Decision Trees

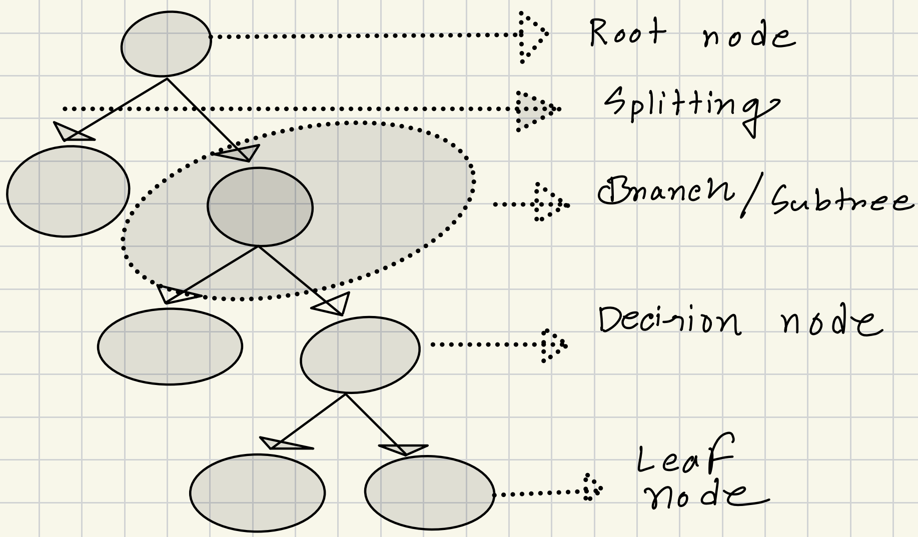
One of the more intuitive methods. Kind of nested if-else conditions.

→ Is a classification problem.

* Pseudo Code

1. Begin with your training dataset, which should have some feature variables and classification or regression output.
2. Determine the "best feature" in the dataset to split the data on.
3. Split the data into subsets that contain the correct values for this best feature. This splitting basically defines a node on the tree. (Each node is a splitting point based on our data)
4. Recursively generate new tree nodes by using the subset of data created from step 3.

* Terminology



Advantages

1) Intuitive & easy to understand
2) Minimal data preparation is required.
3) The cost is logarithmic in the number of data points used to train the tree.

Disadvantages:

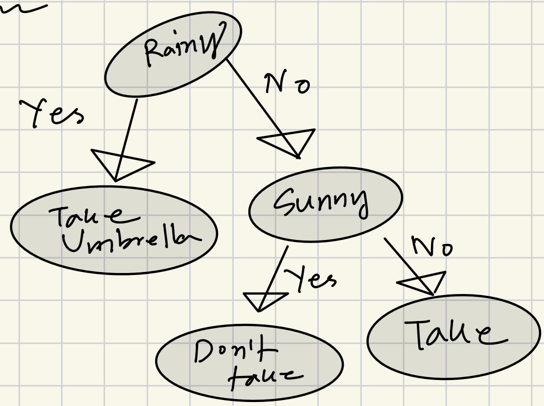
1) Overfitting.
2) Prone to errors for imbalanced dataset.

Weird fact:

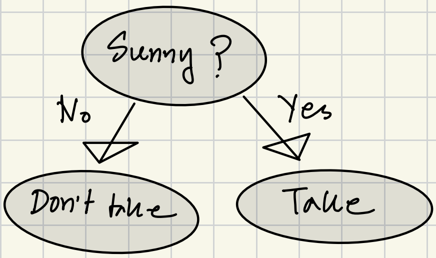
DTs are mainly used for classification problem but can also be used for regression problem.

