AI VIET NAM
@aivietnam.edu.vn

Module 03 – Exercise Class

DECISION TREE
Classification & Regression

Nguyen Quoc Thai

Objectives

Decision Tree

- Introduction
- Regression & Classification Problem
- Terminology

Decision Tree for Regression

- Variance
- Sum of Squared Errors (SSE)
- Sklearn Library

Decision Tree for Classification

- Constructing Decision Tree: ID3
- Gini Impurity
- Entropy
- Information Gain
- Sklearn library

Improved Decision Tree

- Overfitting
- Stopping Early
- Post-Pruning (Reduced-Error & Rule Post-Pruning)
- Missing Attribute Values

AI VIET NAM
@aivietnam.edu.vn

Outline

SECTION 1

Decision Tree

SECTION 2

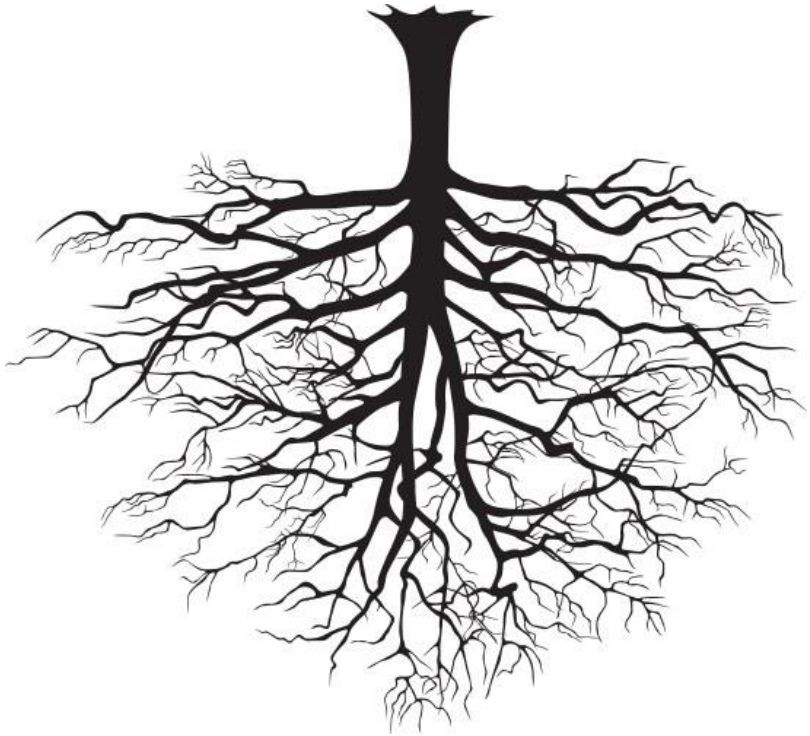
DT for Classification

SECTION 3

DT for Regression

SECTION 4

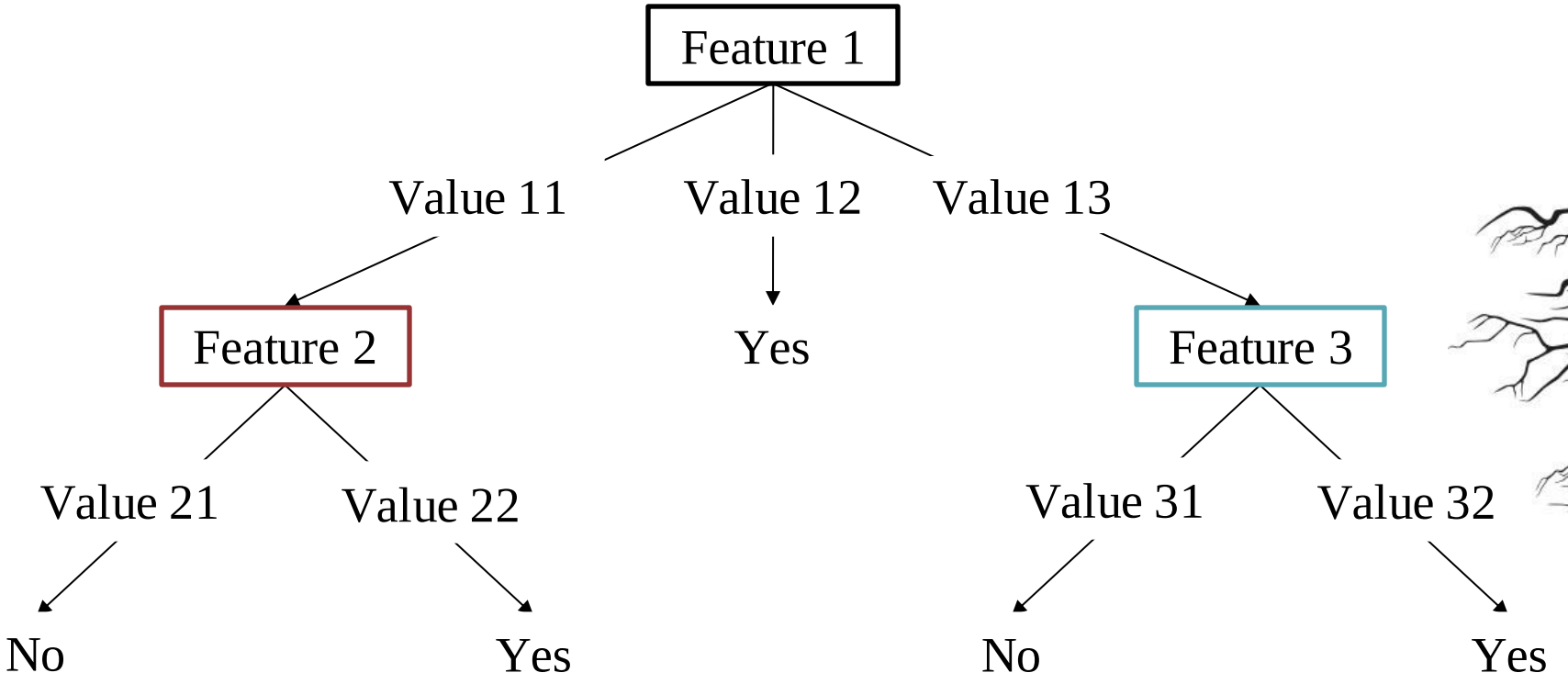
Improved DT

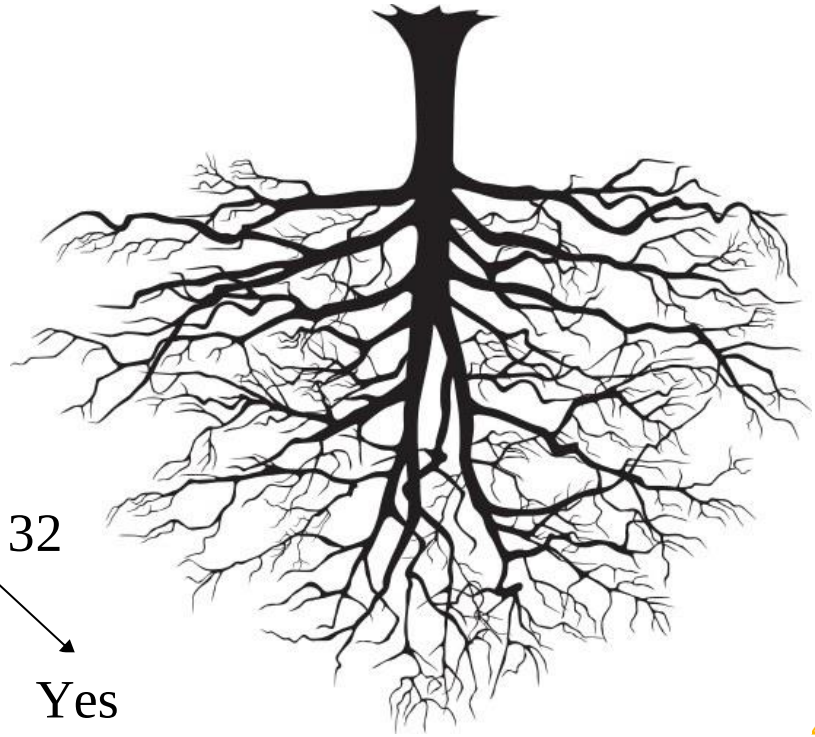


Decision Tree

Introduction

A Decision Tree is a non-parametric Supervised Machine Learning algorithm
Used for both regression and classification problem





Decision Tree

Non-parametric Supervised Machine Learning

Parametric SML

X	Y
2	4
6	12

2	*	**2**	=	4
6				12

Parameter

Non-parametric SML

X	Y
2	0
6	1

2	$X < 4$	0
6	$X \geq 4$	1

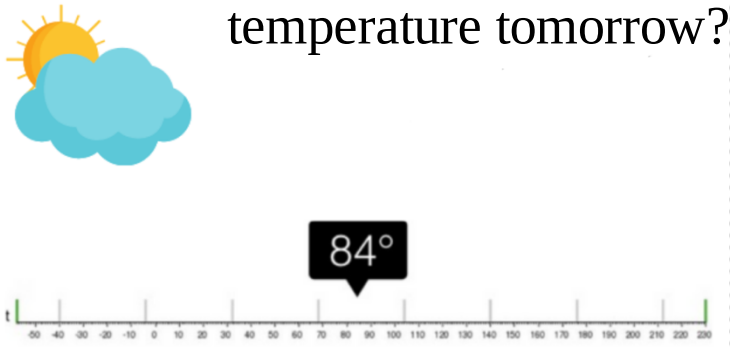
Decision Tree

Regression & Classification Problem

Regression

- Predict a continuous value based on the input variables

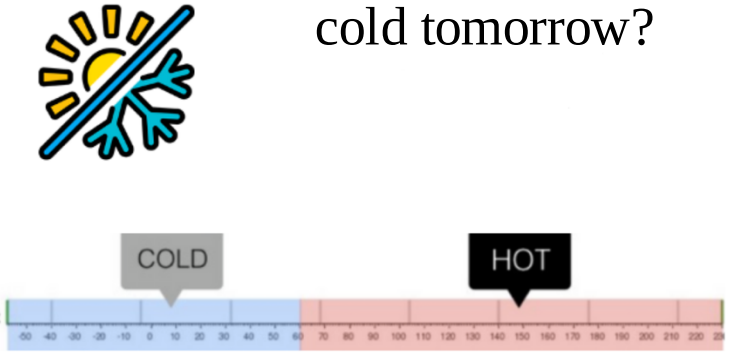
What will be the temperature tomorrow?



Classification

- Classify input variables to identify discrete output variables (labels, categories)

Will it be hot or cold tomorrow?



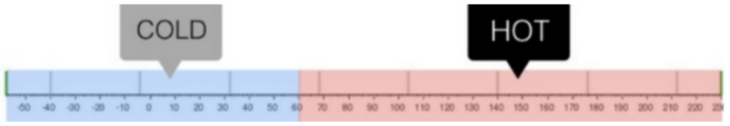
Decision Tree

Regression & Classification Problem

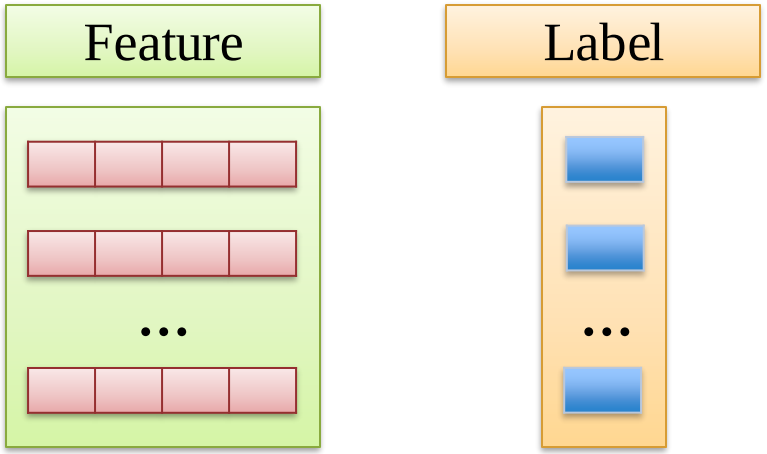
Classification

- Classify input variables to identify discrete output variables (labels, categories)

Will it be hot or cold tomorrow?



Training Data



Test Data

Decision Tree

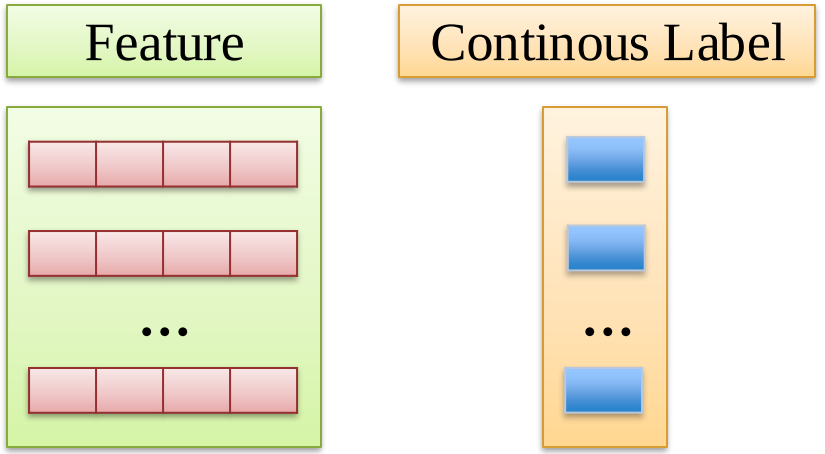
Regression & Classification Problem

Regression

- Predict a continuous value based on the input variables

What will be the temperature tomorrow?

