Decision Trees in Machine Learning

Prepared by [Your Name]

[Your Institution]

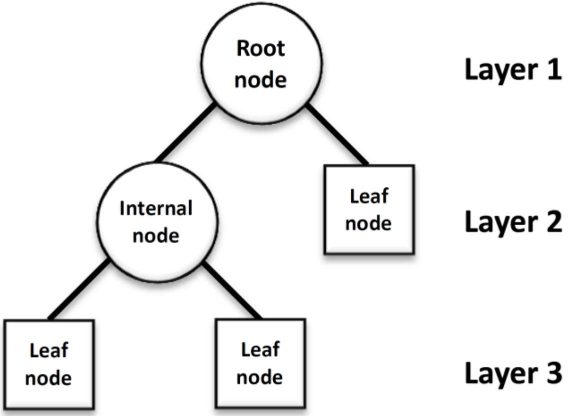
January 25, 2025

Introduction to Decision Trees

- A supervised learning algorithm used for classification and regression.
- Splits data into branches based on feature values.
- Popular due to simplicity, interpretability, and flexibility.

Structure of a Decision Tree

- **Root Node:** Represents the entire dataset.
- **Internal Nodes (attribute):** Decision points based on features.
- **Leaf Nodes:** Outcomes or predictions.



How Decision Trees Work

1. **Splitting:** Data is divided based on features.
2. **Feature Selection:**
    - Gini Impurity
    - Entropy and Information Gain
    - Variance Reduction
3. **Stopping Criteria:**
    - Maximum depth
    - Minimum samples per leaf

Decision Tree Example: Should I Accept a New Job Offer?

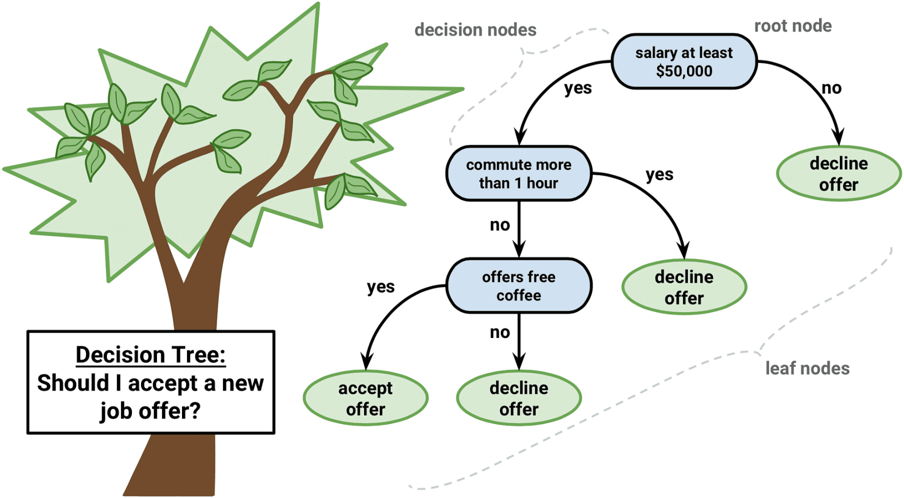
Overview:

- This decision tree evaluates three factors to determine whether to accept a job offer:
  - Salary
  - Commute Time
  - Workplace Perks
- The goal is to make an informed decision based on these criteria.

Decision Flow:

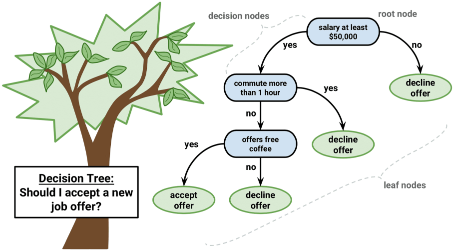
- Start with the root node (salary evaluation).
- Follow branches based on the answers.
- Arrive at a leaf node that provides the final decision (accept or decline).

Decision Tree Algorithm



Decision Tree Visualization

Tree Representation:



You can find more about it here.

Explanation of Nodes:

- **Root Node:** Salary at least $50,000.
- **Decision Nodes:**
  - Commute time more than 1 hour.
  - Offers free coffee.
- **Leaf Nodes:** Accept or decline the job offer.

Example: Classification

Dataset:

Day	Outlook	Temperature	Humidity	Wind	Play?
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	No
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

Step 1: Create a root node

- **How to chose the root node?**

Attribute that best classifies the training data, use this attribute at the root of the tree.

- **How to chose the best attribute?**

ID3 Algoritm

Step 1: Create a root node (continue)

- Calculate **Entropy** (Amount of uncertainty in dataset)

Entropy = (-p/(p+n))log2(P/(p+n))-(n/(p+n))log2(n/(p+n)

- Calculate **Average information**

I(attribute) = sum((pi+ni)/(p+n))Entropy(A)

- Calculate **Information Gain** (Difference in Entropy before and after splitting dataset on attribute A)

Gain = Entropy(S)-I(atribute)

Step 1: Create a root node (continue)

- Calculate **Entropy** (Amount of uncertainty in dataset)

$$\text{Entropy} = -\left(\frac{p}{p+n}\right)\log_2\left(\frac{p}{p+n}\right) - \left(\frac{n}{p+n}\right)\log_2\left(\frac{n}{p+n}\right)$$

where p is the number of positive examples and n is the number of negative examples.

- Calculate **Average information**

$$I(\text{attribute}) = \sum_{v \in \text{Values(attribute)}} \left(\frac{p_v + n_v}{p+n}\right) \text{Entropy}(A_v)$$

where p_v and n_v are the number of positive and negative examples in the subset corresponding to value v, and $\text{Entropy}(A_v)$ is the entropy for that subset.

