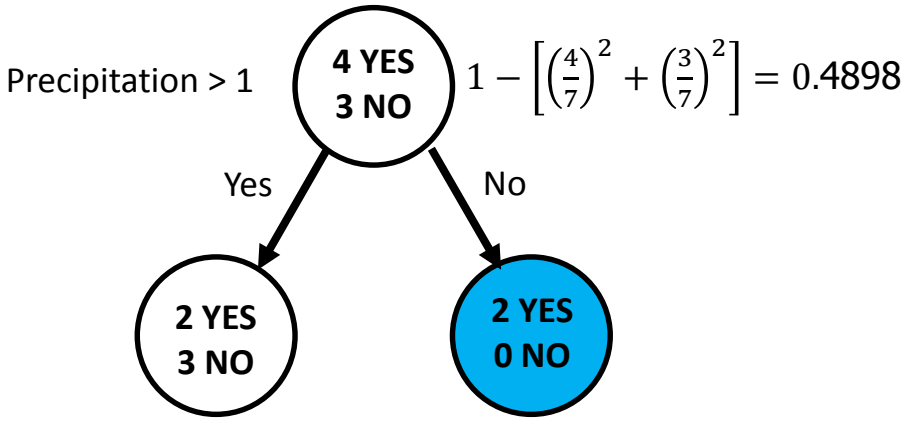
We gain information by 0.0370
(small change in entropy)

# Gini

- Measures the probability of a variable being wrongly classified when it is randomly chosen
- $H = 1 - \sum_j p_j^2$

# Gini

- Precipitation, mm




$1 - \left[\left(\frac{4}{7}\right)^2 + \left(\frac{3}{7}\right)^2\right] = 0.4898$

$1 - \left[\left(\frac{2}{5}\right)^2 + \left(\frac{3}{5}\right)^2\right] = 0.4800$

$1 - \left[\left(\frac{2}{2}\right)^2 + \left(\frac{0}{2}\right)^2\right] = 0.0$

$$0.4898 - \frac{5}{7}(0.4800) - \frac{2}{7}(0) = 0.1469$$

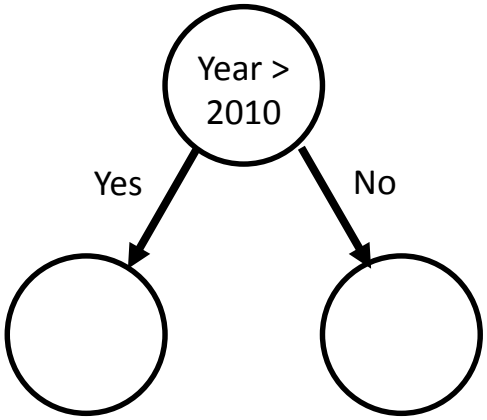
We gain information by 0.1469.

# Regression Trees

- Each leaf node has numeric output instead of category or class
- Goodness of a split is quantified using sum of squared errors
- $SSE = \sum_j \sum_{i \in c_j} \left( y_i - \mu_{c_j} \right)^2$

# Regression Trees

- Year

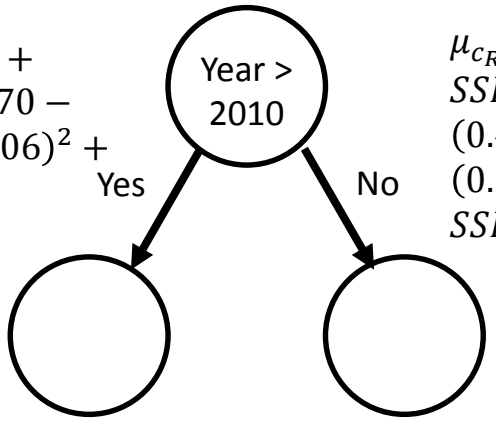



| Year | # rooms | Buildup Area | Price (in Mil) |
|---|---|---|---|
| 2010 | 3 | 1500 | 0.20 |
| 2015 | 4 | 2000 | 0.35 |
| 2012 | 3 | 1450 | 0.25 |
| 2000 | 3 | 1600 | 0.15 |
| 2018 | 4 | 1900 | 0.40 |
| 2014 | 3 | 1450 | 0.27 |
| 2008 | 5 | 2500 | 0.45 |
| 2011 | 3 | 1600 | 0.26 |

# Regression Trees

- Year

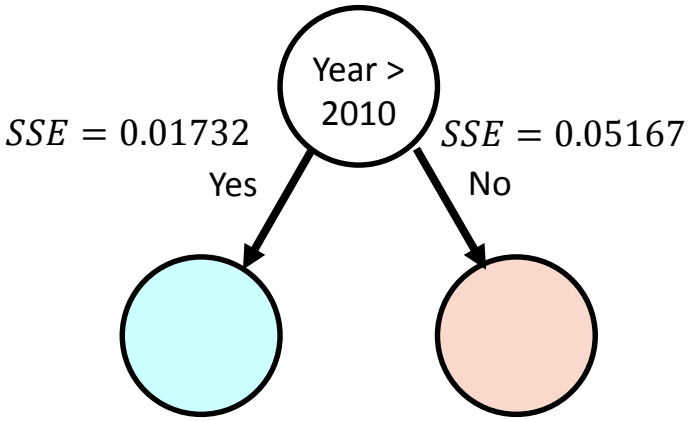
$\mu_{c_L} = 0.306$
$SSE = (0.26 - 0.306)^2 + (0.25 - 0.306)^2 + (0.270 - 0.306)^2 + (0.350 - 0.306)^2 + (0.40 - 0.306)^2$
$SSE = 0.01732$




$\mu_{c_R} = 0.267$
$SSE = (0.15 - 0.267)^2 + (0.45 - 0.267)^2 + (0.20 - 0.267)^2$
$SSE = 0.05167$