Training Data

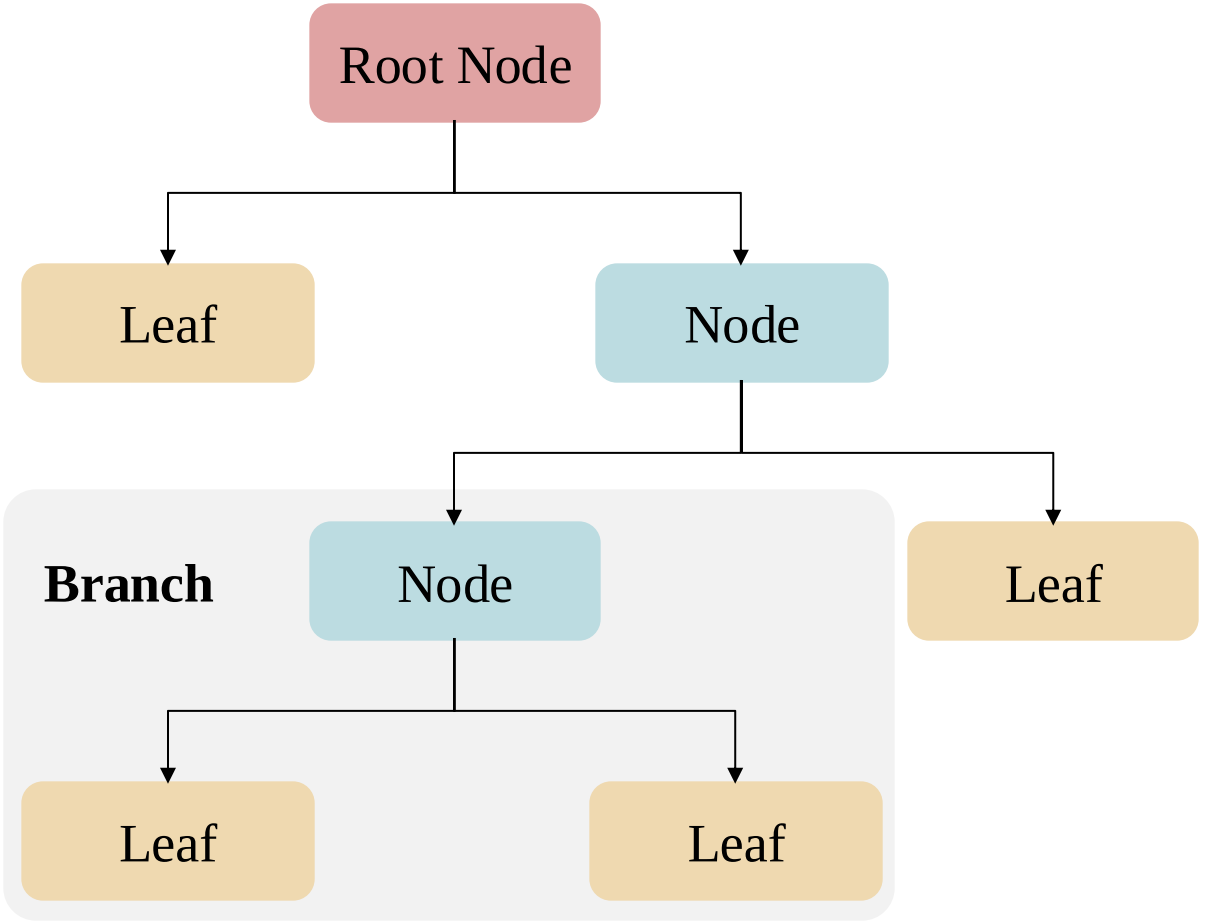


Test Data

Decision Tree

Terminology

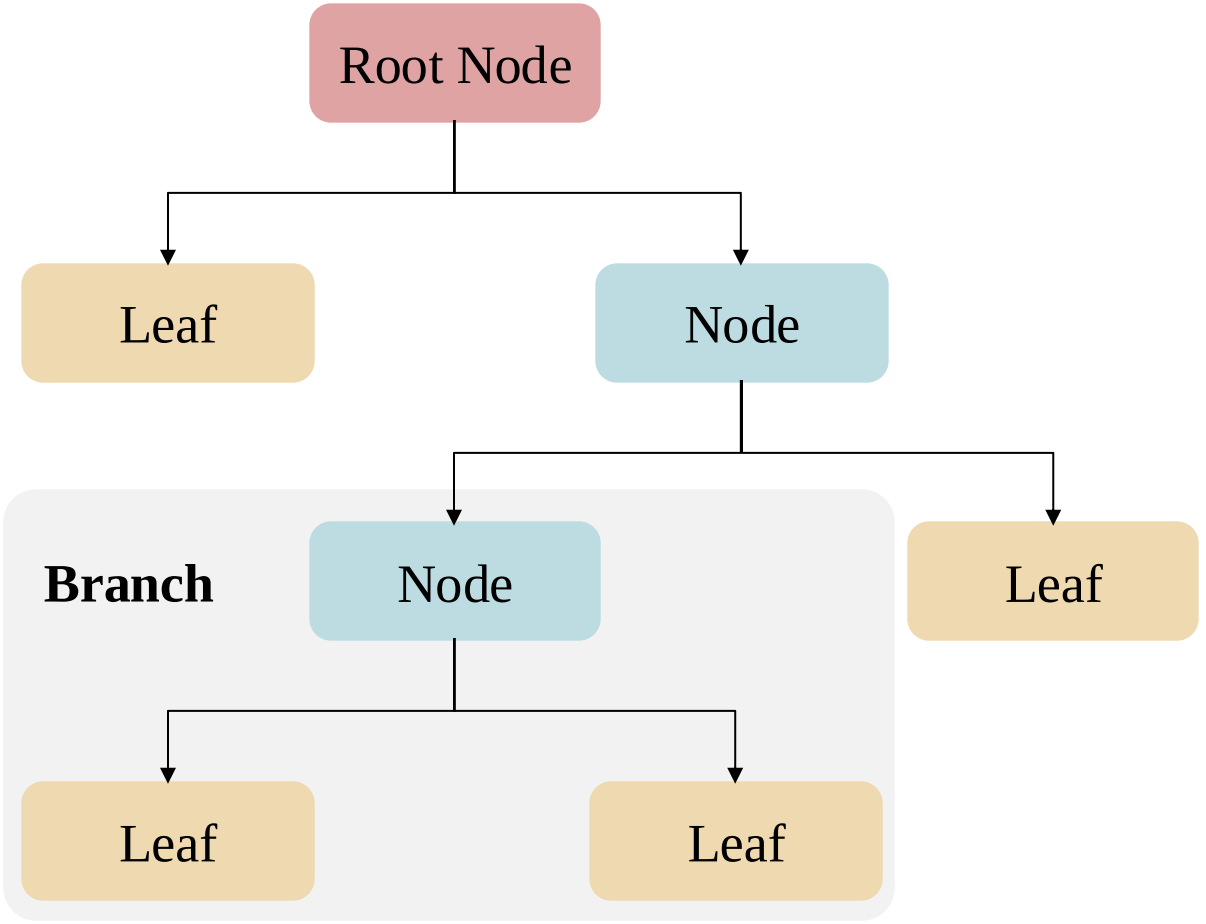
- **Root Node**: the top-level node
- **Node**: internal node or decision node
- **Parent Node**: a node that precedes a (child) node
- **Leaf**: terminal node – a node at the end of a branch – represents outcome of the tree (label or numerical value)
- **Branches**: a subset of a tree, starting at an (internal) node until the leaves



Decision Tree

Terminology

- **Splitting:** divide a node into two child nodes depending on a criterion and a selected feature
- **Pruning**: removing a branch from a tree
- Described by **IF-ELSE** Sets



Outline

SECTION 1

Decision Tree

SECTION 2

DT for Classification

SECTION 3

DT for Regression

SECTION 4

Improved DT

Dataset D (n sample)

Dataset D_1 (n_1 sample)

Dataset D_2 (n_2 sample)

Age	Likes English	Likes AI	Raise Salary
23	0	0	**0**
25	1	1	**0**
27	1	0	**1**
29	0	1	**1**
29	0	0	**0**

DT for Classification

Decision Tree

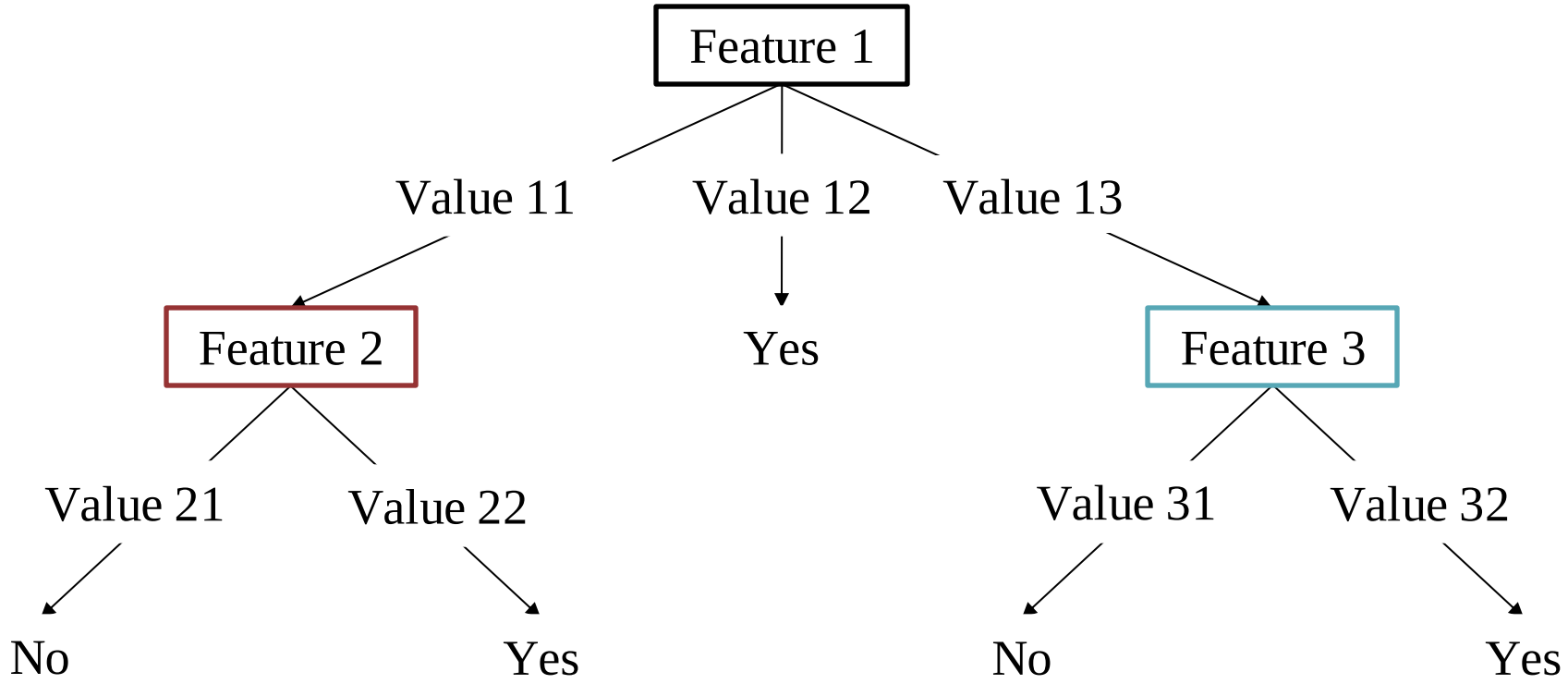
- Example Dataset

Age	Likes English	Likes AI	Raise Salary
23	0	0	**0**
25	1	1	**0**
27	1	0	**1**
29	0	1	**1**
29	0	0	**0**

