

DT Classifier

1. Purity (Impurity)

1.1 Entropy

1.2 Gini Index

2. Information Gain

2.1 Categorical Features

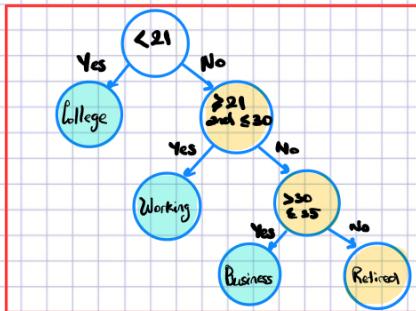
2.2 Numerical Features

DT Regressor

Decision Tree Classifier

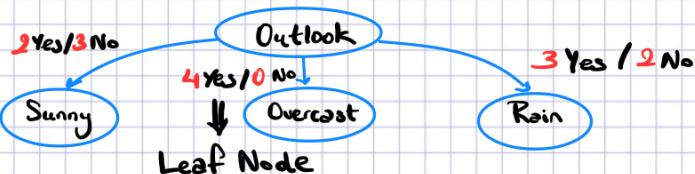
D3 is nested if-else clause:

```
Ex. if (p-age < 21):
    print ('college')
elif (p-age ≥ 21 and p-age ≤ 30):
    print ('work')
elif (p-age ≥ 30 and p-age ≤ 35):
    print ('business')
```



Decision Tree: 1. ID3 Algorithm

2. CART (Classification and Regression Tree)



1. Purity (Impurity Criterion)

- 1.1 Entropy $I_H = -\sum_{j=1}^C p_j \log_2(p_j)$
- 1.2 Gini Index $I_G = 1 - \sum_{j=1}^C p_j^2$

2. Information Gain [How features are selected]

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

PlayTennis: training examples

9 Yes / 5 No

C : number of classes
 p_j : probability of randomly picking an element of class i .

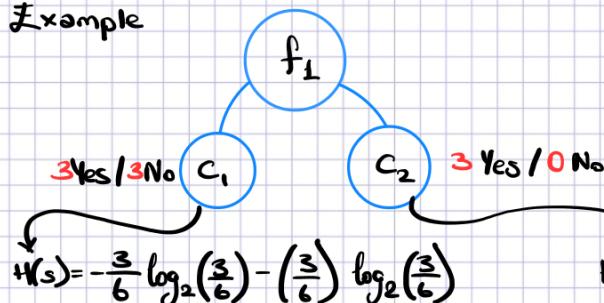
1. Purity

1.1 Entropy

$$\text{binary } H(s) = -p_+ \log_2 p_+ - p_- \log_2 p_-$$

$$\text{multi } H(s) = -p_{c_1} \log_2 p_{c_1} - p_{c_2} \log_2 p_{c_2} - p_{c_3} \log_2 p_{c_3}$$

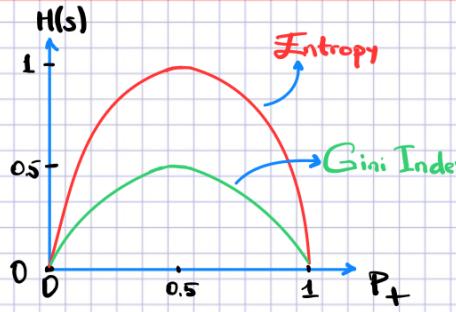
Example



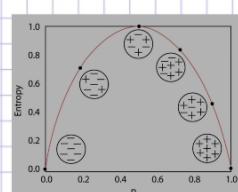
$$H(s) = -\frac{3}{6} \log_2 \left(\frac{3}{6}\right) - \left(\frac{3}{6}\right) \log_2 \left(\frac{3}{6}\right)$$

 $H(s) = 1$ Very impure split

$$H(s) = -\frac{3}{3} \log_2 \left(\frac{3}{3}\right) =$$

 $= -\log_2(1) = 0$ Pure Split

$$\log_b a = x \Leftrightarrow b^x = a$$



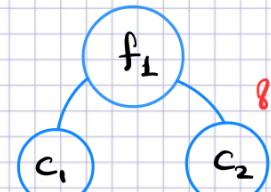
$$H(s) = -\frac{Y}{Y+N} \log_2 \left(\frac{Y}{Y+N}\right) - \frac{N}{Y+N} \log_2 \left(\frac{N}{Y+N}\right)$$

1.2 Gini Index

$$G.I. = 1 - \sum_{i=1}^C (p_i)^2$$

$$\text{Example: } GI = 1 - \left[(P_+)^2 + (P_-)^2 \right] = 1 - \left[\left(\frac{1}{2}\right)^2 + \left(\frac{1}{2}\right)^2 \right] = 1 - \frac{1}{2} = 0.5$$

p stands for "proportion" or probability of randomly picking an element of class i .

