MACHINE LEARNING

Decision Tree Classifier - Multi-Dataset Analysis

Name := Nagula Anish

SRN := PES2UG23CS358

Section := F

---

1. Performance Comparison

Dataset	Accuracy	Precision (weighted)	Precision (macro)	Recall (weighted)	Recall (macro)	F1-Score (macro)	F1-Score (weighte
Mushrooms	1.000 (100%)	1.000	1.000	1.000	1.000	1.000	1.000
Nursery	0.9867 (98.67%)	0.9867	0.7604	0.9867	0.7654	0.9872	0.7628
Tic-Tac-Toe	0.8730 (87.30%)	0.8741	0.8590	0.8730	0.8638	0.8734	0.8613

- The **Mushrooms** dataset achieved a **perfect 100%** across all metrics, showing that it is a well-fit and clean dataset. The model had made all correct predictions which meant it found all the poisonous mushrooms.
- **The Nursery** dataset also performed very well, with close to 99% accuracy. This tells us that the dataset is almost perfectly-fit and clean and has a little bit of noise present in it. Most of the predictions made were correct but has made a few miss-predictions.
- The **Tic-Tac-Toe** dataset was the most challenging, with an accuracy of approximately 88%. This tells us that the dataset is not clean as the other two and that its accuracy can be improved.

---

2. Tree Characteristics Analysis

Dataset	Max Depth	Total Nodes	**Leaf Nodes**	Internal Nodes
Mushrooms	4	29	24	5
Nursery	7	952	680	272
Tic-Tac-Toe	7	281	180	101

- Most Important Features:
  - For the **Mushrooms** dataset, the `odor` attribute is the most crucial for classification.
  - In the **Nursery** dataset, `finance`, `social`, and `health` features dominate the early splits.
  - For **Tic-Tac-Toe**, the `center` and `corner` squares are the most important features.
- Tree Complexity:
  - The **Mushrooms** tree is the simplest and most shallow, reflecting the clear separation of its classes by a few strong features.
  - The **Nursery** tree is the largest and most complex, as it needs to manage many possible outcomes.

- The **Tic-Tac-Toe** tree is also quite large, suggesting the need to capture complex patterns to make accurate predictions.

---

3. Dataset-Specific Insights

- Mushrooms :=
  - **Feature Importance:** The **odor** attribute is the most decisive factor.
  - **Class Distribution:** The classes are balanced and perfectly separated by the features.
  - **Decision Patterns:** The decision rules are very short, such as "odor=foul → poisonous".
  - **Overfitting Indicators:** There are no signs of overfitting, as the tree is small and still achieves 100% accuracy.
- Nursery :=
  - **Feature Importance:** `Finance` **,** `social` **, and** `health` features are the most dominant.
  - **Class Distribution:** The data is skewed, which weakens the model's performance on smaller classes, hence is unbalanced.
  - **Decision Patterns:** The tree creates clear rules like "poor finance → not_recommended" unless other factors are present.
  - **Overfitting Indicators:** The large number of nodes indicates a complex tree, but it generalizes well with high accuracy.
- Tic-Tac-Toe :=
  - **Feature Importance:** The `center` and `corners` of the board are the most important for decision-making.
  - **Class Distribution:** The "negative" class has more examples than the "positive" class, causing the model to struggle with the minority class.
  - **Decision Patterns:** The decision paths are long and specific, with many nodes.
  - **Overfitting Indicators:** The large tree size suggests it could benefit from pruning or depth limits to prevent overfitting.

---

4. Comparative Analysis Report

- **Algorithm Performance** :=
  a. The Mushroom dataset had the highest accuracy. The odor feature made it easy to classify.
  b. Larger dataset increases the number of test cases to be run and hence the accuracy improves, where as the smaller datasets may result in lower accuracy.
  c. Feature quality is more important than quantity. Good features help distinguish between the classes.
- **Data Characteristics Impact** :=
  a. Class imbalance decreases the overall F1 score, for example in nursery weighted values show good scores whereas macro scores are poor.
  b. The algorithm works well with multi-valued features. It performs best when feature values predict the class.
- **Real World Scenarios** :=
  a. Mushrooms → Good for clear yes/no problems (food safety, spam).
  b. Nursery → Models complex decisions (university admissions).
  c. Tic-Tac-Toe → Represents rule-based systems (game AI, network security).

d. How to Improve :=
    a. Mushroom: No changes needed. It works perfectly.
    b. Nursery: Fix the class imbalance and then, use a stronger algorithm.
    c. Tic-Tac-Toe: Use a different algorithm for a slightly better score.