DT for Classification

Decision Tree

- Main step: splitting the data according to a splitting criterion



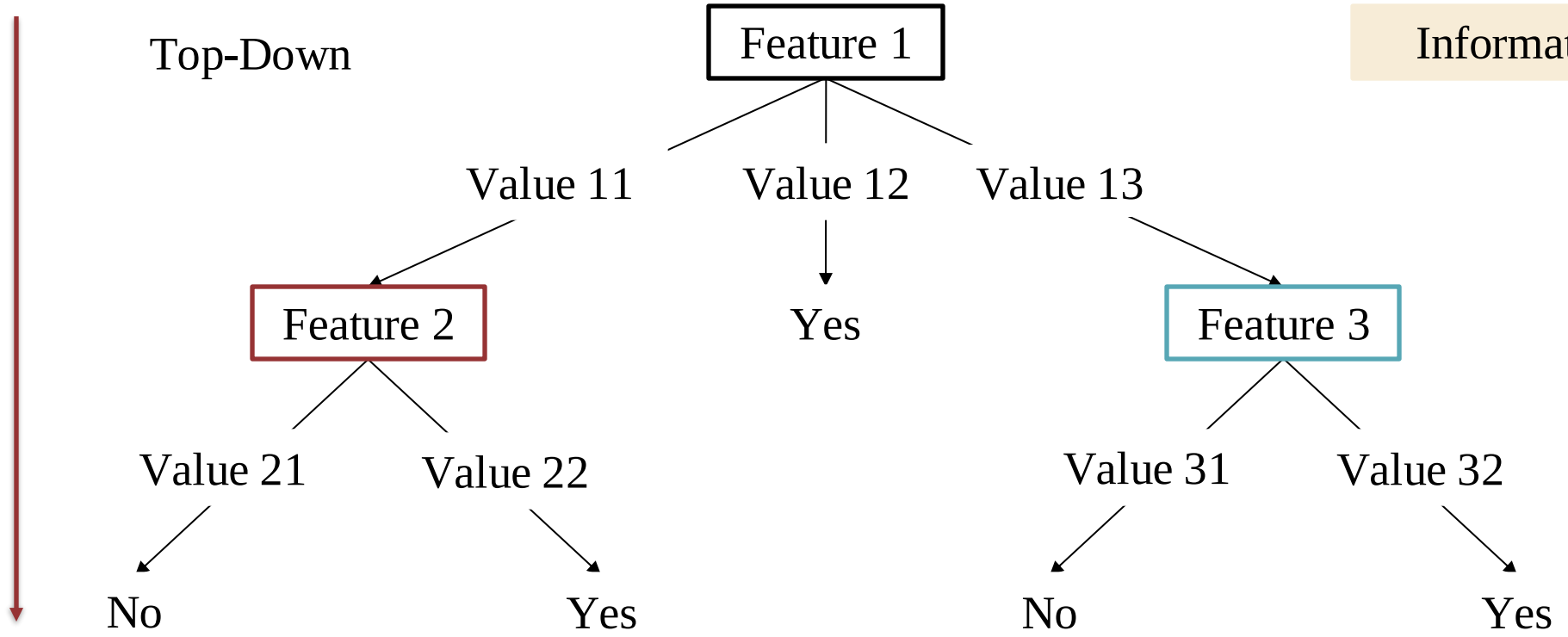
DT for Classification

Constructing Decision Tree: ID3

- Iterative Dichotomiser 3

Gini Impurity

Information Gain



DT for Classification

Gini Impurity

$$Gini(D) = \frac{n_1}{n} Gini(D_1) + \frac{n_2}{n} Gini(D_2)$$

$$Gini(D_i) = 1 - \sum_{j=1}^{c} p_j^2$$

p_j the probability that a random drawn sample from this node belongs to class j and c the number of classes

1. For each possible split, create child nodes and calculate the Gini Impurity of each child node.
2. Calculate the Gini Impurity of the split as the weighted average Gini Impurity of child nodes.
3. Select the split with the lowest Gini Impurity.

Dataset D (n sample)

Dataset D_1 (n_1 sample)

Dataset D_2 (n_2 sample)

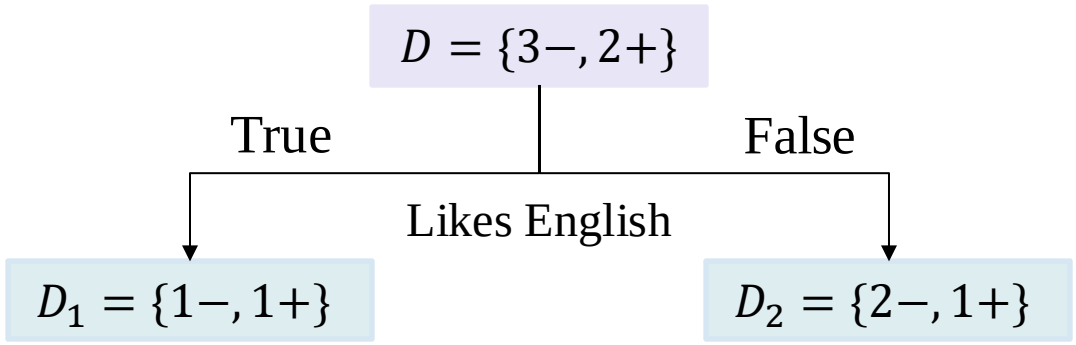
Age	Likes English	Likes AI	Raise Salary
23	0	0	**0**
25	1	1	**0**
27	1	0	**1**
29	0	1	**1**
29	0	0	**0**

DT for Classification

Gini Impurity

$$Gini(D) = \frac{n_1}{n} Gini(D_1) + \frac{n_2}{n} Gini(D_2)$$

$$Gini(D_i) = 1 - \sum_{j=1}^{c} p_j^2$$



$$Gini(D_1) = 1 - \left(\frac{1}{1+1}\right)^2 - \left(\frac{1}{1+1}\right)^2 = \frac{4}{9}$$

$$Gini(D_2) = 1 - \left(\frac{2}{2+1}\right)^2 - \left(\frac{1}{2+1}\right)^2 = \frac{4}{9}$$

$$Gini(Likes\ English) = \frac{2}{5} * \frac{1}{2} + \frac{3}{5} * \frac{4}{9} \approx 0.47$$

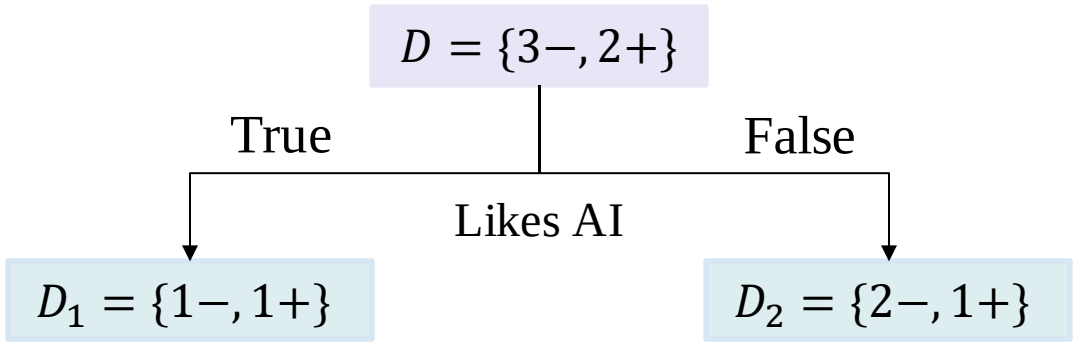
Age	Likes English	Likes AI	Raise Salary
23	0	0	**0**
25	1	1	**0**
27	1	0	**1**
29	0	1	**1**
29	0	0	**0**

DT for Classification

Gini Impurity

$$Gini(D) = \frac{n_1}{n} Gini(D_1) + \frac{n_2}{n} Gini(D_2)$$

$$Gini(D_i) = 1 - \sum_{j=1}^{c} p_j^2$$



$$Gini(D_1) = 1 - \left(\frac{1}{1+1}\right)^2 - \left(\frac{1}{1+1}\right)^2 = \frac{1}{2}$$

$$Gini(D_2) = 1 - \left(\frac{2}{2+1}\right)^2 - \left(\frac{1}{2+1}\right)^2 = \frac{4}{9}$$

$$Gini(Likes\ AI) = \frac{2}{5} * \frac{1}{2} + \frac{3}{5} * \frac{4}{9} \approx 0.47$$

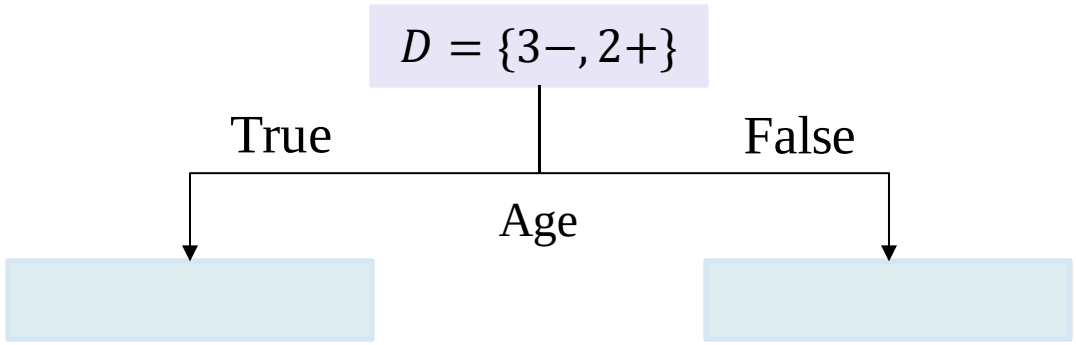
Age	Likes English	Likes AI	Raise Salary
23	0	0	**0**
25	1	1	**0**
27	1	0	**1**
29	0	1	**1**
29	0	0	**0**

DT for Classification

Gini Impurity

$$Gini(D) = \frac{n_1}{n} Gini(D_1) + \frac{n_2}{n} Gini(D_2)$$

$$Gini(D_i) = 1 - \sum_{j=1}^{c} p_j^2$$



Age	Likes English	Likes AI	Raise Salary
23	0	0	**0**
25	1	1	**0**
27	1	0	**1**
29	0	1	**1**
29	0	0	**0**

DT for Classification

Gini Impurity

$$Gini(D) = \frac{n_1}{n} Gini(D_1) + \frac{n_2}{n} Gini(D_2)$$

$$Gini(D_i) = 1 - \sum_{j=1}^{c} p_j^2$$

Numerical Feature	Ascending Ordering	Calculate mean	Determine Gini

$D = \{3-, 2+\}$

True | False

Age $\leq$ 24

$D_1 = \{1-, 0+\}$ $\quad$ $D_2 = \{2-, 2+\}$

$$Gini(D_1) = 1 - \left(\frac{1}{1+0}\right)^2 - \left(\frac{0}{1+0}\right)^2 = 0$$

$$Gini(D_2) = 1 - \left(\frac{2}{2+2}\right)^2 - \left(\frac{2}{2+2}\right)^2 = \frac{1}{2}$$

$$Gini(\text{Age} \leq 24) = \frac{1}{5} * 0 + \frac{4}{5} * \frac{1}{2} = 0.40$$

	Age	Likes English	Likes AI	Raise Salary
24	23	0	0	**0**
26	25	1	1	**0**
28	27	1	0	**1**
29	29	0	1	**1**
	29	0	0	**0**

DT for Classification

Gini Impurity

$$Gini(D) = \frac{n_1}{n} Gini(D_1) + \frac{n_2}{n} Gini(D_2)$$

$$Gini(D_i) = 1 - \sum_{j=1}^{c} p_j^2$$

Numerical Feature	Ascending Ordering	Calculate mean	Determine Gini

$D = \{3-, 2+\}$

True | False

Age $\leq$ 26

$D_1 = \{2-, 0+\}$ $\quad$ $D_2 = \{1-, 2+\}$

$$Gini(D_1) = 1 - \left(\frac{2}{2+0}\right)^2 - \left(\frac{0}{2+0}\right)^2 = 0$$

$$Gini(D_2) = 1 - \left(\frac{1}{1+2}\right)^2 - \left(\frac{2}{1+2}\right)^2 = \frac{4}{9}$$

$$Gini(\text{Age} \leq 26) = \frac{2}{5} * 0 + \frac{3}{5} * \frac{4}{9} \approx 0.27$$

	Age	Likes English	Likes AI	Raise Salary
24	23	0	0	**0**
26	25	1	1	**0**
28	27	1	0	**1**
29	29	0	1	**1**
	29	0	0	**0**

DT for Classification

Gini Impurity

$$Gini(D) = \frac{n_1}{n} Gini(D_1) + \frac{n_2}{n} Gini(D_2)$$

$$Gini(D_i) = 1 - \sum_{j=1}^{c} p_j^2$$

Numerical Feature	Ascending Ordering	Calculate mean	Determine Gini

$D = \{3-, 2+\}$

True | False

Age $\leq$ 28

$D_1 = \{2-, 1+\}$ $\quad$ $D_2 = \{1-, 1+\}$

$$Gini(D_1) = 1 - \left(\frac{2}{2+1}\right)^2 - \left(\frac{1}{2+1}\right)^2 = \frac{4}{9}$$

$$Gini(D_2) = 1 - \left(\frac{1}{1+1}\right)^2 - \left(\frac{1}{1+1}\right)^2 = \frac{1}{2}$$

$$Gini(\text{Age} \leq 28) = \frac{3}{5} * \frac{4}{9} + \frac{2}{5} * \frac{1}{2} \approx 0.47$$