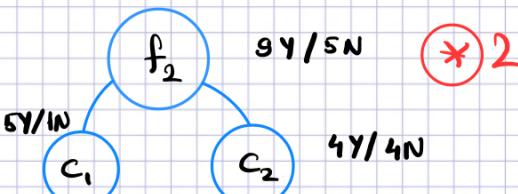
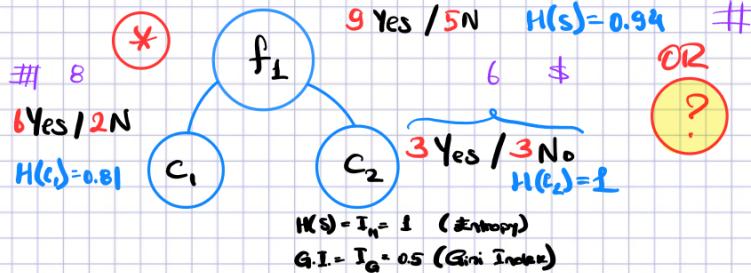


8 Yes / 2 No

$$G.I. = 1 - \left[\left(\frac{8}{10}\right)^2 + \left(\frac{2}{10}\right)^2 \right] = 1 - \left(\frac{64}{100} + \frac{4}{100} \right) = 1 - 0.68 = 0.32$$

2. Information Gain

2.1 Categorical Features



Feature, that comes first into order of splitting, can be defined by Information Gain.

Gain (s, f_i): Gain of ' f_i ' with respect to the sample 's'.

$$\text{Gain}(s, f_i) = H(s) - \sum_{v \in \text{values}} \frac{|s_v|}{|s|} \cdot H(s_v)$$

Information Gain is calculated for the entire tree, not 1 Node

Example #1: $H(s) = -P_+ \log_2 P_+ - P_- \log_2 P_-$

$$= -\left(\frac{9}{14}\right) \cdot \log_2\left(\frac{9}{14}\right) - \left(\frac{5}{14}\right) \cdot \log_2\left(\frac{5}{14}\right) \approx 0.94$$

$$H(c_1) = -\left(\frac{9}{14}\right) \cdot \log_2(0.75) - 0.125 \cdot \log_2(0.125) = 0.81$$

$$H(c_2) = -\left(\frac{3}{6}\right) \cdot \log_2\left(\frac{3}{6}\right) - \left(\frac{3}{6}\right) \cdot \log_2\left(\frac{3}{6}\right) = \log_2\left(\frac{1}{2}\right) \cdot (-1) = -\log_2\left(\frac{1}{2}\right) = 1$$

$$\text{Gain}(s, f_1) = 0.94 - \left[\frac{9}{14} \cdot 0.81 + \frac{6}{14} \cdot 1 \right] = 0.0486$$

Example #2: $H(s) = 0.94$

$$H(c_1) = -\frac{5}{6} \log_2\left(\frac{5}{6}\right) - \frac{1}{6} \log_2\left(\frac{1}{6}\right) = -\frac{5}{6} \cdot (-0.23) - \frac{1}{6} \cdot (-2.58) = 0.19 + 0.43 = 0.62$$

$$H(c_2) = 1$$

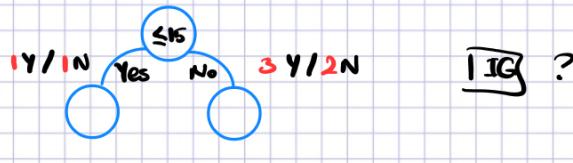
$$\text{Gain}(s, f_2) = 0.94 - \left[\frac{6}{14} \cdot 0.62 + \frac{8}{14} \cdot 1 \right] = 0.94 - 0.837 = 0.103$$

I.G. (f_2) > I.G. (f_1)