# Regression Trees

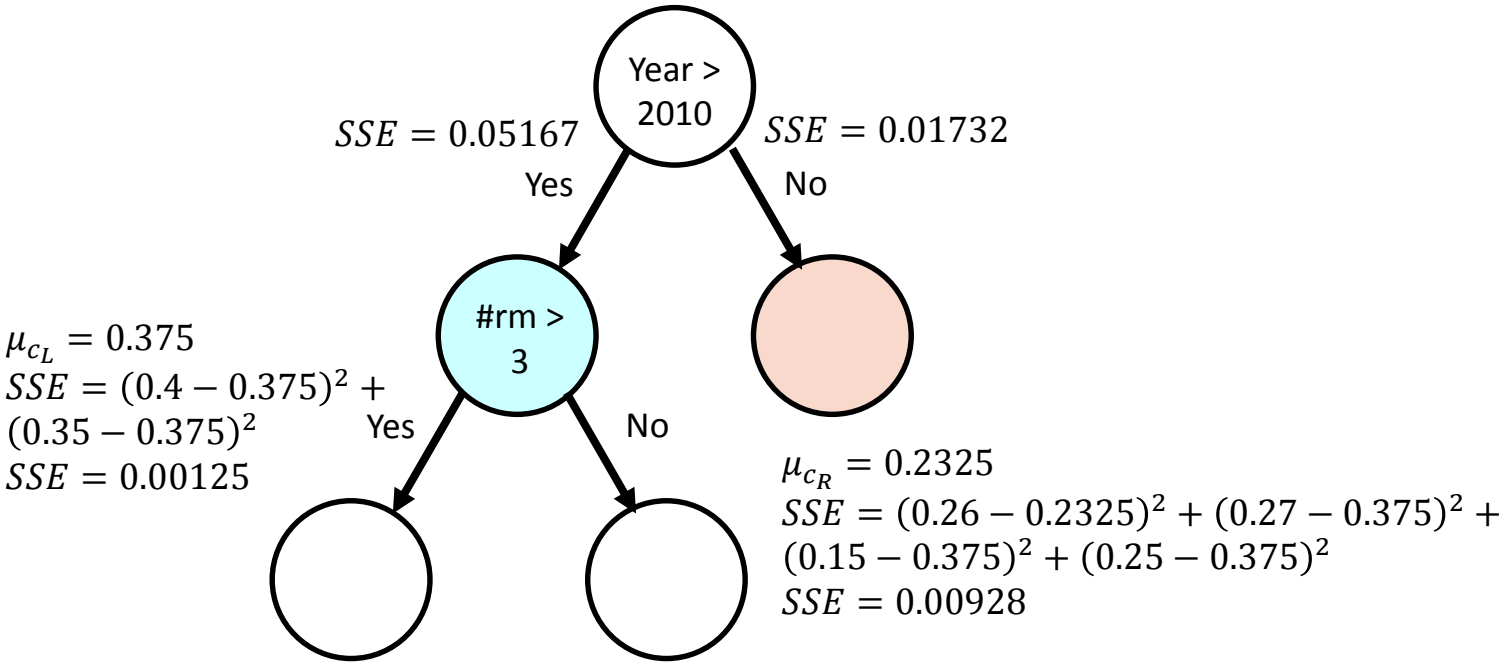
- Year




| Year | # rooms | Buildup Area | Price (in Mil) |
|---|---|---|---|
| 2010 | 3 | 1500 | 0.20 |
| 2015 | 4 | 2000 | 0.35 |
| 2012 | 3 | 1450 | 0.25 |
| 2000 | 3 | 1600 | 0.15 |
| 2018 | 4 | 1900 | 0.40 |
| 2014 | 3 | 1450 | 0.27 |
| 2008 | 5 | 2500 | 0.45 |
| 2011 | 3 | 1600 | 0.26 |