	Age	Likes English	Likes AI	Raise Salary
24	23	0	0	**0**
26	25	1	1	**0**
28	27	1	0	**1**
29	29	0	1	**1**
	29	0	0	**0**

DT for Classification

Gini Impurity

$$Gini(D) = \frac{n_1}{n} Gini(D_1) + \frac{n_2}{n} Gini(D_2)$$

$$Gini(D_i) = 1 - \sum_{j=1}^{c} p_j^2$$

Numerical Feature	Ascending Ordering	Calculate mean	Determine Gini

$D = \{3-, 2+\}$

True | False

Age $\leq$ 29

$D_1 = \{3-, 2+\}$ $\quad$ $D_2 = \{0-, 0+\}$

$$Gini(D_1) = 1 - \left(\frac{3}{3+2}\right)^2 - \left(\frac{2}{3+2}\right)^2 = \frac{12}{25}$$

$$Gini(D_2) = 1 - \left(\frac{0}{0+0}\right)^2 - \left(\frac{0}{0+0}\right)^2 = 1$$

$$Gini(\text{Age} \leq 29) = \frac{5}{5} * \frac{12}{25} + \frac{0}{5} * 1 = 0.48$$

	Age	Likes English	Likes AI	Raise Salary
24	23	0	0	**0**
26	25	1	1	**0**
28	27	1	0	**1**
29	29	0	1	**1**
	29	0	0	**0**

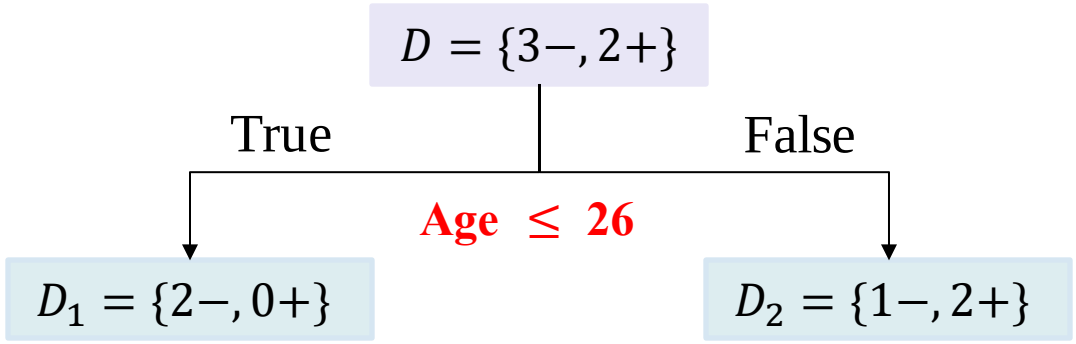
DT for Classification

Gini Impurity

$$Gini(D) = \frac{n_1}{n} Gini(D_1) + \frac{n_2}{n} Gini(D_2)$$

$$Gini(D_i) = 1 - \sum_{j=1}^{c} p_j^2$$

Numerical Feature	Ascending Ordering	Calculate mean	Determine Gini



$Gini(Likes\ English) \approx 0.47$

$Gini(Likes\ AI) \approx 0.47$

$Gini(\text{Age} \leq 24) = 0.40$

$\boldsymbol{Gini(\text{Age} \leq 26) = 0.27}$

$Gini(\text{Age} \leq 28) = 0.47$

$Gini(\text{Age} \leq 29) = 0.48$

	Age	Likes English	Likes AI	Raise Salary
	23	0	0	**0**
24	25	1	1	**0**
26	27	1	0	**1**
28	29	0	1	**1**
29	29	0	0	**0**

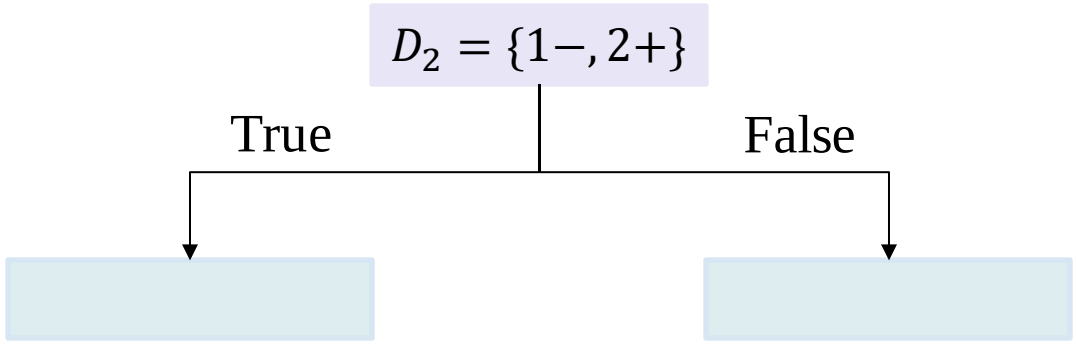
DT for Classification

Gini Impurity

$$Gini(D) = \frac{n_1}{n} Gini(D_1) + \frac{n_2}{n} Gini(D_2)$$

$$Gini(D_i) = 1 - \sum_{j=1}^{c} p_j^2$$

Numerical Feature	Ascending Ordering	Calculate mean	Determine Gini



$Gini(Likes\ English)$ $Gini(\text{Age} \leq 28)$

$Gini(Likes\ AI)$ $Gini(\text{Age} \leq 29)$

	Age	Likes English	Likes AI	Raise Salary
	23	0	0	**0**
24	25	1	1	**0**
26	27	1	0	**1**
28	29	0	1	**1**
29	29	0	0	**0**

DT for Classification

Gini Impurity

$$Gini(D) = \frac{n_1}{n} Gini(D_1) + \frac{n_2}{n} Gini(D_2)$$

$$Gini(D_i) = 1 - \sum_{j=1}^{c} p_j^2$$

Numerical Feature	Ascending Ordering	Calculate mean	Determine Gini

$D_2 = \{1-, 2+\}$

True — Likes English — False

$D_{21} = \{0-, 1+\}$

$$Gini(D_{21}) = 1 - \left(\frac{0}{0+1}\right)^2 - \left(\frac{1}{0+1}\right)^2 = 0$$

$D_{22} = \{1-, 1+\}$

$$Gini(D_{22}) = 1 - \left(\frac{1}{1+1}\right)^2 - \left(\frac{1}{1+1}\right)^2 = \frac{1}{2}$$

$$Gini(Likes\ English) = \frac{1}{3} * 0 + \frac{2}{3} * \frac{1}{2} \approx 0.33$$

	Age	Likes English	Likes AI	Raise Salary
24	23	0	0	**0**
26	25	1	1	**0**
28	27	1	0	**1**
29	29	0	1	**1**
	29	0	0	**0**

DT for Classification

Gini Impurity

$$Gini(D) = \frac{n_1}{n} Gini(D_1) + \frac{n_2}{n} Gini(D_2)$$

$$Gini(D_i) = 1 - \sum_{j=1}^{c} p_j^2$$

Numerical Feature	Ascending Ordering	Calculate mean	Determine Gini

$D_2 = \{1-, 2+\}$

True — Likes AI — False

$D_{21} = \{0-, 1+\}$

$$Gini(D_{21}) = 1 - \left(\frac{0}{0+1}\right)^2 - \left(\frac{1}{0+1}\right)^2 = 0$$

$D_{22} = \{1-, 1+\}$

$$Gini(D_{22}) = 1 - \left(\frac{1}{1+1}\right)^2 - \left(\frac{1}{1+1}\right)^2 = \frac{1}{2}$$

$$Gini(Likes\ AI) = \frac{1}{3} * 0 + \frac{2}{3} * \frac{1}{2} \approx 0.33$$

	Age	Likes English	Likes AI	Raise Salary
24	23	0	0	**0**
26	25	1	1	**0**
28	27	1	0	**1**
29	29	0	1	**1**
	29	0	0	**0**

DT for Classification

Gini Impurity

$$Gini(D) = \frac{n_1}{n} Gini(D_1) + \frac{n_2}{n} Gini(D_2)$$

$$Gini(D_i) = 1 - \sum_{j=1}^{c} p_j^2$$

Numerical Feature	Ascending Ordering	Calculate mean	Determine Gini

$D_2 = \{1-, 2+\}$

True | False

Age $\leq$ 28

$D_{21} = \{0-, 1+\}$ $D_{22} = \{1-, 1+\}$

$$Gini(D_{21}) = 1 - \left(\frac{0}{0+1}\right)^2 - \left(\frac{1}{0+1}\right)^2 = 0$$

$$Gini(D_{22}) = 1 - \left(\frac{1}{1+1}\right)^2 - \left(\frac{1}{1+1}\right)^2 = \frac{1}{2}$$

$$Gini(\text{Age} \leq 28) = \frac{1}{3} * 0 + \frac{2}{3} * \frac{1}{2} \approx 0.33$$

	Age	Likes English	Likes AI	Raise Salary
24	23	0	0	**0**
26	25	1	1	**0**
28	27	1	0	**1**
29	29	0	1	**1**
	29	0	0	**0**

DT for Classification

Gini Impurity

$$Gini(D) = \frac{n_1}{n} Gini(D_1) + \frac{n_2}{n} Gini(D_2)$$

$$Gini(D_i) = 1 - \sum_{j=1}^{c} p_j^2$$

Numerical Feature	Ascending Ordering	Calculate mean	Determine Gini

$D_2 = \{1-, 2+\}$

True | False

Age $\leq$ 29

$D_{21} = \{1-, 2+\}$ $D_{22} = \{0-, 0+\}$

$$Gini(D_{21}) = 1 - \left(\frac{1}{1+2}\right)^2 - \left(\frac{2}{1+2}\right)^2 = \frac{4}{9}$$

$$Gini(D_{22}) = 1 - \left(\frac{0}{0+0}\right)^2 - \left(\frac{0}{0+0}\right)^2 = 1$$

$$Gini(\text{Age} \leq 29) = \frac{3}{3} * \frac{4}{9} + \frac{0}{3} * 1 \approx 0.44$$

	Age	Likes English	Likes AI	Raise Salary
24	23	0	0	**0**
26	25	1	1	**0**
28	27	1	0	**1**
29	29	0	1	**1**
	29	0	0	**0**

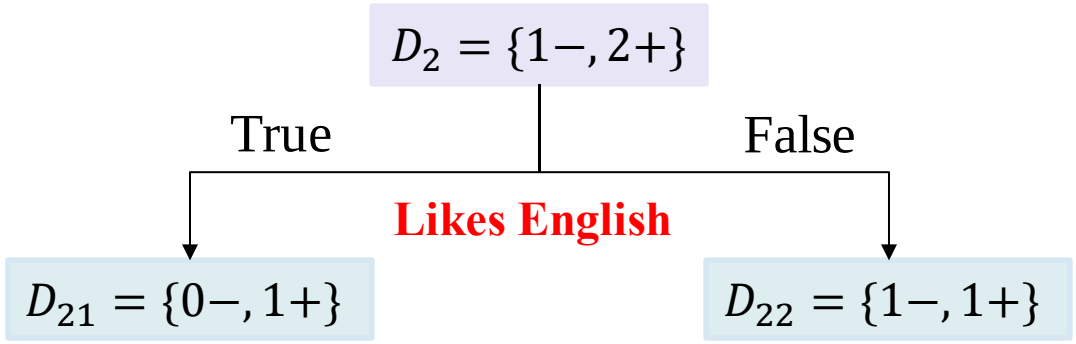
DT for Classification

Gini Impurity

$$Gini(D) = \frac{n_1}{n} Gini(D_1) + \frac{n_2}{n} Gini(D_2)$$

$$Gini(D_i) = 1 - \sum_{j=1}^{c} p_j^2$$

Numerical Feature	Ascending Ordering	Calculate mean	Determine Gini



$Gini(Likes\ English) \approx 0.33$

$Gini(Likes\ AI) \approx 0.33$

$Gini(\text{Age} \leq 28) \approx 0.33$

$Gini(\text{Age} \leq 29) \approx 0.44$

	Age	Likes English	Likes AI	Raise Salary
	23	0	0	**0**
24	25	1	1	**0**
26	27	1	0	**1**
28	29	0	1	**1**
29	29	0	0	**0**