Sunny	Rainy	Decision	
Yes	No	Nein	
- No	Yes	Ja	
- No	No	Ja	

Tree 1



Tree 2:



* Calculating Entropy

$$E(Sunny=Yes) = -\left(P_{DT}\log_2(P_{DT}) + P_{TU}\log_2(P_{TU})\right)$$

$$= -\left(\frac{1}{1}\log_2\left(\frac{1}{1}\right) + \frac{0}{1}\log_2\left(\frac{0}{1}\right)\right) = 0$$

$$E(Sunny=No) = -\left(P_{DT}\log_2(P_{DT}) + P_{TU}\log_2(P_{TU})\right)$$

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

$$= 0$$

In both cases, entropy is 0. That means this is pure.

Using ID3 to calculate best feature

features — Label

Day	Outlook	Temp.	Humidity	Wind	Decision
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No
7	Overcast	Cool	Normal	Strong	Yes
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
10	Rain	Mild	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

Step 1: Calculate entropy of decision

$(P=Yes) = 9/14 \quad (P=No) = 5/14$

$\therefore E(Decision) = -\frac{9}{19}\log_2\frac{9}{19} - \frac{5}{19}\log_2\frac{5}{19}$

$= 0.940$

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

$= 0.971$

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

$= 0.971$

$E(Outlook = OC) = 0$

Probability of apprarance

$\therefore$ Information gain of outlook $= E(Decision) - \Sigma\, E(Outlook) \times Weight$

$= 0.940 - \frac{5}{19}\times 0.971 - \frac{5}{19}\times 0.971$

$- \frac{4}{19}\times 0$

$= 0.246$

$\therefore IG(Outlook) = 0.246$

* IG of wind calc

$\therefore IG(Wind) = 0.08$

$IG(Wind) = E(Decision) - \Sigma$ Probability of outcome $\times E(wind = x)$

$= 0.940 - \left(\frac{6}{19}\times -\left(\frac{6}{8}\log_2\left(\frac{6}{8}\right) + \frac{6}{8}\log_2\left(\frac{6}{8}\right)\right)\right) - \left(\frac{8}{19}\times -\left(\frac{3}{6}\log_2\left(\frac{3}{6}\right) + \frac{3}{6}\log_2\left(\frac{3}{6}\right)\right)\right)$

$= 0.940 - 0.29 - 0.57 = 0.08$

In the same way,

$IG(\text{Humidity}) = 0.151$

We'll select the feature with most IG & then we subdivide the dataset and so on and so forth.

* Entropy :

$$\boxed{E = \sum_{i=1}^{c} - P_i \log_2 P_i}$$

$P_i \rightarrow$ Simply the frequentist probability of an element / class 'i' in our data.

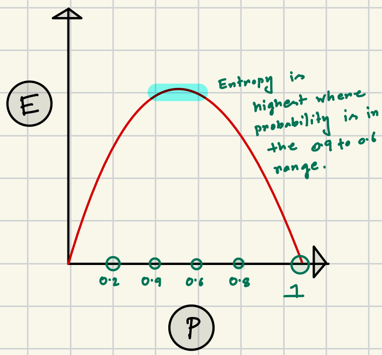
Salary	Age	Purchase
20 k	21	Y
60 k	27	Y
10 k	45	F
15 k	31	F
12 k	18	F

$$E(D_1) = -\frac{2}{5}\log_2\left(\frac{2}{5}\right) - \frac{3}{5}\log_2\frac{3}{5}$$

$$= 0.97$$

(*) We can use $\log_2$ & $\log_e = \ln$ can be used to calculate entropy.

Entropy Vs Probability Graph



* Information Gain:

→ Is a metric used to train decision trees. This metric measures the quality of a split.

→ The information gain is based on the decrease in entropy after a dataset is split on an attribute. Constructing a decision tree is all about finding attributes that returns the highest information gain.