# Regression Trees

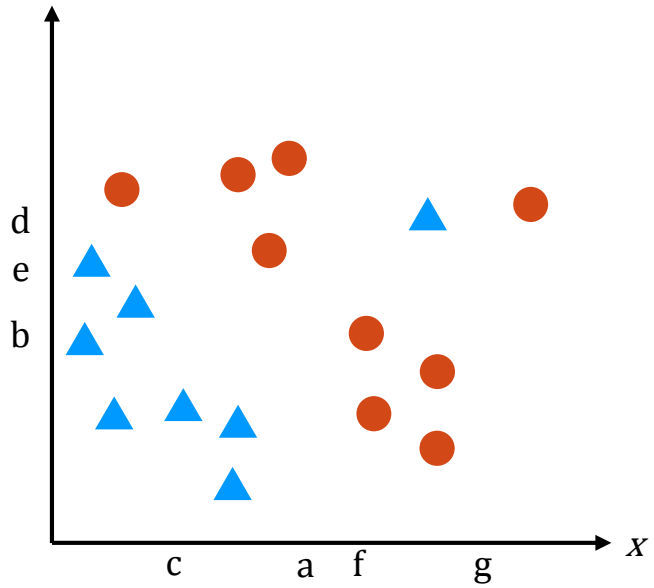
- Year



# Pruning Decision Trees

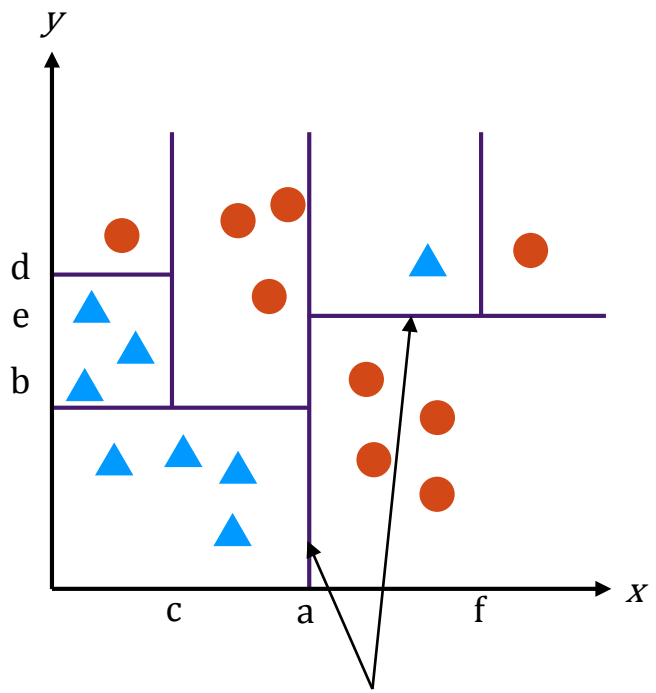
# Pruning

- Consider this dataset

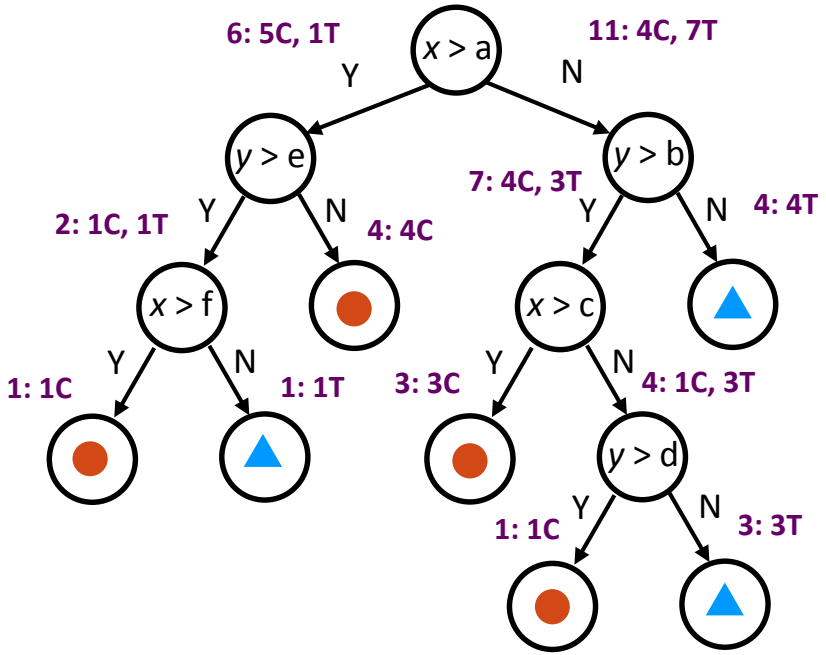


# Pruning

- Consider this dataset



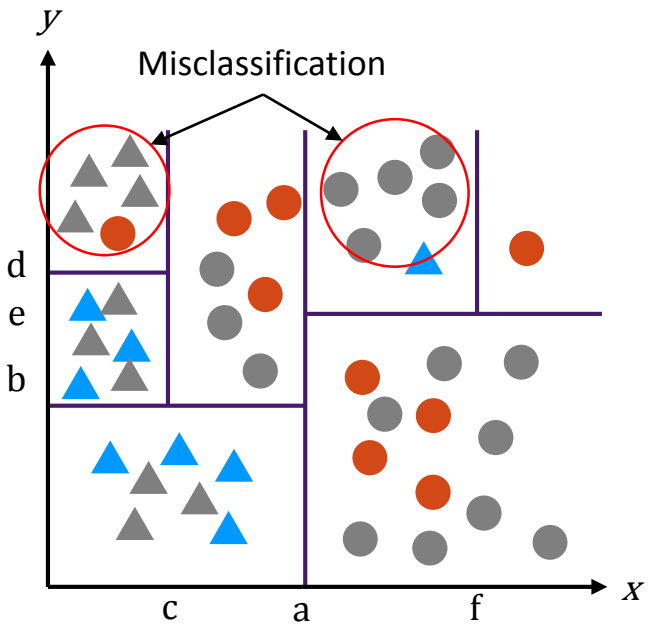

Decision boundaries



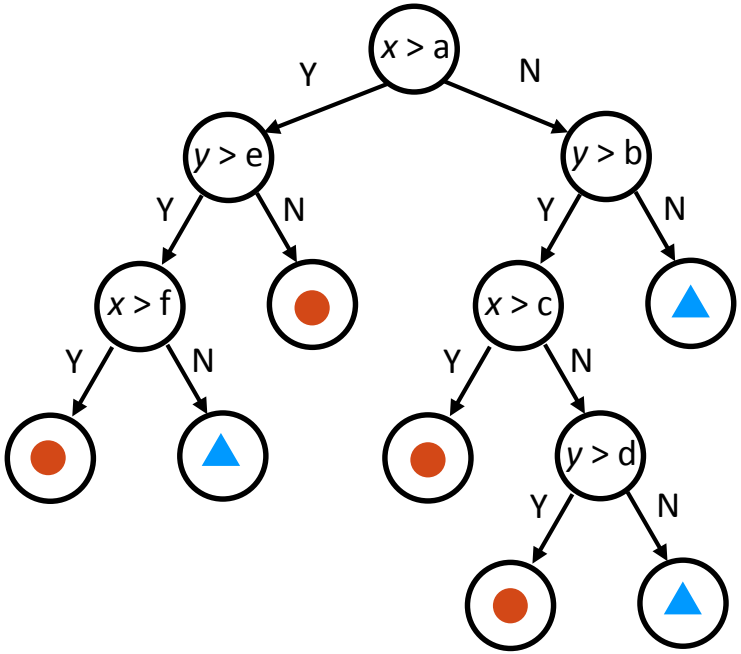