DT for Classification

Gini Impurity

$$Gini(D) = \frac{n_1}{n} Gini(D_1) + \frac{n_2}{n} Gini(D_2)$$

$$Gini(D_i) = 1 - \sum_{j=1}^{c} p_j^2$$

Numerical Feature	Ascending Ordering	Calculate mean	Determine Gini

$D_{22} = \{1-, 1+\}$

True — Likes AI — False

$D_{221} = \{0-, 1+\}$ $D_{222} = \{1-, 0+\}$

$Gini(Likes\ AI) = 0.0$

Age $\leq$ 29

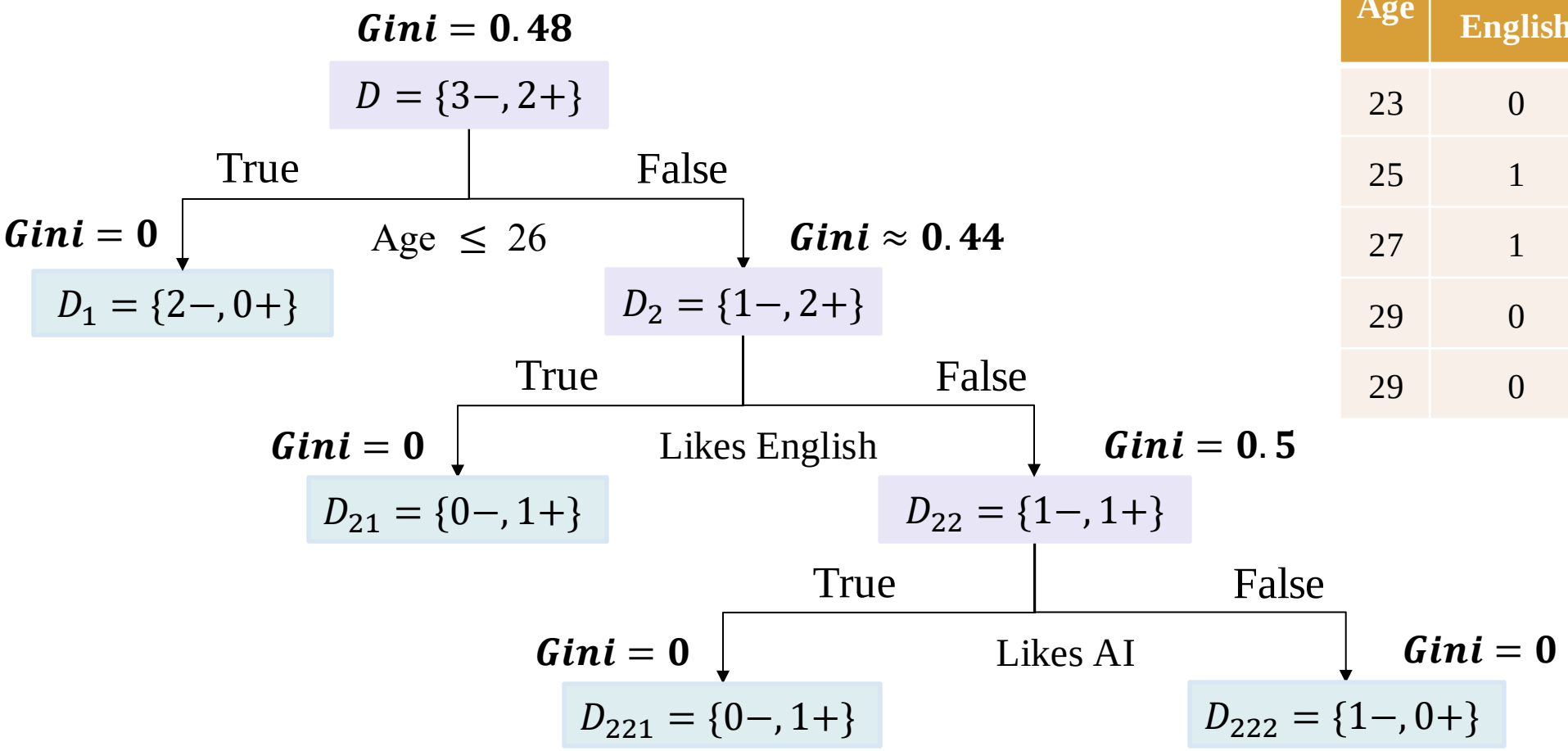
$D_{221} = \{1-, 1+\}$ $D_{222} = \{0-, 0+\}$

$Gini(\text{Age} \leq 29) = 0.5$

	Age	Likes English	Likes AI	Raise Salary
24	23	0	0	**0**
25	25	1	1	**0**
26	27	1	0	**1**
27	29	0	1	**1**
28	29	0	0	**0**
29				

DT for Classification

Gini Impurity



Age	Likes English	Likes AI	Raise Salary
23	0	0	**0**
25	1	1	**0**
27	1	0	**1**
29	0	1	**1**
29	0	0	**0**

DT for Classification

Gini Impurity

$$Gini(D) = \frac{n_1}{n} Gini(D_1) + \frac{n_2}{n} Gini(D_2)$$

$$Gini(D_i) = 1 - \sum_{j=1}^{c} p_j^2$$

p_j the probability that a random drawn sample from this node belongs to class j and c the number of classes

1. For each possible split, create child nodes and calculate the Gini Impurity of each child node.
2. Calculate the Gini Impurity of the split as the weighted average Gini Impurity of child nodes.
3. Select the split with the lowest Gini Impurity.

Dataset D (n sample)

Dataset D_1 (n_1 sample) | Dataset D_2 (n_2 sample)

	Age	Likes English	Likes AI	Raise Salary
24	23	0	0	**0**
26	25	1	1	**0**
27	27	1	0	**1**
29	29	0	1	**1**
	29	0	0	**0**

DT for Classification

Infomation Gain

$$Gain(D) = 1 - Entropy(D)$$

$$Entropy(D) = \frac{n_1}{n} Entropy(D_1) + \frac{n_2}{n} Entropy(D_2)$$

$$Entropy(D_i) = -\sum_{j=1}^{c} p_j \log_2 p_j$$

p_j the probability that a random drawn sample from this node belongs to class j and c the number of classes

1. For each possible split, create child nodes and calculate the Entropy of each child node.
2. Calculate the Entropy of the split as the weighted average Entropy of child nodes.
3. Select the split with the lowest Entropy or highest Information Gain, respectively.

Dataset D (n sample)

Dataset D_1 (n_1 sample)

Dataset D_2 (n_2 sample)

Age	Likes English	Likes AI	Raise Salary
23	0	0	**0**
25	1	1	**0**
27	1	0	**1**
29	0	1	**1**
29	0	0	**0**

DT for Classification

Infomation Gain

$$Gain(D) = 1 - Entropy(D)$$

$$Entropy(D) = \frac{n_1}{n} Entropy(D_1) + \frac{n_2}{n} Entropy(D_2)$$

$$Entropy(D_i) = -\sum_{j=1}^{c} p_j \log_2 p_j$$

Numerical Feature	Ascending Ordering	Calculate mean	Determine Entropy / Gain

$$D = \{3-, 2+\}$$

$$Entropy(D) = -\frac{3}{5}\log_2\frac{3}{5} - \frac{2}{5}\log_2\frac{2}{5} \approx 0.971$$

Age	Likes English	Likes AI	Raise Salary
23	0	0	**0**
25	1	1	**0**
27	1	0	**1**
29	0	1	**1**
29	0	0	**0**

DT for Classification

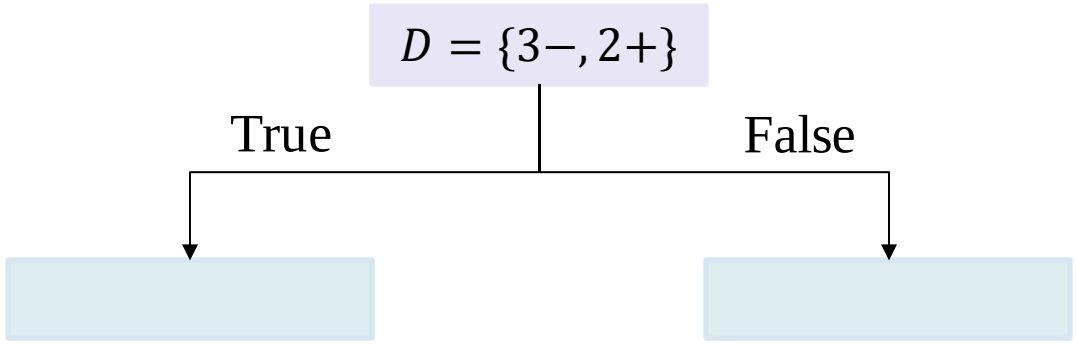
Infomation Gain

$$Gain(D) = 1 - Entropy(D)$$

$$Entropy(D) = \frac{n_1}{n} Entropy(D_1) + \frac{n_2}{n} Entropy(D_2)$$

$$Entropy(D_i) = -\sum_{j=1}^{c} p_j \log_2 p_j$$

Numerical Feature	Ascending Ordering	Calculate mean	Determine Entropy / Gain



$Gain(Likes\ English)$

$Gain(Likes\ AI)$

$Gain(\text{Age} \leq 24)$

$Gain(\text{Age} \leq 26)$

$Gain(\text{Age} \leq 28)$

$Gain(\text{Age} \leq 29)$

Age	Likes English	Likes AI	Raise Salary
23	0	0	**0**
25	1	1	**0**
27	1	0	**1**
29	0	1	**1**
29	0	0	**0**

DT for Classification

Infomation Gain

$$Gain(D) = 1 - Entropy(D)$$

$$Entropy(D) = \frac{n_1}{n} Entropy(D_1) + \frac{n_2}{n} Entropy(D_2)$$

$$Entropy(D_i) = -\sum_{j=1}^{c} p_j \log_2 p_j$$

Numerical Feature	Ascending Ordering	Calculate mean	Determine Entropy / Gain

$D = \{3-, 2+\}$

True — Likes English — False

$D_1 = \{1-, 1+\}$ $\quad$ $D_2 = \{2-, 1+\}$

$$Entropy(D_1) = -\frac{1}{2}\log_2\frac{1}{2} - \frac{1}{2}\log_2\frac{1}{2} = 1.0$$

$$Entropy(D_2) = -\frac{2}{3}\log_2\frac{2}{3} - \frac{1}{3}\log_2\frac{1}{3} \approx 0.92$$

$$Gain(Likes\ English) = 1 - \frac{2}{5} * 1.0 - \frac{3}{5} * 0.92 \approx 0.048$$

Age	Likes English	Likes AI	Raise Salary
23	0	0	**0**
25	1	1	**0**
27	1	0	**1**
29	0	1	**1**
29	0	0	**0**

DT for Classification

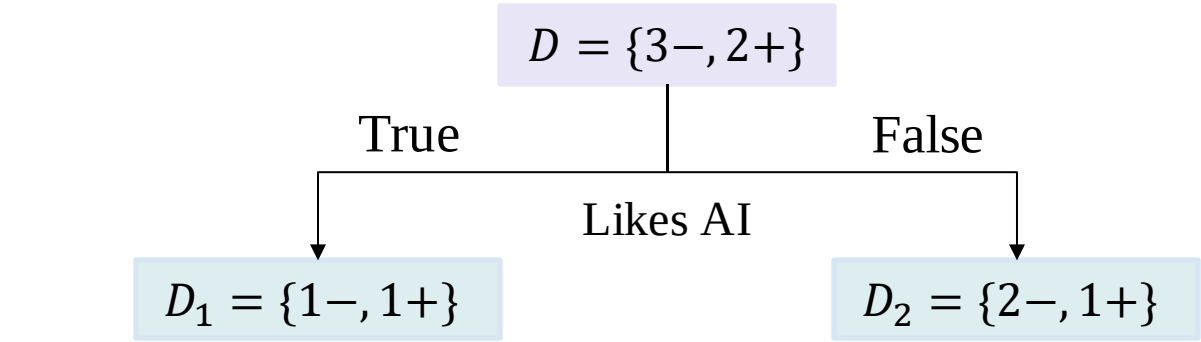
Infomation Gain

$$Gain(D) = 1 - Entropy(D)$$

$$Entropy(D) = \frac{n_1}{n} Entropy(D_1) + \frac{n_2}{n} Entropy(D_2)$$

$$Entropy(D_i) = -\sum_{j=1}^{c} p_j \log_2 p_j$$

Numerical Feature	Ascending Ordering	Calculate mean	Determine Entropy / Gain



$$Entropy(D_1) = -\frac{1}{2}\log_2\frac{1}{2} - \frac{1}{2}\log_2\frac{1}{2} = 1.0$$

$$Entropy(D_2) = -\frac{2}{3}\log_2\frac{2}{3} - \frac{1}{3}\log_2\frac{1}{3} \approx 0.92$$

$$Gain(Likes\ AI) = 1 - \frac{2}{5} * 1.0 - \frac{3}{5} * 0.92 \approx 0.048$$

Age	Likes English	Likes AI	Raise Salary
23	0	0	**0**
25	1	1	**0**
27	1	0	**1**
29	0	1	**1**
29	0	0	**0**

DT for Classification

Infomation Gain

$$Gain(D) = 1 - Entropy(D)$$

$$Entropy(D) = \frac{n_1}{n} Entropy(D_1) + \frac{n_2}{n} Entropy(D_2)$$

$$Entropy(D_i) = -\sum_{j=1}^{c} p_j \log_2 p_j$$

Numerical Feature	Ascending Ordering	Calculate mean	Determine Entropy / Gain

$D = \{3-, 2+\}$

True | False

$Age \leq 26$

$D_1 = \{2-, 0+\}$

$D_2 = \{1-, 2+\}$

$$Entropy(D_1) = -\frac{2}{2}\log_2\frac{2}{2} - \frac{0}{2}\log_2\frac{0}{2} = 0.0$$

$$Entropy(D_2) = -\frac{1}{3}\log_2\frac{1}{3} - \frac{2}{3}\log_2\frac{2}{3} \approx 0.92$$

$$Gain(Age \leq 26) = 1 - \frac{2}{5} * 0.0 - \frac{3}{5} * 0.92 \approx 0.448$$

Age	Likes English	Likes AI	Raise Salary
23	0	0	**0**
25	1	1	**0**
27	1	0	**1**
29	0	1	**1**
29	0	0	**0**