- Calculate **Information Gain** (Difference in Entropy before and after splitting dataset on attribute A)

$$\text{Gain} = \text{Entropy}(S) - I(\text{attribute})$$

where $\text{Entropy}(S)$ is the entropy of the original dataset and $I(\text{Attribute})$ is the average information after the split.

1. Calculate Entropy

P=:9, N=:5 and Total=:14

Day	Outlook	Temperature	Humidity	Wind	Play?
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	No
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

1. Calculate Entropy

$$\textbf{Entropy} = -\left(\frac{p}{p+n}\right)\log_2\left(\frac{p}{p+n}\right) - \left(\frac{n}{p+n}\right)\log_2\left(\frac{n}{p+n}\right)$$

$$\textbf{Entropy(S)} = -\left(\frac{9}{9+5}\right)\log_2\left(\frac{9}{9+5}\right) - \left(\frac{5}{9+5}\right)\log_2\left(\frac{5}{9+5}\right)$$

$$\textbf{Entropy(S)} = -\left(\frac{9}{14}\right)\log_2\left(\frac{9}{14}\right) - \left(\frac{5}{14}\right)\log_2\left(\frac{5}{14}\right) = 0.940$$

2. Average information

For each attribute: (let's say **Outlook**)

- Calculate Entropy for each value, i.e., for 'Sunny', 'Rainy', and 'Overcast'

Outlook	**Play Tennis?**
Sunny	No
Sunny	No
Sunny	No
Sunny	Yes
Sunny	No

Table 1

Outlook	**Play Tennis?**
Sunny	No
Sunny	No
Sunny	No
Sunny	Yes
Sunny	No

Table 2

Outlook	**Play Tennis?**
Sunny	No
Sunny	No
Sunny	No
Sunny	Yes
Sunny	No

Table 3

Outlook	**P**	**N**	**Entropy**
Sunny	2	3	0.971
Rainy	3	2	0.971
Overcast	4	0	0

Table: Table 3

Calculate Entropy(Outlook=value)

$$\textbf{Entropy} = -\left(\frac{p}{p+n}\right)\log_2\left(\frac{p}{p+n}\right) - \left(\frac{n}{p+n}\right)\log_2\left(\frac{n}{p+n}\right)$$

$$\textbf{Entropy(Sunny)} = -\left(\frac{2}{5}\right)\log_2\left(\frac{2}{5}\right) - \left(\frac{3}{5}\right)\log_2\left(\frac{3}{5}\right) = 0.971$$

$$\textbf{Entropy(Rainy)} = -\left(\frac{3}{5}\right)\log_2\left(\frac{3}{5}\right) - \left(\frac{2}{5}\right)\log_2\left(\frac{2}{5}\right) = 0.971$$

$$\textbf{Entropy(Overcast)} = -\left(\frac{4}{4}\right)\log_2\left(\frac{4}{4}\right) - \left(\frac{0}{4}\right)\log_2\left(\frac{0}{4}\right) = 0$$

Calculate Average Information Entropy

$$I(\text{Outlook}) = \sum_{v \in \text{Values}(\text{Outlook})} \left(\frac{p_v + n_v}{p + n} \right) \text{Entropy}(A_v)$$

$$I(\text{Outlook}) = \left(\frac{p_{\text{sunny}} + n_{\text{sunny}}}{p + n} \right) \text{Entropy}(\text{Outlook} = \text{Sunny}) +$$
$$\left(\frac{p_{\text{rainy}} + n_{\text{rainy}}}{p + n} \right) \text{Entropy}(\text{Outlook} = \text{Rainy}) +$$
$$\left(\frac{p_{\text{overcast}} + n_{\text{overcast}}}{p + n} \right) \text{Entropy}(\text{Outlook} = \text{Overcast})$$

$$I(\text{Outlook}) = \left(\frac{3+2}{9+5} \right) * 0.971 + \left(\frac{3+2}{9+5} \right) * 0.971 + \left(\frac{3+2}{9+5} \right) * 0.971 = 0.693$$

Calculate Gain : attribute is outlook

$$\text{Gain} = \text{Entropy}(S) - I(\text{attribute})$$

$$\text{Entropy}(S) = 0.940$$

$$\text{Gain}(\text{Outlook}) = 0.940 - 0.693 = 0.247$$

Advantages and Disadvantages

Advantages:

- Easy to interpret and explain.
- Handles categorical and numerical data.
- No assumptions about data distribution.

Disadvantages:

- Prone to overfitting with deep trees.
- Sensitive to data imbalance.
- Greedy algorithm may miss global optima.

Pruning and Ensemble Methods

- **Pruning:**
  - Pre-pruning: Stop splits early.
  - Post-pruning: Remove unnecessary branches.
- **Ensemble Methods:**
  - Random Forest
  - Gradient Boosting

Scikit-Learn Implementation

Python Code Example:

Python Code

```
from sklearn.tree import DecisionTreeClassifier
X = [[1, 0], [0, 1], [1, 1], [0, 0]]
y = [1, 0, 1, 0]
clf = DecisionTreeClassifier()
clf.fit(X, y)
print(clf.predict([[1, 1]]))
```

Visualization: Use `plot_tree`.

Conclusion

- Decision trees are a powerful and interpretable tool.
- Best suited for datasets with clear feature splits.
- Use ensemble methods to overcome limitations like overfitting.

Thank You!