Classification accuracy is 100%

# Pruning

- Consider this dataset



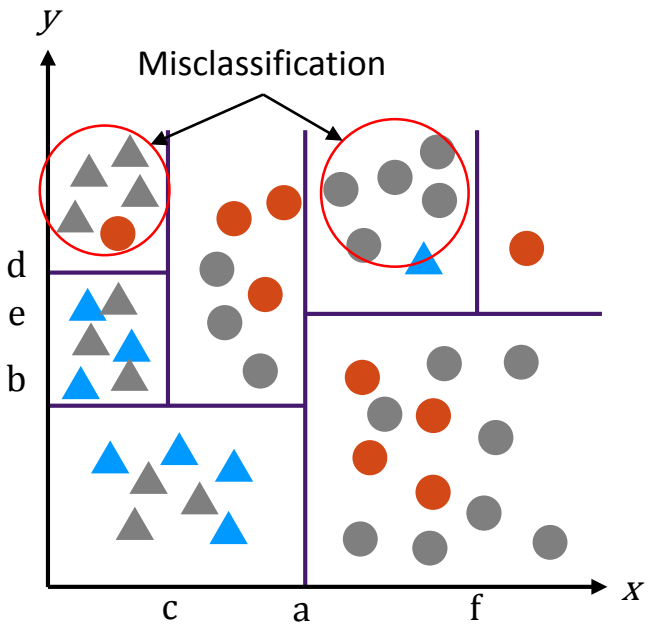

Changes in data affects the accuracy of the prediction – high variance model

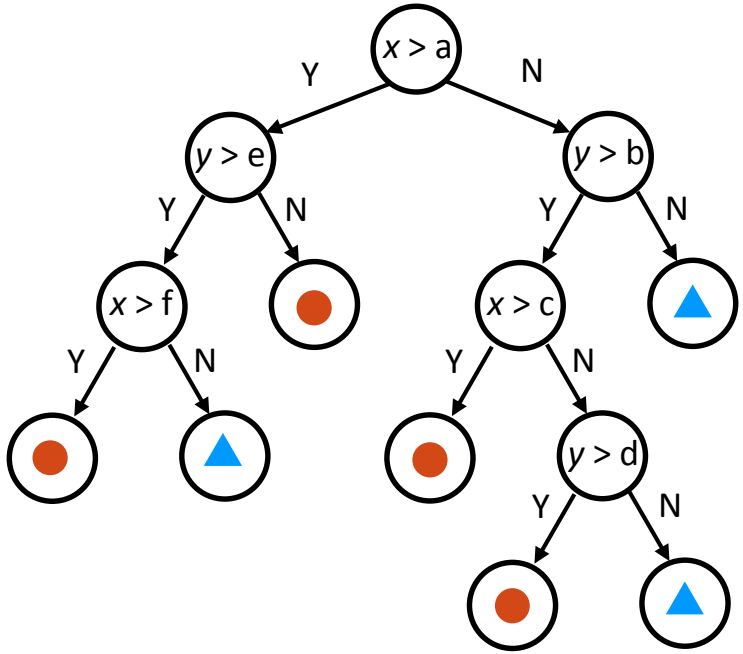



The decision tree model does not generalized well outside training data

# Pruning

- Consider this dataset