DT for Classification

Constructing Decision Tree

$$Gain(D) = 1 - Entropy(D)$$

$$Entropy(D) = \frac{n_1}{n} Entropy(D_1) + \frac{n_2}{n} Entropy(D_2)$$

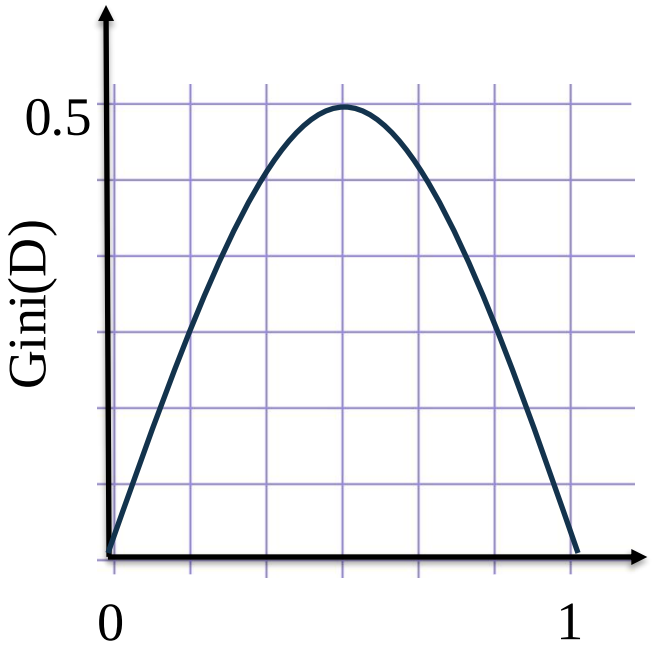
$$Entropy(D_i) = -\sum_{j=1}^{c} p_j \log_2 p_j$$

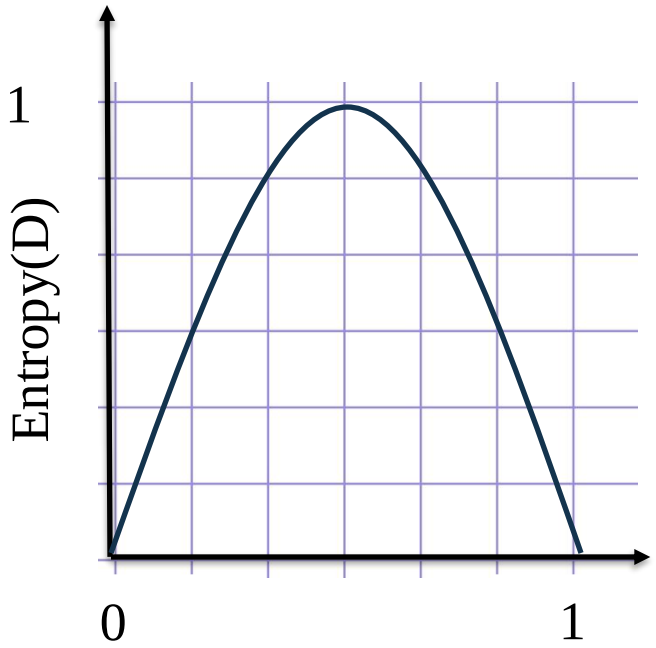
$$Gini(D) = \frac{n_1}{n} Gini(D_1) + \frac{n_2}{n} Gini(D_2)$$

$$Gini(D_i) = 1 - \sum_{j=1}^{c} p_j^2$$

$p_1 = p_2$
$\Rightarrow Gini(D) = 0.5$
$\Rightarrow Entropy(D) = 1.0$

$p_1 or\ p_2 = 0$
$\Rightarrow Gini(D) = 0.0$
$\Rightarrow Entropy(D) = 0.0$





DT for Classification

Decision Tree for IRIS Dataset

```
# Load the diabetes dataset
iris_X, iris_y = datasets.load_iris(return_X_y=True)

# Split train:test = 8:2
X_train, X_test, y_train, y_test = train_test_split(
    iris_X, iris_y, test_size=0.2, random_state=42
)

# Train
dt_classifier = DecisionTreeClassifier()
dt_classifier.fit(X_train, y_train)

# Predict
y_pred = dt_classifier.predict(X_test)

# Evaluation
accuracy_score(y_test, y_pred)
```


QUIZ TIME

Outline

SECTION 1

Decision Tree

SECTION 2

DT for Classification

SECTION 3

DT for Regression

SECTION 4

Improved DT

Dataset D (n sample)

Dataset D_1 (n_1 sample)

Dataset D_2 (n_2 sample)

Age	Likes English	Likes AI	Salary
23	0	0	**200**
25	1	1	**400**
27	1	0	**300**
29	0	1	**500**
29	0	0	**400**

DT for Regression

Decision Tree

- Example Dataset

Age	Likes English	Likes AI	Salary
23	0	0	**200**
25	1	1	**400**
27	1	0	**300**
29	0	1	**500**
29	0	0	**400**

DT for Regression

Sum of Squared Errors (SSE)

$$SSE(D) = SSE(D_1) + SSE(D_2)$$

$$SSE(D_i) = \frac{1}{n_i}\sum_{j=1}^{n_i}(x_j - \bar{x}_i)^2$$

$x_i, \dots, x_{n_i}$ the items of D_i

1. For each possible split, create child nodes and calculate the variance of each child node.
2. Calculate the variance of the split as the weighted average variance of child nodes.
3. Select the split with the lowest variance.

Dataset D (n sample)

Dataset D_1 (n_1 sample) | Dataset D_2 (n_2 sample)

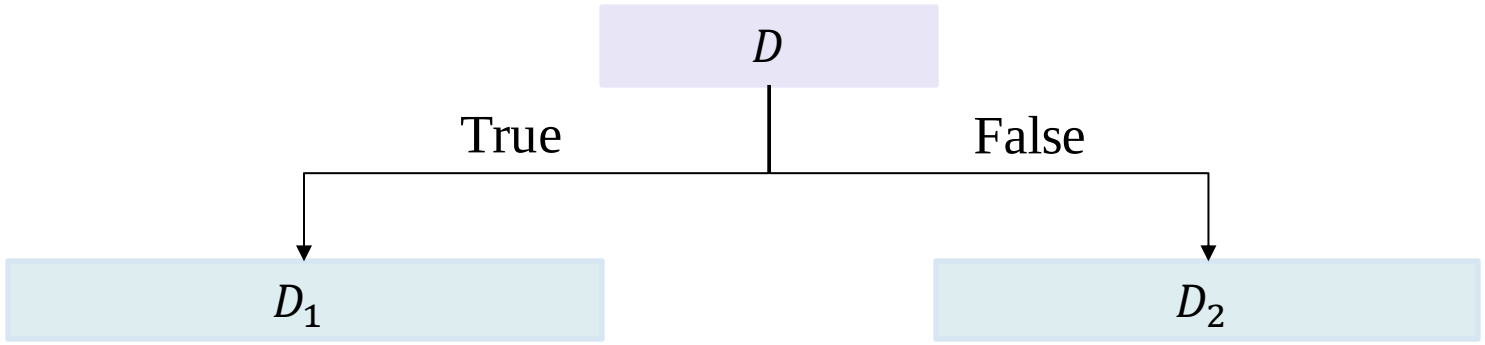
Age	Likes English	Likes AI	Salary
23	0	0	**200**
25	1	1	**400**
27	1	0	**300**
29	0	1	**500**
29	0	0	**400**

DT for Regression

Sum of Squared Errors (SSE)

$$SSE(D) = SSE(D_1) + SSE(D_2)$$

$$SSE(D_i) = \frac{1}{n_i}\sum_{j=1}^{n_i}(x_j - \bar{x}_i)^2$$



$SSE(Likes\ English)$

$SSE(Likes\ AI)$

$SSE(\text{Age} \leq 24)$

$SSE(\text{Age} \leq 26)$

$SSE(\text{Age} \leq 28)$

$SSE(\text{Age} \leq 29)$

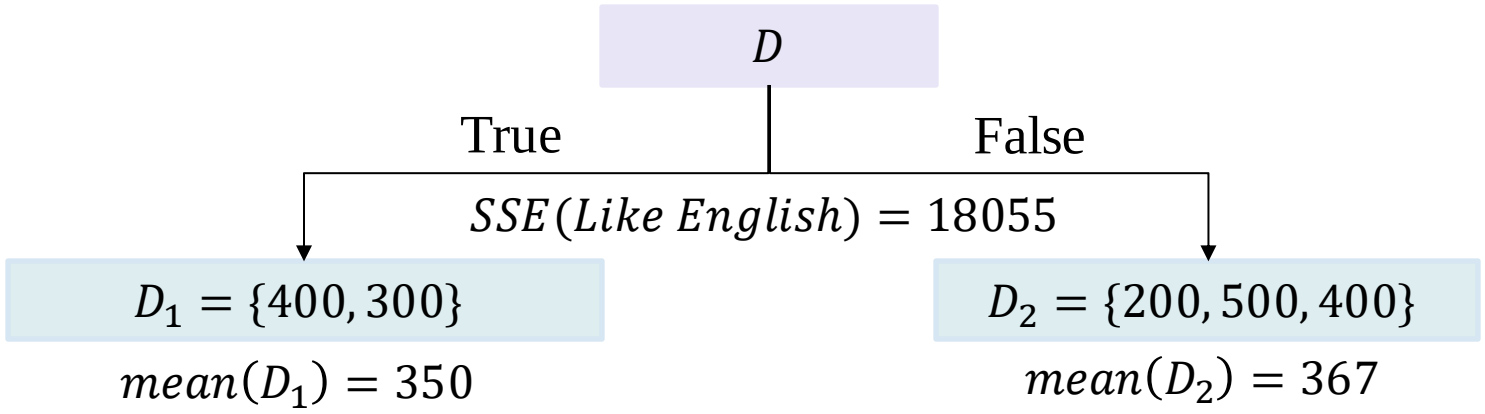
Age	Likes English	Likes AI	Salary
23	0	0	**200**
25	1	1	**400**
27	1	0	**300**
29	0	1	**500**
29	0	0	**400**

DT for Regression

Sum of Squared Errors (SSE)

$$SSE(D) = SSE(D_1) + SSE(D_2)$$

$$SSE(D_i) = \frac{1}{n_i}\sum_{j=1}^{n_i}(x_j - \bar{x}_i)^2$$



$$SSE(D_1) = \frac{(400-350)^2 + (300-350)^2}{2} = 2500$$

$$SSE(D_2) = \frac{(200-367)^2 + (500-367)^2 + (400-367)^2}{3} = 15555$$

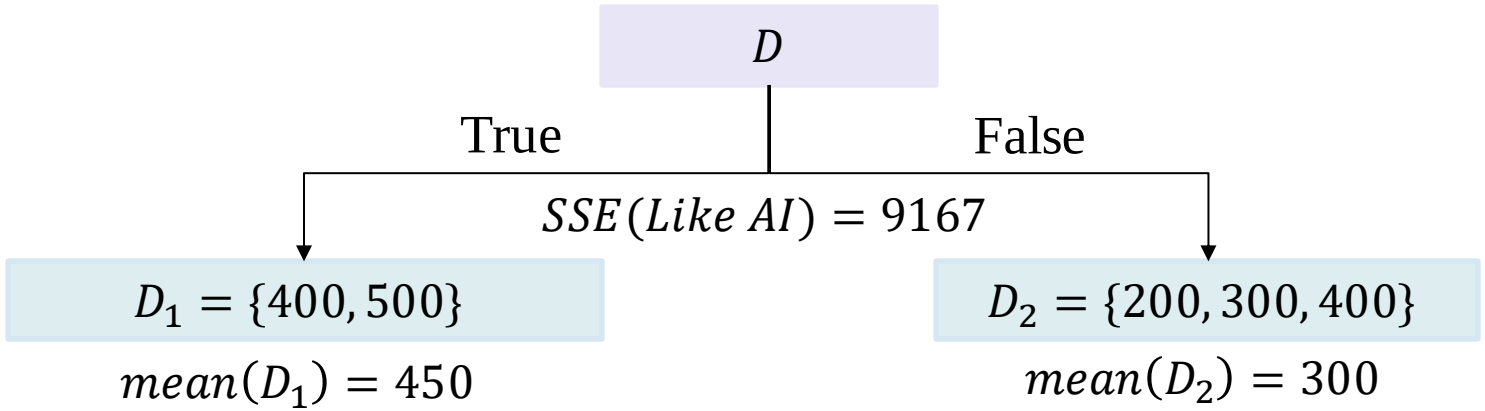
Age	Likes English	Likes AI	Salary
23	0	0	**200**
25	1	1	**400**
27	1	0	**300**
29	0	1	**500**
29	0	0	**400**

