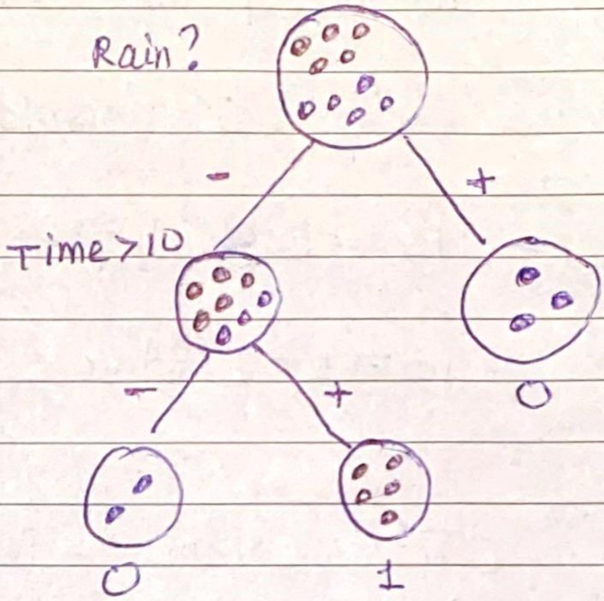
Implementing Decision Tree

Example: Person walking? or Taking a bus?

Rain	Time	Walk
1	30	No
1	15	No
1	5	No
0	10	No
0	5	No
0	15	yes
0	20	yes
0	25	yes
0	30	yes
0	30	yes



- Time they have.
- The only thing that we have to find out is at ~~what~~ which node we want to apply which question.
  - So why do we ask for rain at first & not for the time
  - & why do ask for time is greater than 10 &
  - why do not we as for if it's greater than 5 or greater than 20.

- The best split feature (value)(threshold)
  - that we want to find out
  - At each node we want to find the best split value

- Entropy

$$E = -\sum P(X) \cdot \log_2(P(X)) \qquad P(X) = \frac{\#x}{n}$$

Example: $S = [0, 0, 0, 0, 0, 1, 1, 1, 1, 1]$

$$E = -\frac{5}{10}\log_2\left(\frac{5}{10}\right) - \frac{5}{10}\log_2\left(\frac{5}{10}\right) = -0.5\log_2(0.5) - 0.5\log_2(0.5)$$

$$E = -0.5 \cdot (-1) - 0.5(-1) = 1$$

Information Gain

$IG = E(parent) - [weighted\ average] \cdot E(children)$

Example: $S = [0,0,0,0,0,1,1,1,1,1]$ $S1 = [0,0,1,1,1,1,1]$, $S2 = [0,0,0]$

$$IG = E(S) - [(\tfrac{7}{10}) \times E(S1) + (\tfrac{3}{10}) \times E(S2)]$$

$$IG = 1 - [(\tfrac{7}{10}) \times 0.863 + (\tfrac{3}{10}) \times 0] = 0.395$$

Approach

- Train algorithm := Build the tree

1. Start at the top node & at each node select the best split based on the best information gain.
2. Greedy search: Loop over all features & over all thresholds (all possible feature values).
3. Save the best split feature & split threshold at each node.
4. Build the tree recursively.
5. Apply some stopping criteria to stop growing
   e.g. here: maximum depth, minimum samples at node, no more class distribution in node.
6. When we have a leaf node, store the most common class label of this node.

- Predict := Traverse tree

7. Traverse the tree recursively.
8. At each node look at the best split feature of the test feature vector x & go left or right depending on x[feature_idx] <= threshold
9. When we reach the leaf node we return the stored most common class label.