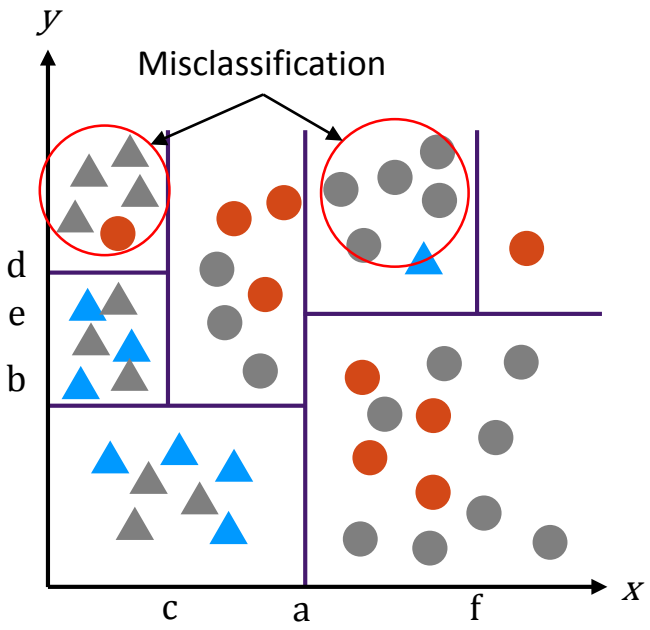



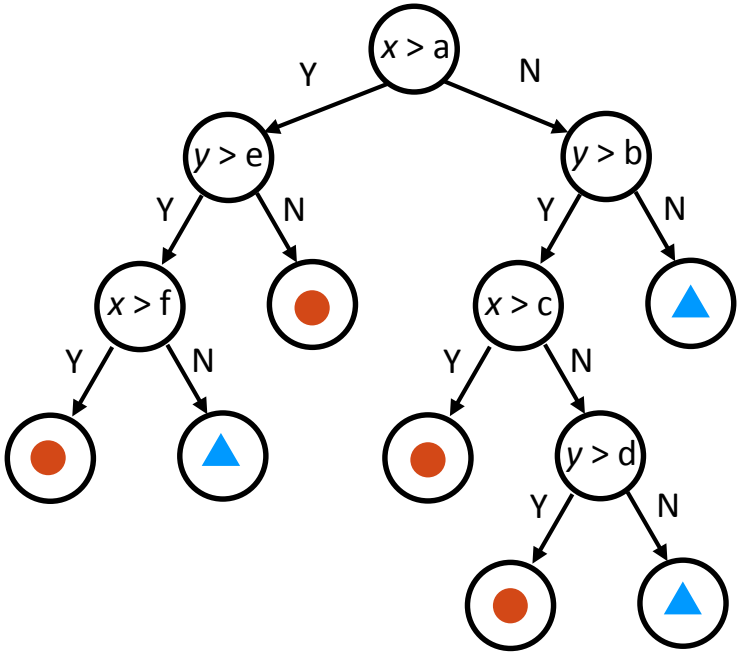

The model is overfitting the training data

Classification accuracy reduces to 61%

# Pruning

- Consider this dataset



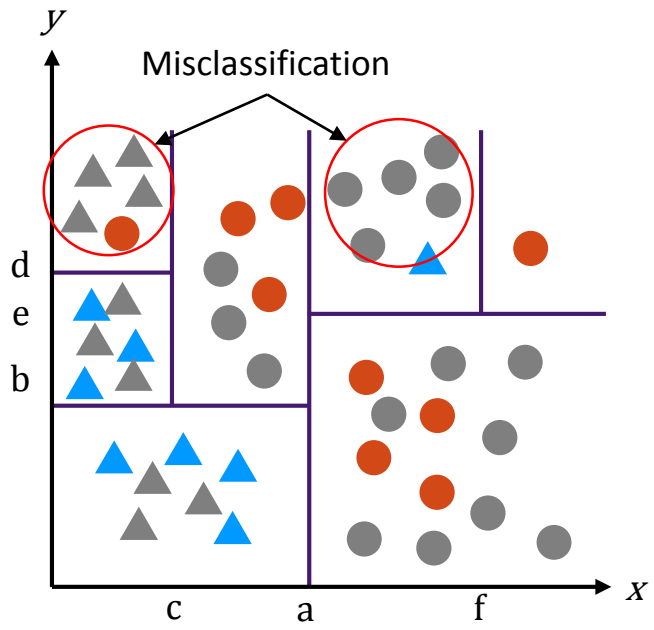



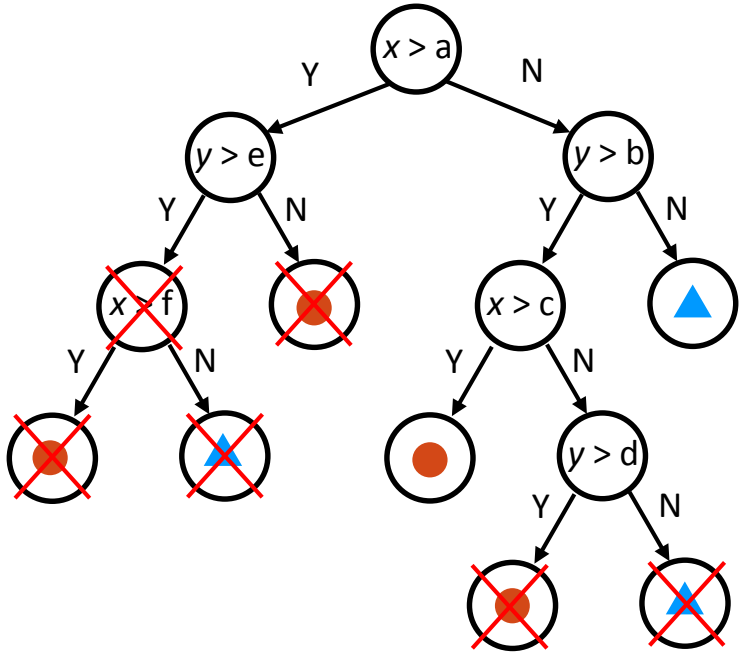

The model is overfitting the training data
How do we prevent overfitting?