DT for Regression

Sum of Squared Errors (SSE)

$$SSE(D) = SSE(D_1) + SSE(D_2)$$

$$SSE(D_i) = \frac{1}{n_i}\sum_{j=1}^{n_i}(x_j - \bar{x}_i)^2$$



$$SSE(D_1) = \frac{(400-450)^2 + (500-450)^2}{2} = 2500$$

$$SSE(D_2) = \frac{(200-300)^2 + (300-300)^2 + (400-300)^2}{3} = 6667$$

Age	Likes English	Likes AI	Salary
23	0	0	**200**
25	1	1	**400**
27	1	0	**300**
29	0	1	**500**
29	0	0	**400**

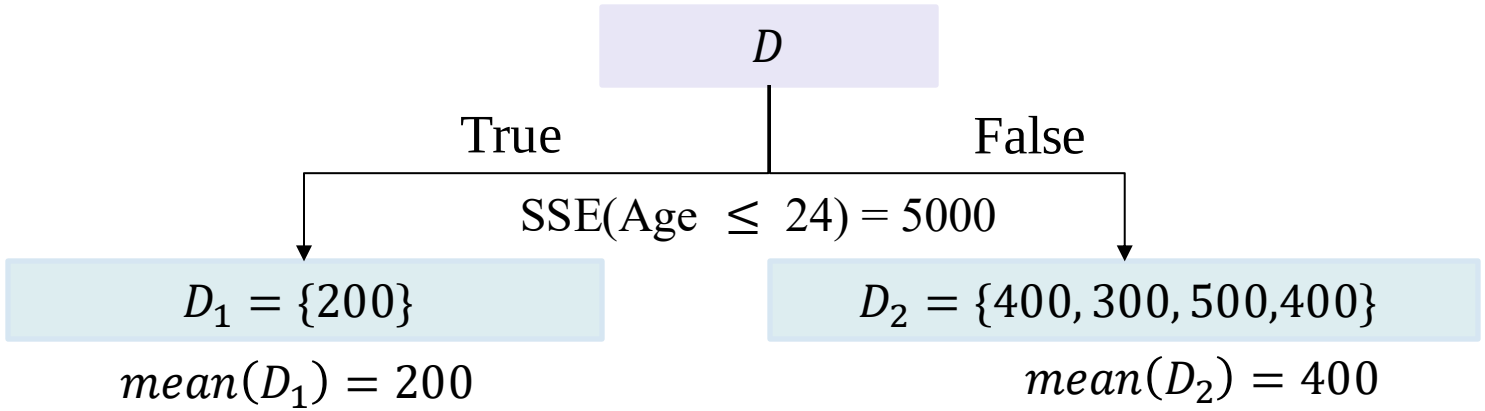
DT for Regression

Sum of Squared Errors (SSE)

$$SSE(D) = SSE(D_1) + SSE(D_2)$$

$$SSE(D_i) = \frac{1}{n_i}\sum_{j=1}^{n_i}(x_j - \bar{x}_i)^2$$

Numerical Feature	Ascending Ordering	Calculate mean	Determine Gini



$$SSE(D_1) = \frac{(200-200)^2}{1} = 0$$

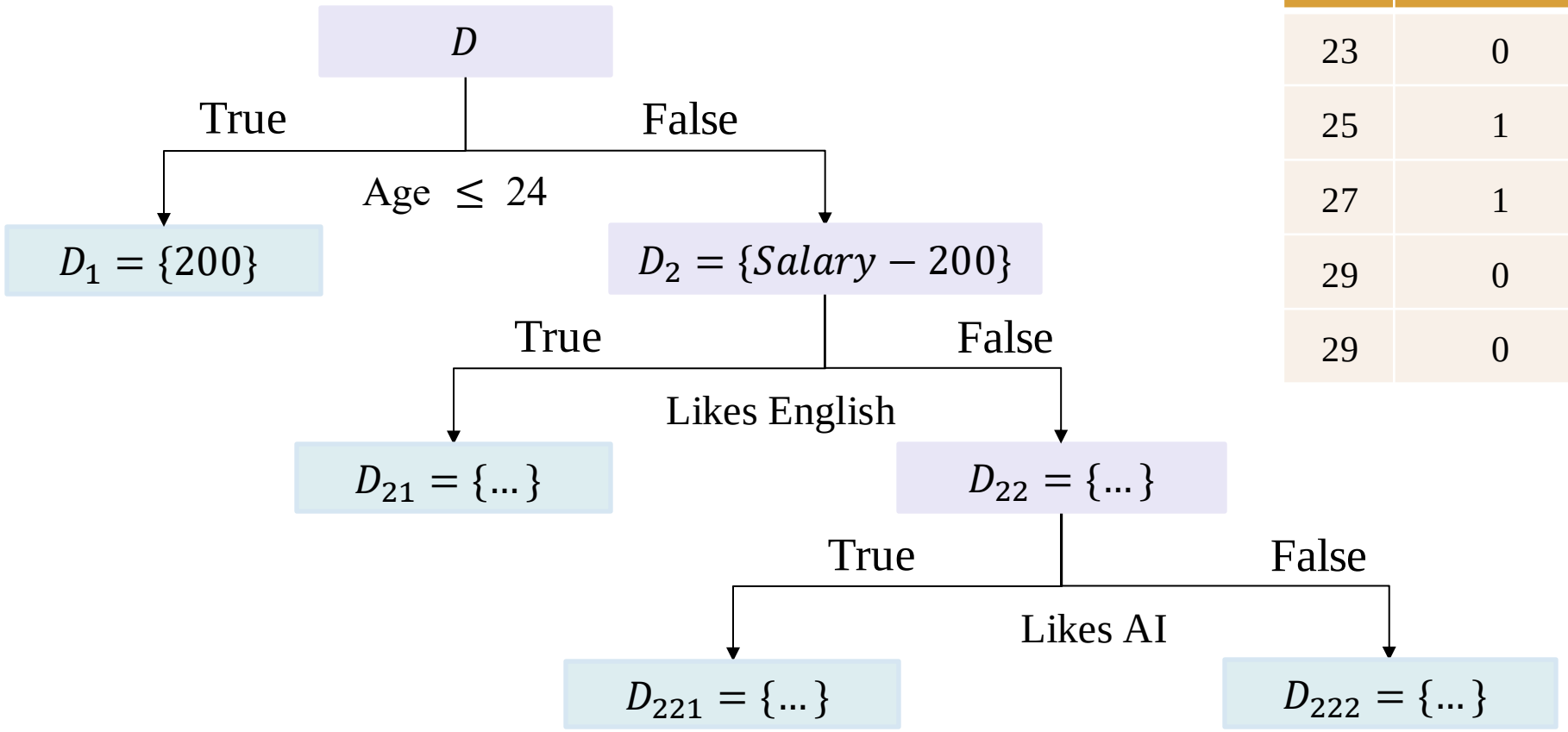
$$SSE(D_2) = \frac{(400-400)^2 + (300-400)^2 + (500-400)^2 + (400-400)^2}{4} = 5000$$

Age	Likes English	Likes AI	Salary
23	0	0	**200**
25	1	1	**400**
27	1	0	**300**
29	0	1	**500**
29	0	0	**400**

DT for Regression

Sum of Squared Errors (SSE)

Age	Likes English	Likes AI	Salary
23	0	0	**200**
25	1	1	**400**
27	1	0	**300**
29	0	1	**500**
29	0	0	**400**



DT for Regression

Decision Tree for CPU Machine Regression Dataset

```
# Load dataset
from sklearn.datasets import fetch_openml

machine_cpu = fetch_openml(name='machine_cpu')

machine_data = machine_cpu.data
machine_labels = machine_cpu.target

X_train, X_test, y_train, y_test = train_test_split(
    machine_data, machine_labels, test_size=0.2,random_state=20
)

# train
tree_reg = DecisionTreeRegressor()
tree_reg.fit(X_train, y_train)

# predict
y_pred = tree_reg.predict(X_test)

# evaluation
mean_squared_error(y_test, y_pred)
```

Outline

SECTION 1

Decision Tree

SECTION 2

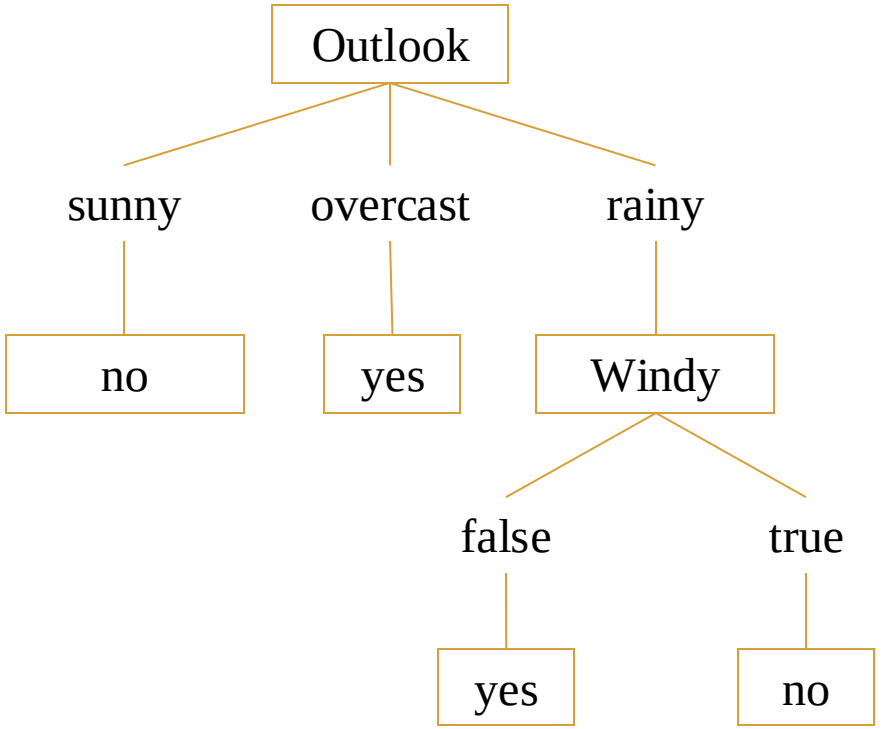
DT for Classification

SECTION 3

DT for Regression

SECTION 4

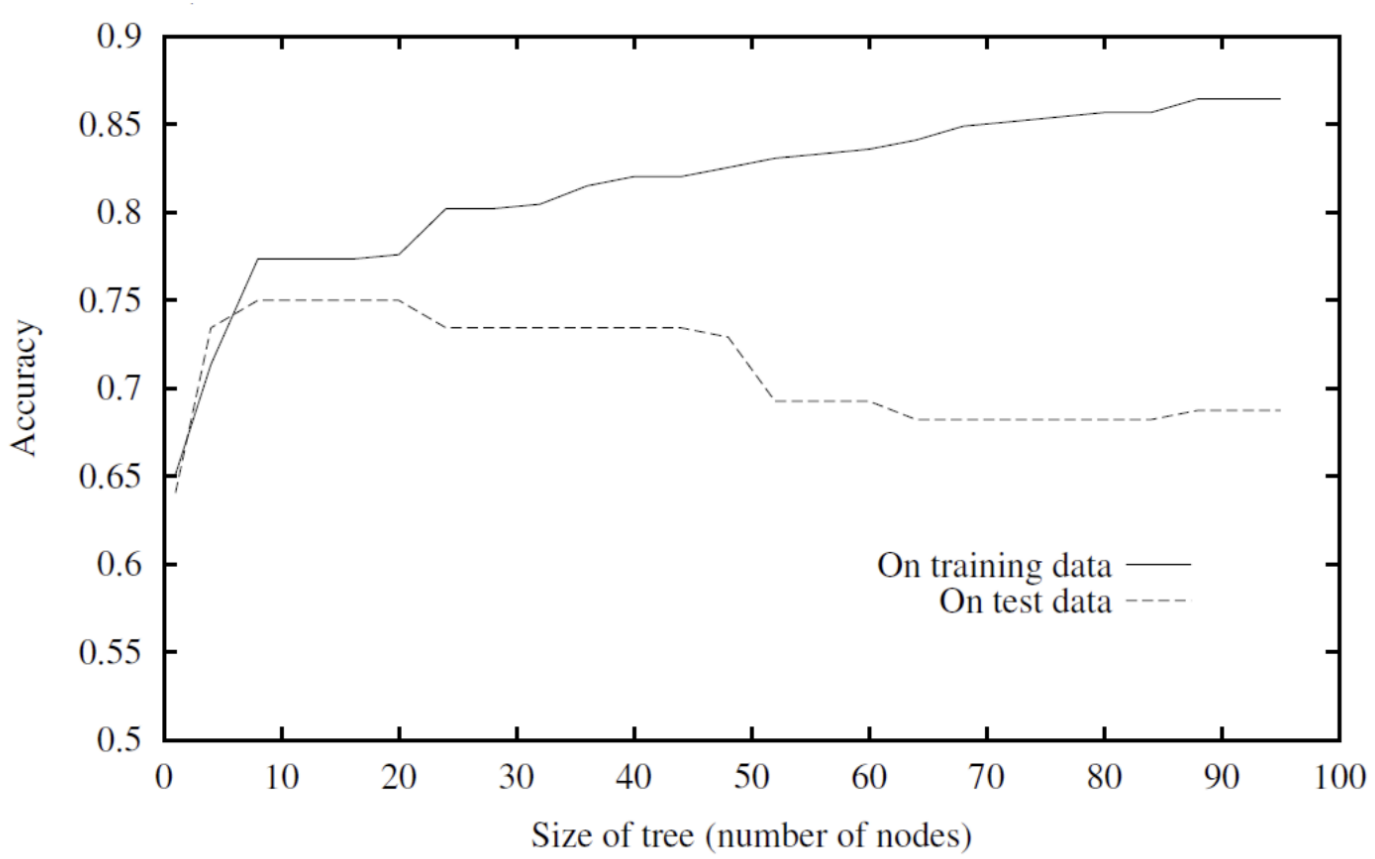
Improved DT



Improved Decision Tree

Over-fitting

- Desired: a DT that is not too big in size, yet fits the training data reasonably