* Decision tree steps

Outlook	Temperature	Humidity	Windy	PlayTennis
Sunny	Hot	High	False	No
Sunny	Hot	High	True	No
Overcast	Hot	High	False	Yes
Rainy	Mild	High	False	Yes
Rainy	Cool	Normal	False	Yes
Rainy	Cool	Normal	True	No
Overcast	Cool	Normal	True	Yes
Sunny	Mild	High	False	No
Sunny	Cool	Normal	False	Yes
Rainy	Mild	Normal	False	Yes
Sunny	Mild	Normal	True	Yes
Overcast	Mild	High	True	Yes
Overcast	Hot	Normal	False	Yes
Rainy	Mild	High	True	No

Step 1:

Entropy of Parent

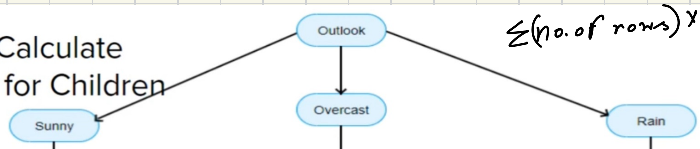
$E(P) = -p_y \log_2(p_y) - p_n \log_2(p_n)$

$= 9/14 \log_2(9/14) - 5/14/\log_2(5/14)$

$E(P) = \mathbf{0.94}$

Step 2: Calculate Entropy for Children

Σ(no. of rows) $\times$ Entropy



Day	Outlook	Humidity	Wind	Play
D1	Sunny	High	Weak	No
D2	Sunny	High	Strong	No
D8	Sunny	High	Weak	No
D9	Sunny	Normal	Weak	Yes
D11	Sunny	Normal	Strong	Yes

Day	Outlook	Humidity	Wind	Play
D3	Overcast	High	Weak	Yes
D7	Overcast	Normal	Strong	Yes
D12	Overcast	High	Strong	Yes
D13	Overcast	Normal	Weak	Yes

Day	Outlook	Humidity	Wind	Play
D4	Rain	High	Weak	Yes
D5	Rain	Normal	Weak	Yes
D6	Rain	Normal	Strong	No
D10	Rain	Normal	Weak	Yes
D14	Rain	High	Strong	No

E(S) =	E(O) =	E(R) =
-2/5log(2/5) - 3/5log(3/5)	-5/5log(5/5) - 0/5log(0/5)	-3/5log(3/5) - 2/5log(2/5)
E(S)= 0.97	E(O)= 0	E(S)= 0.97

↳ This is a leaf node because entropy is 0 here.

Step 3 : Calculate weighted Entropy of Children

Weighted Entropy = 5/14 * 0.97 + 4/14 * 0 + 5/14 * 0.97

W.E(Children) = **0.69**

Information Gain: Entropy of Parent − weighted average of entropy of children.

* Algorithm will split based on the most information gain.

Gini Impurity

→ A way to measure purity

$$\boxed{G_I = 1 - (P_y^2 - P_N^2)}$$

Information Gain = Parent Gini − Weighted average of child's gini.

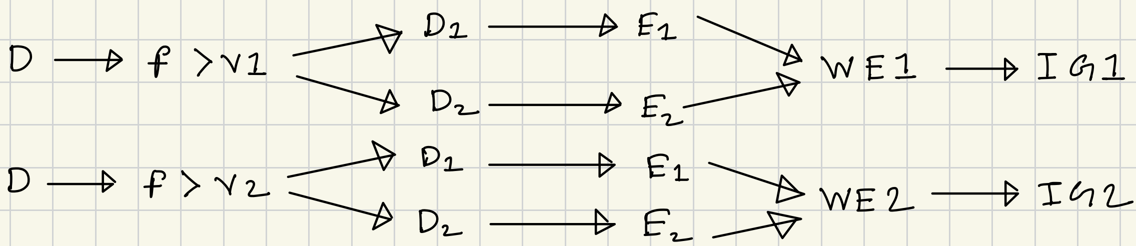
Maximum Gini can be 0.5

→ Gini is computationally faster.

* Handling numerical data

→ Split based on criteria.

Ex: Rating > 1.6



Find max information gain. such as IG2 in this case.

* How to decide which column should be considered as root node ?