**Steps to Build a Decision Tree**

1. **Select the Best Attribute**:
   * Choose the attribute that best separates the data into distinct classes. This is usually done using criteria such as Information Gain, Gini Index, or Chi-Square.
2. **Split the Dataset**:
   * Split the dataset into subsets based on the selected attribute. Each subset should contain data points that share the same value for the chosen attribute.
3. **Create a Decision Node**:
   * A decision node is created in the tree, representing the chosen attribute and the branches to its subsets.
4. **Repeat for Each Subset**:
   * For each subset created, repeat the process: select the best attribute, split the dataset, and create decision nodes.
5. **Stop the Process**:
   * The process stops when one of the following conditions is met:
     + All data points in a subset belong to the same class.
     + There are no more attributes to split on.
     + The subset is empty.

**Manual Calculation Example**

Consider the following simple dataset of weather conditions and the decision to play tennis:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Outlook** | **Temperature** | **Humidity** | **Wind** | **Play Tennis** |
| Sunny | Hot | High | Weak | No |
| Sunny | Hot | High | Strong | No |
| Overcast | Hot | High | Weak | Yes |
| Rain | Mild | High | Weak | Yes |
| Rain | Cool | Normal | Weak | Yes |
| Rain | Cool | Normal | Strong | No |
| Overcast | Cool | Normal | Strong | Yes |
| Sunny | Mild | High | Weak | No |
| Sunny | Cool | Normal | Weak | Yes |
| Rain | Mild | Normal | Weak | Yes |
| Sunny | Mild | Normal | Strong | Yes |
| Overcast | Mild | High | Strong | Yes |
| Overcast | Hot | Normal | Weak | Yes |
| Rain | Mild | High | Strong | No |

**Step-by-Step Calculation**

1. **Calculate the Entropy of the Entire Dataset**:
   * Entropy (S) = - p(yes) log2(p(yes)) - p(no) log2(p(no))
   * p(yes) = 9/14, p(no) = 5/14
   * Entropy(S) = - (9/14) log2(9/14) - (5/14) log2(5/14) ≈ 0.94
2. **Calculate the Entropy for Each Attribute**:
   * For each attribute, calculate the entropy for each subset and the weighted average entropy after the split.
3. **Information Gain for Each Attribute**:
   * Information Gain = Entropy(S) - Weighted Entropy (after split)

Let's calculate for the Outlook attribute:

**Outlook = Sunny**:

* Subset: 5 instances (3 No, 2 Yes)
* Entropy(Sunny) = - (3/5) log2(3/5) - (2/5) log2(2/5) ≈ 0.97

**Outlook = Overcast**:

* Subset: 4 instances (4 Yes)
* Entropy(Overcast) = 0 (since all are Yes)

**Outlook = Rain**:

* Subset: 5 instances (2 No, 3 Yes)
* Entropy(Rain) = - (2/5) log2(2/5) - (3/5) log2(3/5) ≈ 0.97

**Weighted Entropy for Outlook**:

* Weighted Entropy = (5/14) \* 0.97 + (4/14) \* 0 + (5/14) \* 0.97 ≈ 0.693

**Information Gain for Outlook**:

* Information Gain = 0.94 - 0.693 ≈ 0.247

Repeat this for other attributes: Temperature, Humidity, and Wind.

1. **Select the Attribute with the Highest Information Gain**:
   * Suppose after calculation, Outlook has the highest information gain. This becomes the root node.
2. **Split and Repeat**:
   * Split the dataset based on Outlook and repeat the process for each subset (Sunny, Overcast, Rain) until you reach stopping conditions.

**First Level Split: Outlook**

The Outlook attribute was chosen as the root node based on the highest information gain.

markdown

Copy code

Outlook

/ | \

Sunny Overcast Rain

/ | \

**Second Level Split**

We need to calculate the best splits for the subsets created by the Outlook attribute.