Classification accuracy reduces to 61%

# Pruning

- Consider this dataset

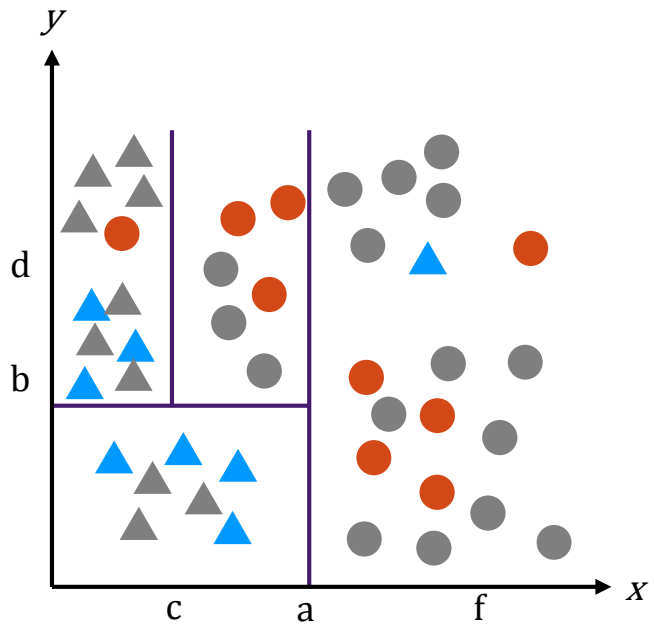


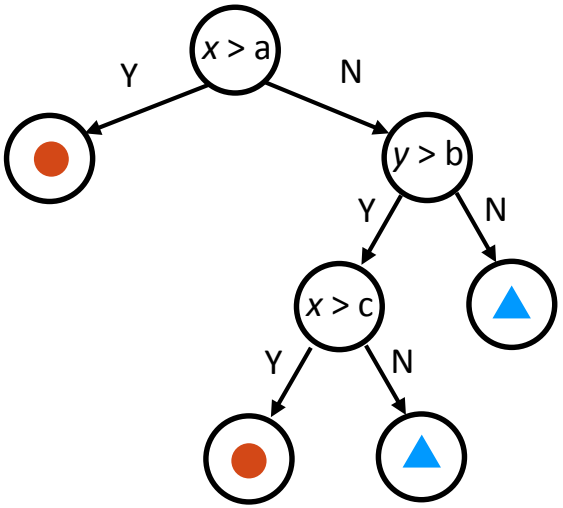




Remove the leaves

# Pruning

- Consider this dataset




Replace the split (decision nodes) with leaves
Classification accuracy is 95%

# Pruning

- Pre-pruning
- Post-pruning

# Pre-pruning

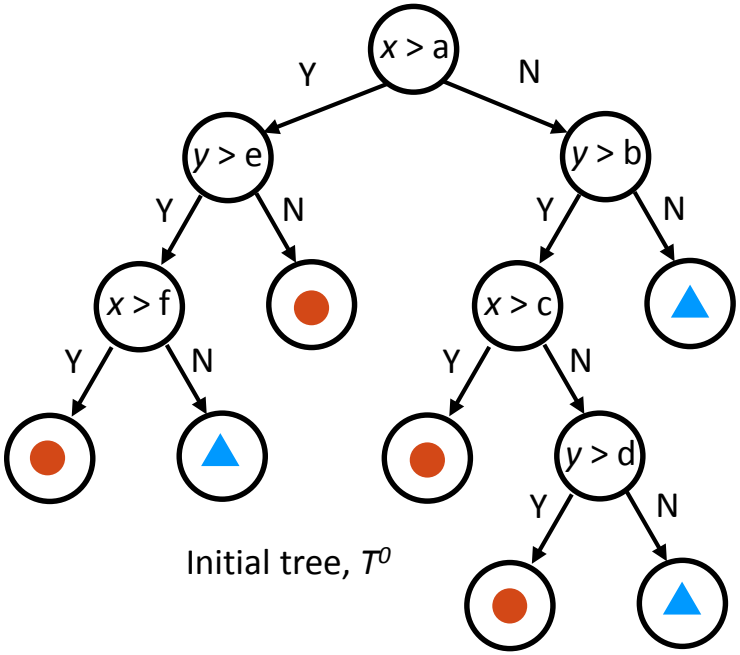
- Early stopping criteria
- Stop the tree-building process early before it produces leaves with very small samples

# Pre-pruning

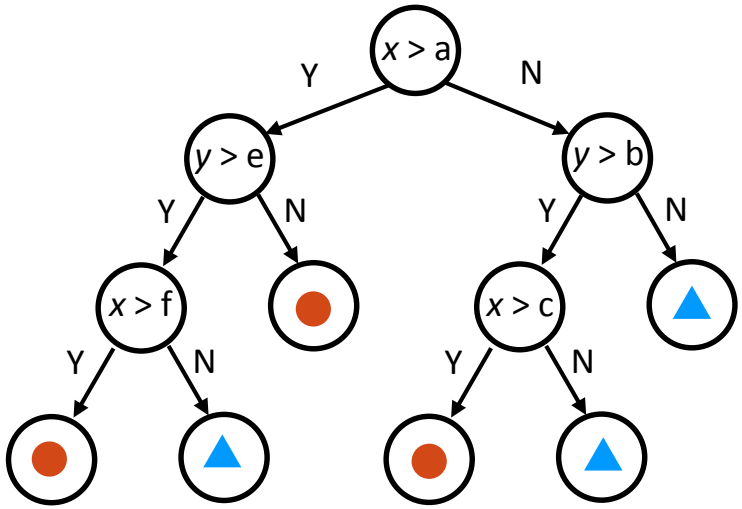
- Early stopping criteria
- Stop the tree-building process early before it produces leaves with very small samples
- Max depth: length of the longest path from root node to a leaf node
- Min split: minimum number of samples that must exist in a node in order for a split to be attempted
  - Min split is 5, a node can be further split if it has more than 5 samples
- Min bucket: minimum number of samples in any terminal
  - Min bucket is 5, every node should have at least five samples

# Post-pruning

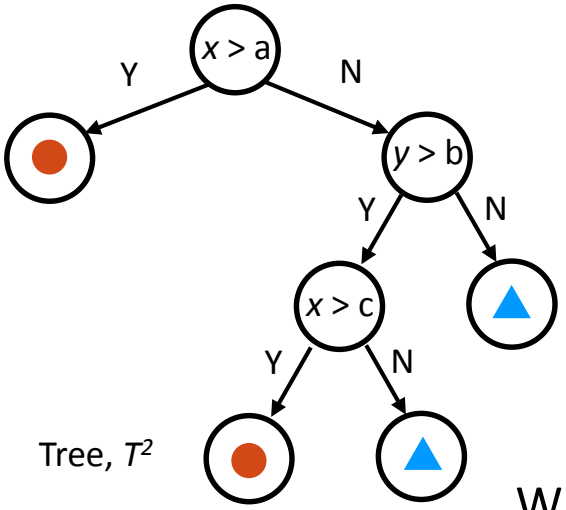
- Allow the tree to grow until all leaves are pure and/or have no training error (it may be overfitted in the absence of early stopping)
- Prune (cut) the tree that has been built to obtain the optimal tree