Source

Improved Decision Tree

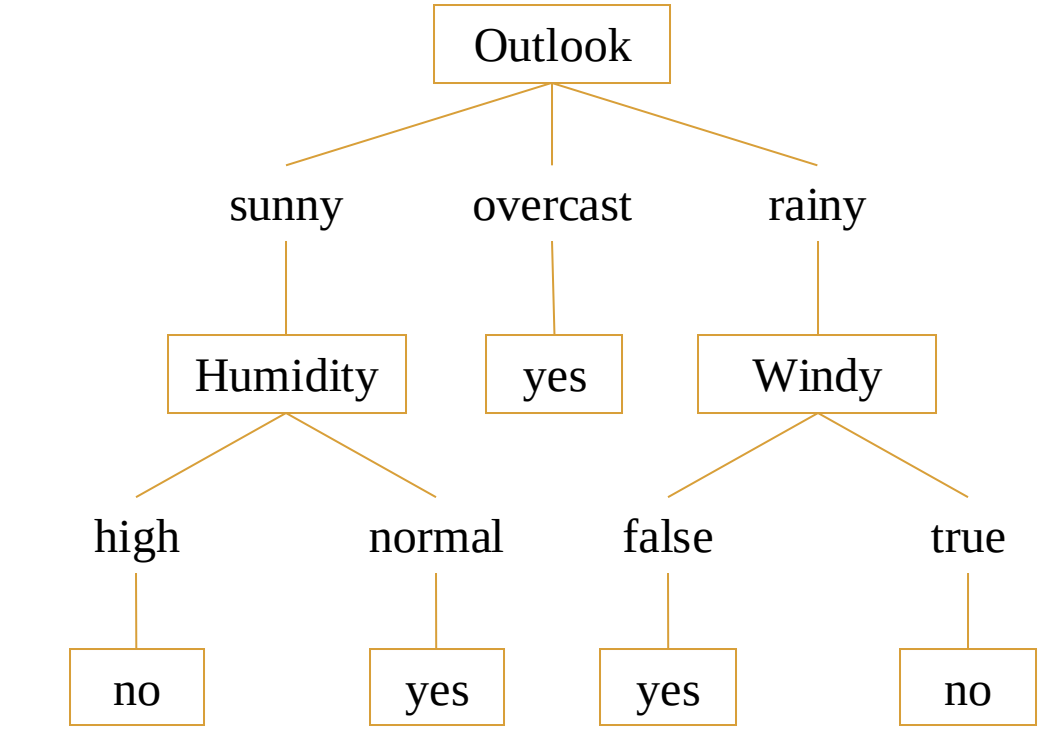
Over-fitting

- Mainly two approaches
  - Prune while building the tree (Stopping Early)
  - Prune after building the tree (Post-Pruning)
- Criteria for judging which nodes could potentially be pruned: evaluate validation set
  - Reduced-error pruning
  - Rule post-pruning

Improved Decision Tree

Reduced-Error Pruning

- Pruning a decision node *d* consists of:
  - Removing the subtree rooted at d
  - Making *d* a leaf node
  - Assigning *d* the most common classification of the training instances associated with *d*.



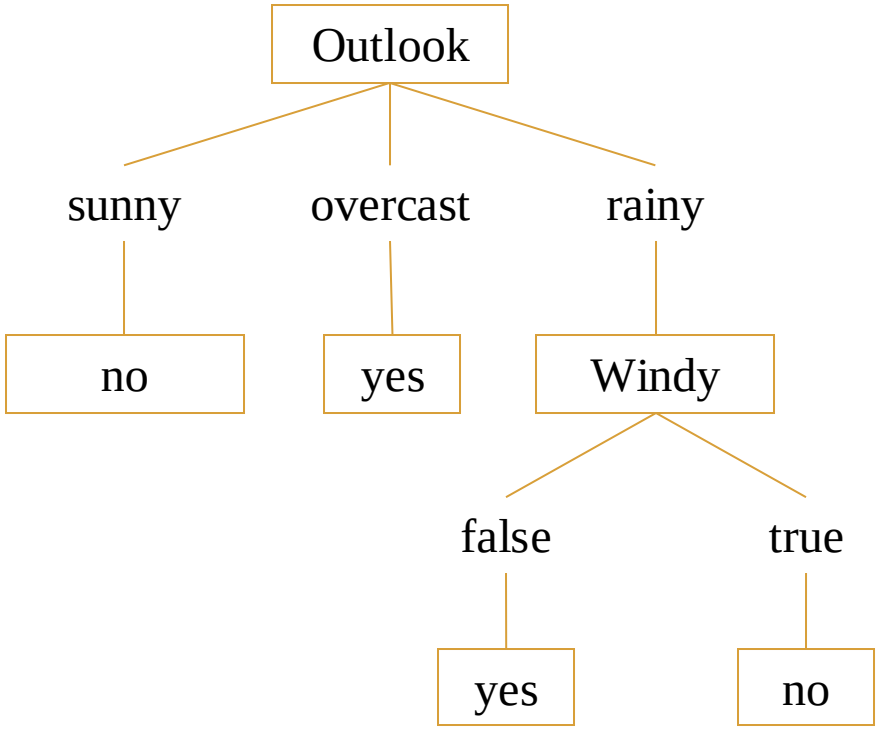
Accuracy of the tree on the validation set is 90%.

Improved Decision Tree

Reduced-Error Pruning

- Pruning a decision node *d* consists of:
  - Removing the subtree rooted at *d*
  - Making *d* a leaf node
  - Assigning *d* the most common classification of the training instances associated with *d*.

- Do until further pruning is harmful:
  - Evaluate impact on validation set of pruning each possible node
  - Greedily remove the one that most improves validation set accuracy.



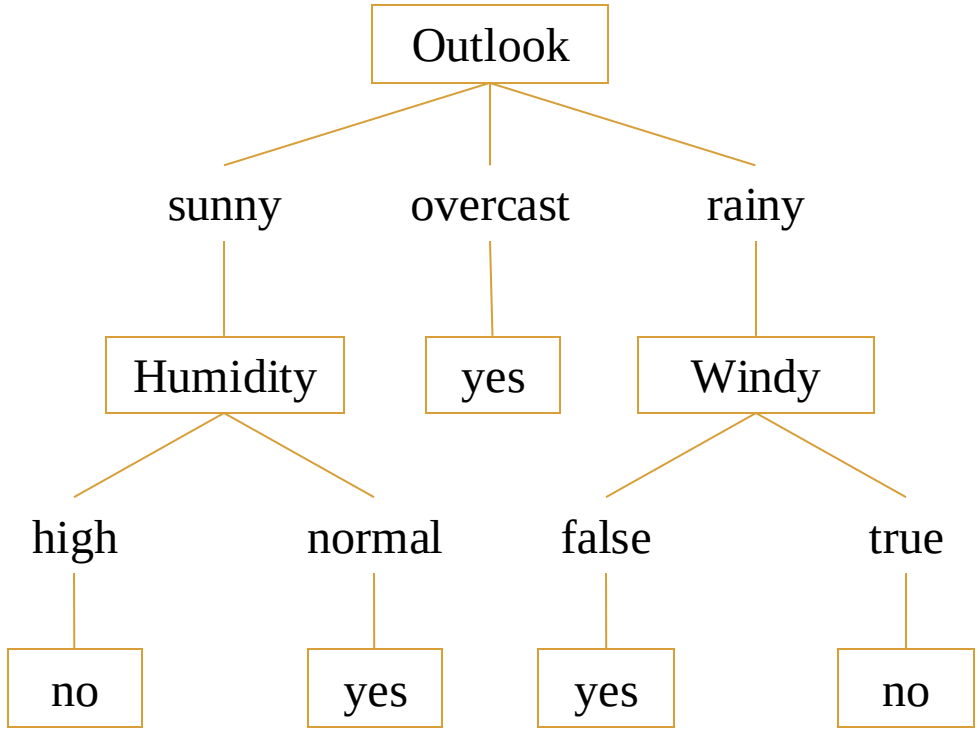
Accuracy of the tree on the validation set is 92.4%.

Improved Decision Tree

Rule Post-Pruning

- Convert tree to equivalent set of rules.
- Prune each rule independently of others
- Soft final rules by their estimated accuracy and consider them in this sequence when classifying subsequent instances.

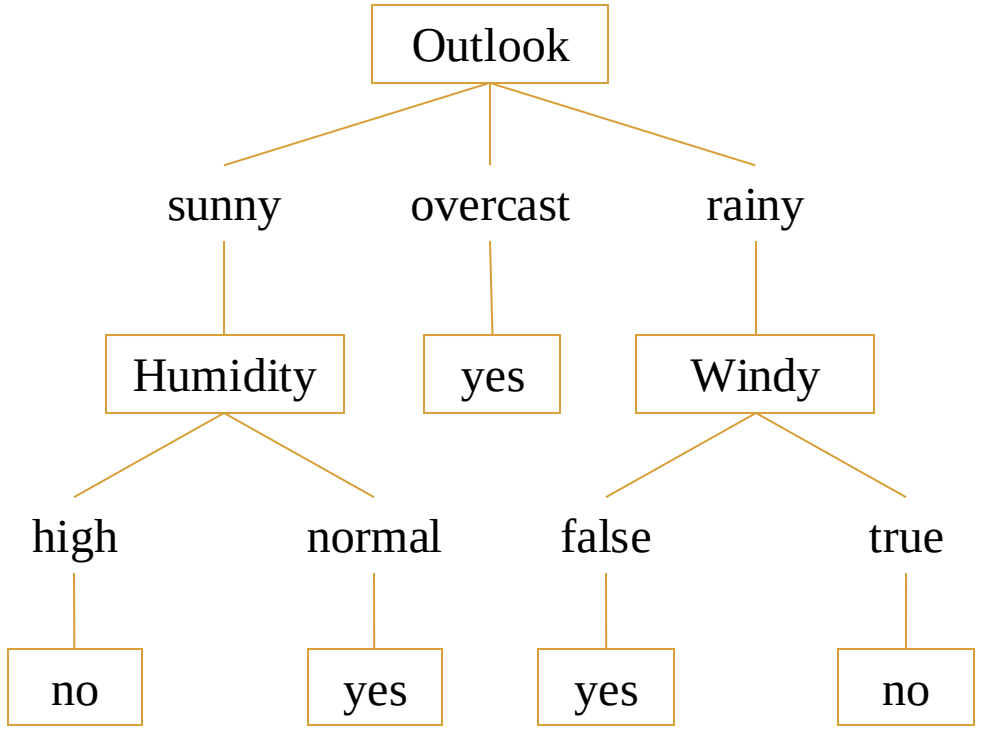
IF (Outlook = Sunny) & (Humidity = High)
THEN PlayTennis = No
IF (Outlook = Sunny) & (Humidity = Normal)
THEN PlayTennis = Yes
……



Improved Decision Tree

Missing Attribute Values

- Assign most common value of A among other instances belonging to the same concept.
- If node n test the attribute A, assign most common value of A among other instances sorted to node n.
- If node n tests the attribute A, assign a probability to each of possible values of A. These probabilities are estimated based on the observed frequencies of the values of A (Based on information gain.



Test = {Outlook=Sunny, **Humidity=?**, Windy=True}

Summary

Decision Tree

- Introduction
- Regression & Classification Problem
- Terminology

Decision Tree for Regression

- Variance
- Sum of Squared Errors (SSE)
- Based on Sklearn Library

Decision Tree for Classification

- Constructing Decision Tree: ID3
- Gini Impurity
- Entropy
- Information Gain
- Based on Sklearn library

Improved Decision Tree

- Overfitting
- Stopping Early
- Post-Pruning (Reduced-Error & Rule Post-Pruning)
- Missing Attribute Values

Thanks!

Any questions?