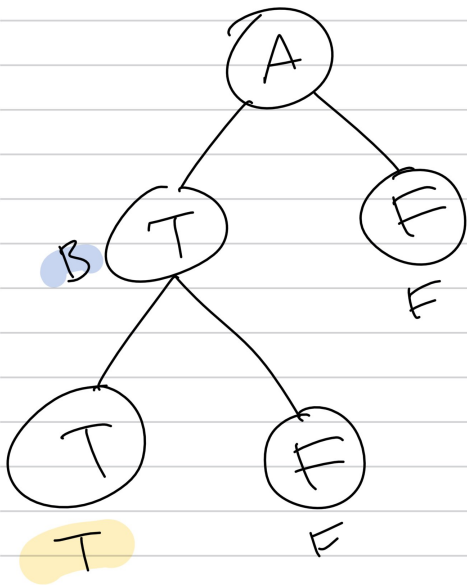


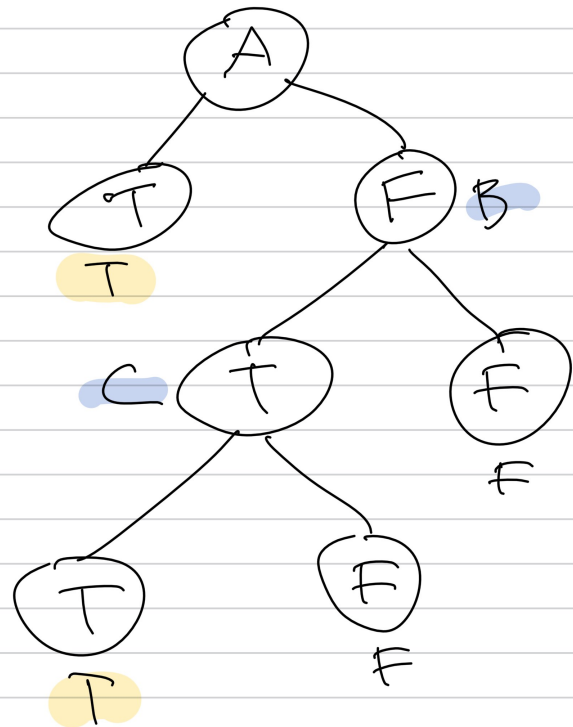
Assignment 1 - Andrew Chan

Question 1 - Decision trees

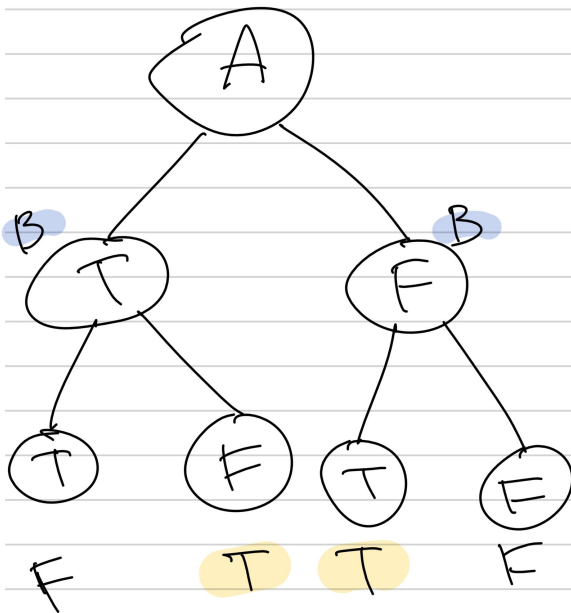
a) $A \wedge \neg B$



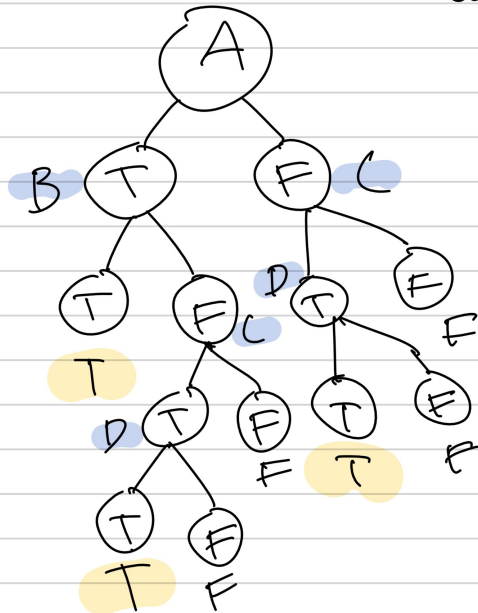
b) $A \vee (B \wedge C)$



c) $A \text{ XOR } B$



d) $(A \wedge B) \vee (C \wedge D)$



Part 1: Root node is: Outlook Branches are: Sunny, Overcast, Rain

Part 2: Sunny IDs: 1, 2, 8, 9, 11 Overcast IDs: 3, 7, 12, 13 Rain IDs: 4, 5, 6, 10, 14

Part 3: Sunny Branch IDs: 1, 2, 8, 9, 11

- PlayTennis = No => 3
- PlayTennis = Yes => 2

Entropy before split:

$$E(\text{Sunny}) = -\frac{2}{5} \log_2 \frac{2}{5} - \frac{3}{5} \log_2 \frac{3}{5} \approx 0.971$$