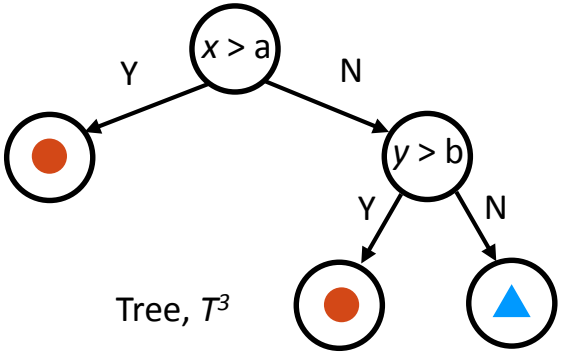

Initial tree, $T^0$



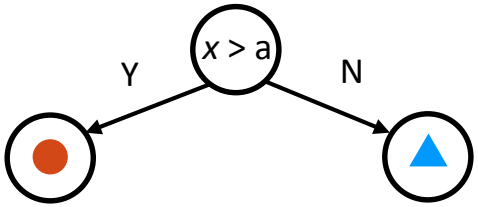

Tree, $T^1$




Tree, $T^2$




Tree, $T^3$




Tree, $T^4$

Which tree is the optimal tree?

# Cost Complexity Pruning

- Tree pruning is based on the cost complexity function

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

- $R(T)$ is the training /learning error (error that is calculated across the leaf nodes of $T$)
- $|T|$ is the number of leaf nodes in $T$

# Cost Complexity Pruning

- Tree pruning is based on the cost complexity function

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

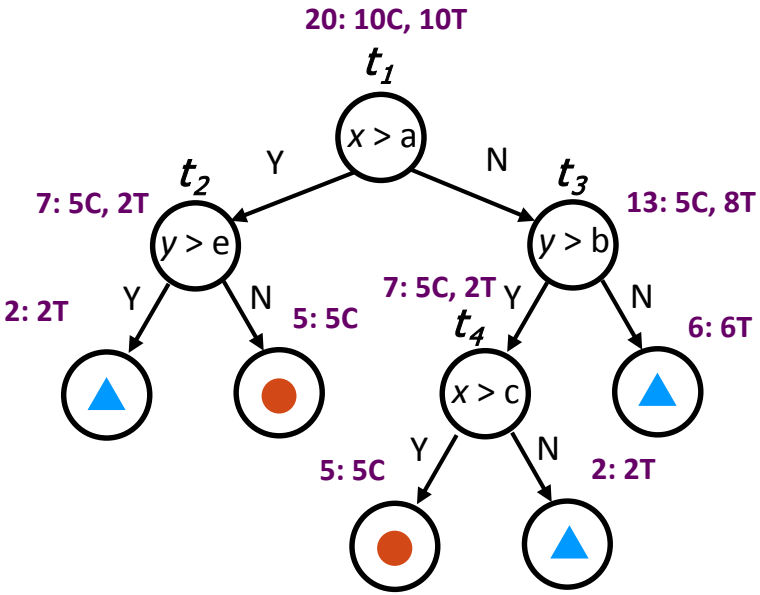
- $\alpha$ is a tuning parameter governs the tradeoff between the tree size and its goodness of fit to the data
- Large value of $\alpha$ results in smaller trees and small value of $\alpha$ will result in bigger trees
- $\alpha = 0 \rightarrow$ the tree is the full tree $T^0$
- $\alpha \approx \propto \rightarrow$ the tree is a tree with the root node only

# Cost Complexity Pruning

- We start from the full tree, $T^0$
- For any decision node $t$, let $T_t$ is the subtree of $T$ with root $t$
- We prune $T_t$ and calculate the $\alpha = g(t) = \frac{R(t)-R(T_t)}{|T_t|-1}$ where
- $R(t) = r(t) \cdot p(t)$
  - $r(t)$: error at decision node $t$
  - $p(t)$: proportion of data items at decision node $t$
- $R(T_t) = \sum_{l \in T_t} R(l)$
  - $l$ is a leaf of subtree $T_t$
- $|T_t|$ is the number of leaves to prune
- Choose $g_i(t)$ that is minimum
- Repeat the above process until the tree is reduced to the root node

# Cost Complexity Pruning

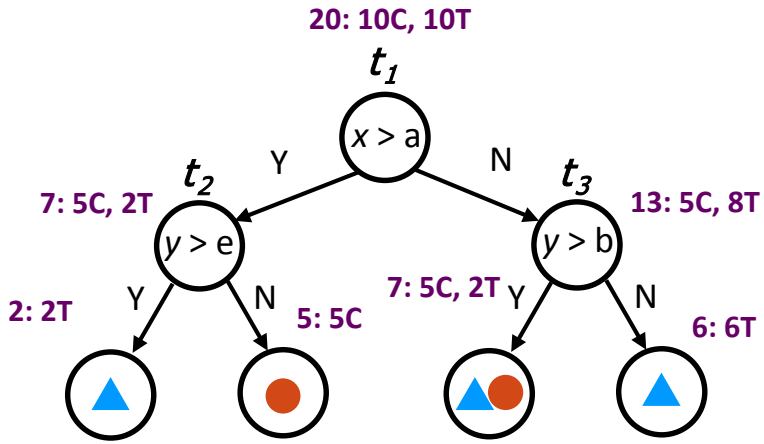
- $\alpha = g_i(t) = \frac{R(t) - R(T_t)}{|T_t| - 1}$ where
  - $R(t) = r(t) \cdot p(t)$
  - $R(T_t) = \sum_{l \in T_t} R(l)$
  - $|T_t|$ is the number of leaves to prune




- For $t_4$
- $R(t_4) = \frac{2}{7} \cdot \frac{7}{20} = \frac{2}{20}$
- $R(T_{t_4}) = 0$ (all leaves are pure)
- $|T_{t_4}| = 2$
- $g(t_4) = \frac{\frac{2}{20} - 0}{2 - 1} = 0.100$