**For Outlook = Sunny:**

Subset:

| **Outlook** | **Temperature** | **Humidity** | **Wind** | **Play Tennis** |
| --- | --- | --- | --- | --- |
| Sunny | Hot | High | Weak | No |
| Sunny | Hot | High | Strong | No |
| Sunny | Mild | High | Weak | No |
| Sunny | Cool | Normal | Weak | Yes |
| Sunny | Mild | Normal | Strong | Yes |

**Entropy Calculation for Sunny Subset:**

* Entropy(Sunny) = - (3/5) log2(3/5) - (2/5) log2(2/5) ≈ 0.97

**Calculate Information Gain for each attribute in Sunny subset:**

1. **Temperature:**
   * Hot: 2 No, 0 Yes (Entropy = 0)
   * Mild: 1 No, 1 Yes (Entropy ≈ 1)
   * Cool: 0 No, 1 Yes (Entropy = 0)

Weighted Entropy for Temperature = (2/5)\*0 + (2/5)\*1 + (1/5)\*0 = 0.4

Information Gain = 0.97 - 0.4 = 0.57

1. **Humidity:**
   * High: 3 No, 0 Yes (Entropy = 0)
   * Normal: 0 No, 2 Yes (Entropy = 0)

Weighted Entropy for Humidity = (3/5)\*0 + (2/5)\*0 = 0

Information Gain = 0.97 - 0 = 0.97

1. **Wind:**
   * Weak: 2 No, 1 Yes (Entropy ≈ 0.918)
   * Strong: 1 No, 1 Yes (Entropy ≈ 1)

Weighted Entropy for Wind = (3/5)\*0.918 + (2/5)\*1 ≈ 0.950

Information Gain = 0.97 - 0.950 = 0.02

The highest information gain is from Humidity. Therefore, we split Sunny on Humidity.

Outlook

/ | \

Sunny Overcast Rain

/ | \

Humidity Yes ...

/ \

High Normal

/ \

No Yes

**For Outlook = Rain:**

Subset:

| **Outlook** | **Temperature** | **Humidity** | **Wind** | **Play Tennis** |
| --- | --- | --- | --- | --- |
| Rain | Mild | High | Weak | Yes |
| Rain | Cool | Normal | Weak | Yes |
| Rain | Cool | Normal | Strong | No |
| Rain | Mild | Normal | Weak | Yes |
| Rain | Mild | High | Strong | No |

**Entropy Calculation for Rain Subset:**

* Entropy(Rain) = - (2/5) log2(2/5) - (3/5) log2(3/5) ≈ 0.97

**Calculate Information Gain for each attribute in Rain subset:**

1. **Temperature:**
   * Mild: 2 No, 1 Yes (Entropy ≈ 0.918)
   * Cool: 1 No, 2 Yes (Entropy ≈ 0.918)

Weighted Entropy for Temperature = (3/5)\*0.918 + (2/5)\*0.918 ≈ 0.918

Information Gain = 0.97 - 0.918 = 0.052

1. **Humidity:**
   * High: 2 No, 0 Yes (Entropy = 0)
   * Normal: 1 No, 2 Yes (Entropy ≈ 0.918)

Weighted Entropy for Humidity = (2/5)\*0 + (3/5)\*0.918 ≈ 0.5508

Information Gain = 0.97 - 0.5508 = 0.4192

1. **Wind:**
   * Weak: 0 No, 3 Yes (Entropy = 0)
   * Strong: 2 No, 0 Yes (Entropy = 0)

Weighted Entropy for Wind = (3/5)\*0 + (2/5)\*0 = 0

Information Gain = 0.97 - 0 = 0.97

The highest information gain is from Wind. Therefore, we split Rain on Wind.

Outlook

/ | \

Sunny Overcast Rain

/ | \

Humidity Yes Wind

/ \ / \

High Normal Weak Strong

**Third Level Split**

We need to explore further splits for any nodes that are not pure yet.

**For Outlook = Sunny and Humidity = High:**

Already pure with the decision: No

**For Outlook = Sunny and Humidity = Normal:**

Already pure with the decision: Yes

**For Outlook = Rain and Wind = Weak:**

Already pure with the decision: Yes

**For Outlook = Rain and Wind = Strong:**

Already pure with the decision: No

**For Outlook = Overcast:**

Already pure with the decision: Yes

Given the dataset and splits we used, the decision tree is already fully pure with the decisions at the second level. Therefore, no third-level splits are required for this specific example as each path from the root to a leaf node ends in a pure subset (all samples in a subset belong to a single class).

**Final Decision Tree (No Additional Splits Required):**

Outlook

/ | \

Sunny Overcast Rain

/ | \

Humidity Yes Wind

/ \ / \

High Normal Weak Strong

/ \ \ \

No Yes Yes No

This tree correctly classifies the given dataset based on the attribute splits, ensuring that each leaf node represents a pure subset. If you have further questions or need additional details, feel free to ask!