2.2 Numerical Features

CART

	f_1	f_2	%P
10	-	0	
15	-	1	
20	-	1	
25	-	0	
30	-	1	
35	-	0	
40	-	1	



Decision Tree Regressor

$f_1 \quad f_2$

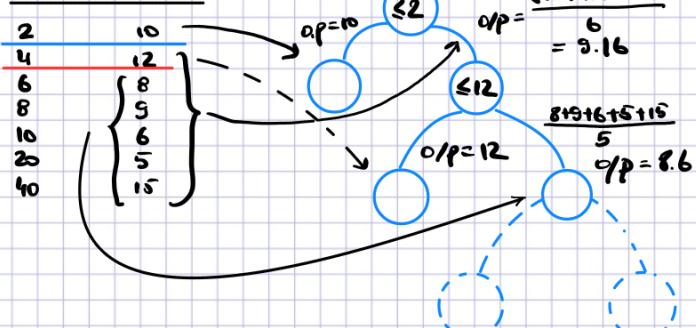
		Price
-	-	20
-	-	24
-	-	28
-	-	14
-	-	16
-	-	20

$$MSE = \frac{1}{2n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

mean

Example:

$f_1 \quad o/p$

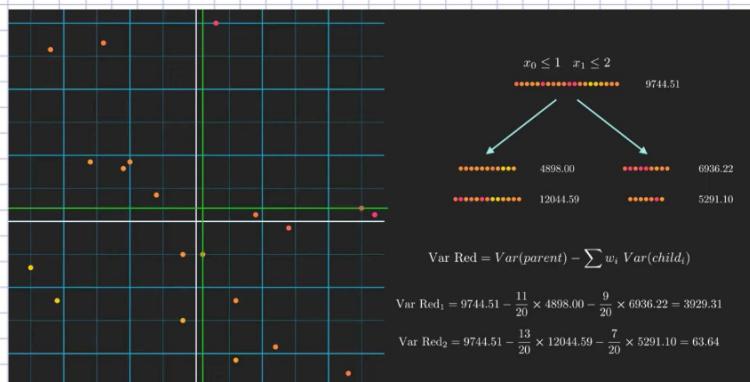
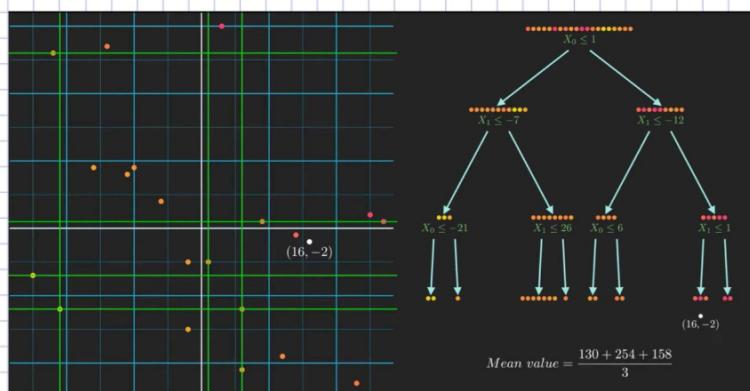
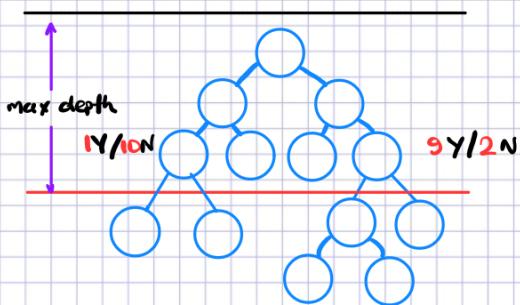


The main goal, while splitting is to minimize MSE or MAE

Overfitting: Training accuracy \uparrow { Low bias }
 Test Accuracy \downarrow { High variance }

To avoid overfitting in Decision Tree Algorithm we can use:

1. post pruning
2. pre pruning



$$\text{Var Red} = \text{Var}(\text{parent}) - \sum w_i \text{Var}(\text{child}_i)$$

$$\text{Var Red}_1 = 9744.51 - \frac{11}{20} \times 4898.00 - \frac{9}{20} \times 6936.22 = 3929.31$$

$$\text{Var Red}_2 = 9744.51 - \frac{13}{20} \times 12044.59 - \frac{7}{20} \times 5291.10 = 63.64$$