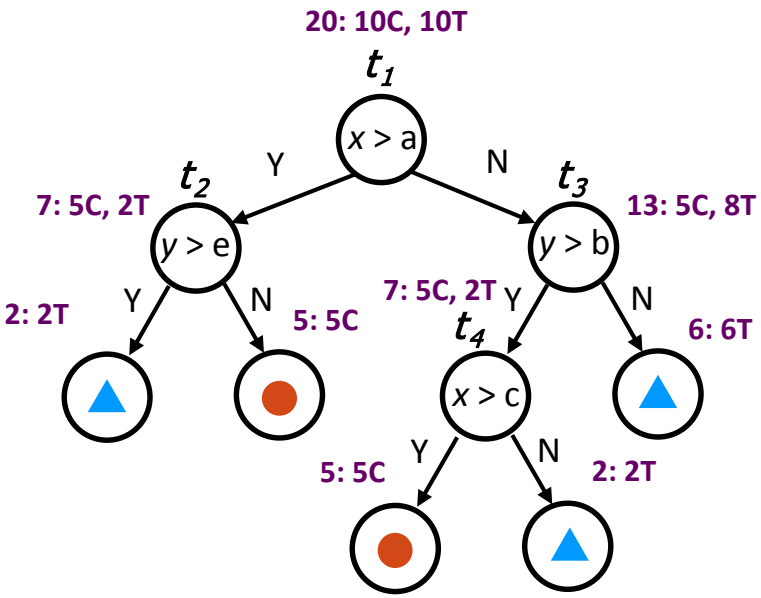
# Cost Complexity Pruning

- $\alpha = g_i(t) = \frac{R(t) - R(T_t)}{|T_t| - 1}$ where
  - $R(t) = r(t) \cdot p(t)$
  - $R(T_t) = \sum_{l \in T_t} R(l)$
  - $|T_t|$ is the number of leaves to prune

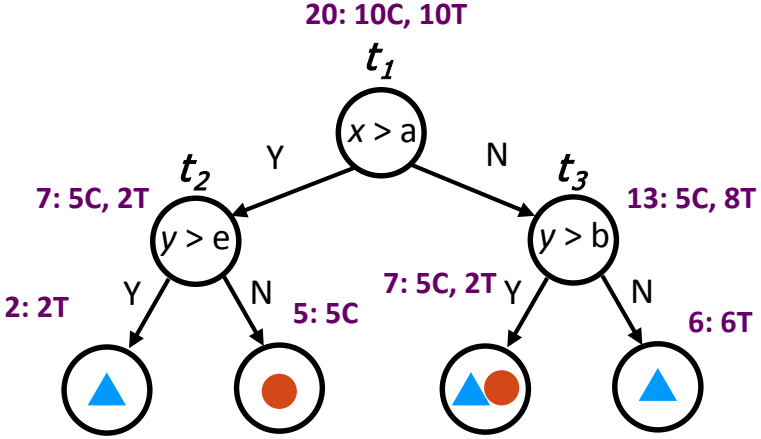



- For $t_3$
- $R(t_3) = \frac{5}{13} \cdot \frac{13}{20} = \frac{5}{20}$
- $R(T_{t_3}) = \frac{2}{20}$
- $|T_{t_3}| = 2$
- $g(t_3) = \frac{\frac{5}{20} - \frac{2}{20}}{2-1} = 0.150$

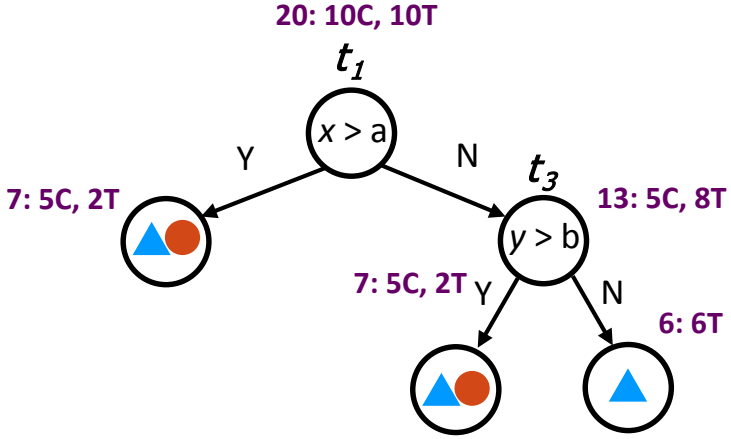
$\alpha^1 = 0$



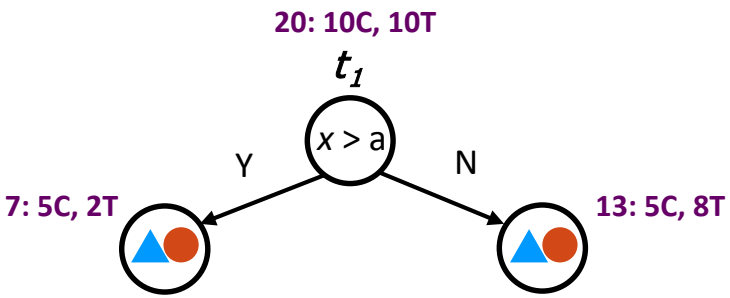

$\alpha^2 = 0.100$




$\alpha^3 = 0.100$




$\alpha^4 = 0.150$



# Cost Complexity Pruning

- Use $K$-fold cross-validation
- For $k = 1{:}\,K$
  - Build the tree $T_{\alpha_i}$ using the training data (fold $k$)
  - Evaluate the tree $T_{\alpha_i}$ - calculate the prediction error
- Select the best tree with the lowest average error

UNIVERSITI SAINS MALAYSIA

# End