Gain(Humidity):

Humidity	Count	Yes	No
High	3	0	3
Normal	2	2	0

$$E(\text{Humidity}) = \frac{3}{5}(0) + \frac{2}{5}(0) = 0$$

$$\text{Gain}(\text{Sunny, Humidity}) = 0.971 - 0 = 0.971$$

Gain(Temperature):

Temperature	Count	Yes	No
Hot	2	0	2
Mild	2	1	1
Cool	1	1	0

- Entropies:
 - Hot: $E = 0$
 - Mild: $E = 1$
 - Cool: $E = 0$

$$E(\text{Temperature}) = \frac{2}{5}(0) + \frac{2}{5}(1) + \frac{1}{5}(0) = 0.4$$

$$\text{Gain}(\text{Sunny, Temperature}) = 0.971 - 0.4 = 0.571$$

Gain(Wind):

Wind	Count	Yes	No
Weak	3	1	2
Strong	2	1	1

- Entropies:

- Weak: $E \approx 0.918$
- Strong: $E = 1$

$$E(\text{Wind}) = \frac{3}{5}(0.918) + \frac{2}{5}(1) \approx 0.9508$$

$$\text{Gain}(\text{Sunny, Wind}) \approx 0.971 - 0.9508 = 0.0202$$

Best attribute for Sunny: Humidity (Gain = 0.971)

Overcast Branch IDs: 3, 7, 12, 13

- Since all labels == Yes, entropy = 0
- This means that there is no further splitting is necessary.

This is a pure leaf node.

Rain Branch IDs: 4, 5, 6, 10, 14

- PlayTennis = Yes => 3
- PlayTennis = No => 2

Entropy before split:

$$E(\text{Rain}) = -\frac{3}{5} \log_2 \frac{3}{5} - \frac{2}{5} \log_2 \frac{2}{5} \approx 0.971$$

Gain(Wind):

Wind	Count	Yes	No
Weak	3	3	0
Strong	2	0	2

$$E(\text{Wind}) = \frac{3}{5}(0) + \frac{2}{5}(0) = 0$$

$$\text{Gain}(\text{Rain, Wind}) = 0.971 - 0 = 0.971$$

Gain(Temperature):

Temperature	Count	Yes	No
Mild	3	2	1
Cool	2	1	1

- Entropies:
 - Mild: $E \approx 0.918$
 - Cool: $E = 1$

$$E(\text{Temperature}) = \frac{3}{5}(0.918) + \frac{2}{5}(1) \approx 0.9508$$

$$\text{Gain}(\text{Rain, Temperature}) = 0.971 - 0.9508 = 0.0202$$

Gain(Humidity):

Humidity	Count	Yes	No
High	2	1	1
Normal	3	2	1

- Entropies:
 - High: $E = 1$
 - Normal: $E \approx 0.918$

$$E(\text{Humidity}) = \frac{2}{5}(1) + \frac{3}{5}(0.918) \approx 0.9508$$

$$\text{Gain}(\text{Rain, Humidity}) = 0.971 - 0.9508 = 0.0202$$

Best attribute for Rain: Wind (Gain = 0.971)

Final Subtree Decisions:

- Sunny Branch: split on **Humidity**
- Overcast Branch: pure leaf, **no split**
- Rain Branch: split on **Wind**

Question 3 Combining Trees

The bank uses two separate decision trees for loan applicants:

- Tree 1: Credit history
- Tree 2: Demographics

Each tree outputs either:

- High Risk
- Low Risk

The bank only offers a loan when **both trees predict "Low Risk"**.

Part 1: Algorithm to Combine the Two Trees

First we need to create a single decision tree that has the same results as if we were to combine the outputs of the original two trees. This would mean that it should only return "Low Risk" if both Tree 1 and Tree 2 are "Low Risk". Otherwise, we would return "High Risk". This is basically the same as:

$$\text{Output} = \text{Tree}_1 \wedge \text{Tree}_2$$

The algorithm's pseudo code would look like this:

- For each leaf node in Tree 1:
 - For each leaf node in Tree 2:
 - Create a path in the combined tree by combining the pre-existing conditions from both Tree 1 and Tree 2.

- Set the results for the leaf:
 - If both trees are "Low Risk" the leaf will be "Low Risk"
 - Else, the leaf is considered "High Risk"

This allows all possible pairs of paths from both Tree 1 and Tree 2 to be mapped into a single tree. It's like nesting Tree 2 inside each leaf of Tree 1 (or vice versa).

Part 2: Upper Bound on Number of Leaves

Let:

- n_1 = number of leaves in Tree 1
- n_2 = number of leaves in Tree 2

All of the leaves in Tree 1 can be combined with all leaves in Tree 2. That means that in the worst-case (no pruning possible), the number of leaves in the combined tree is:

$$n \leq n_1 \times n_2$$

Therefore, the number of leaves in the combined tree is at most:

$$n_1 \times n_2$$

For example, if Tree 1 has 5 leaves and Tree 2 has 7 leaves:

- The combined tree could have up to $5 \times 7 = 35$ leaves.
- Each leaf would be a unique combination (Tree 1 decision \times Tree 2 decision).