Lab 3 Analysis Report

> Implement the ID3 Decision Tree algorithm and perform comparative analysis across three diverse datasets to understand algorithm performance under different data characteristics.

Performance Comparison

Dataset	Accuracy	Precision	Recall	F1-Score
tictactoe.csv	0.883	0.882	0.883	0.882
nursery.csv	0.988	0.988	0.988	0.988
mushrooms.csv	1.000	1.000	1.000	1.000

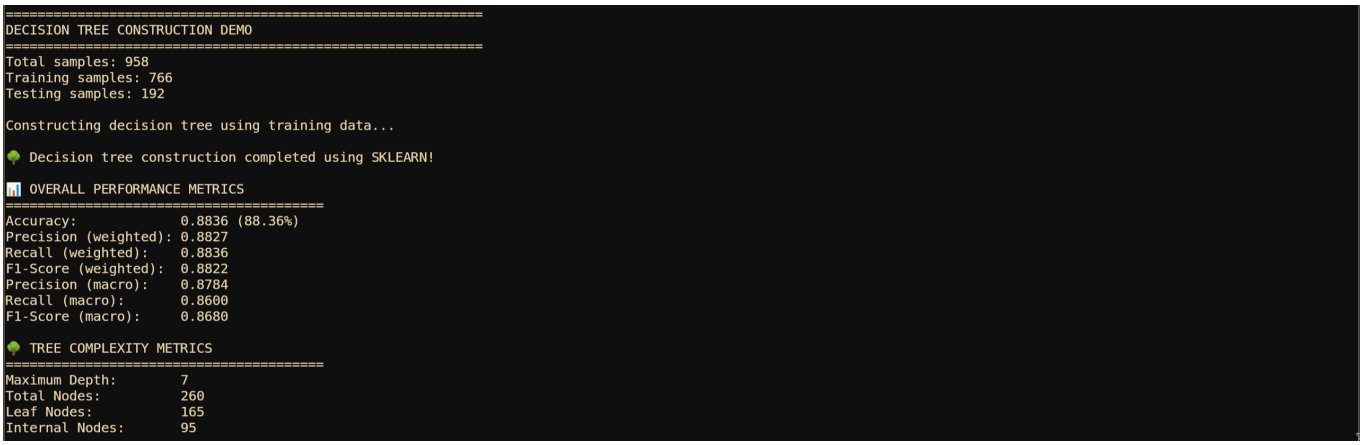
Mushrooms.csv

```
============================================================
DECISION TREE CONSTRUCTION DEMO
============================================================
Total samples: 8124
Training samples: 6499
Testing samples: 1625

Constructing decision tree using training data...

🌳 Decision tree construction completed using SKLEARN!

📊 OVERALL PERFORMANCE METRICS
========================================
Accuracy:              1.0000 (100.00%)
Precision (weighted): 1.0000
Recall (weighted):    1.0000
F1-Score (weighted):  1.0000
Precision (macro):    1.0000
Recall (macro):       1.0000
F1-Score (macro):     1.0000

🌳 TREE COMPLEXITY METRICS
========================================
Maximum Depth:         4
Total Nodes:           29
Leaf Nodes:            24
Internal Nodes:        5
```

Nursery.csv

```
============================================================
DECISION TREE CONSTRUCTION DEMO
============================================================
Total samples: 12960
Training samples: 10368
Testing samples: 2592

Constructing decision tree using training data...

🌳 Decision tree construction completed using SKLEARN!

📊 OVERALL PERFORMANCE METRICS
========================================
Accuracy:              0.9887 (98.87%)
Precision (weighted): 0.9888
Recall (weighted):    0.9887
F1-Score (weighted):  0.9887
Precision (macro):    0.9577
Recall (macro):       0.9576
F1-Score (macro):     0.9576

🌳 TREE COMPLEXITY METRICS
========================================
Maximum Depth:         7
Total Nodes:           983
Leaf Nodes:            703
Internal Nodes:        280
```

TicTacToe.csv



Tree Characteristics

Dataset	Depth	Number. Nodes	Imp Features	Complexity
tictactoe.csv	7	260	Center positions	Medium
nursery.csv	7	983	Health, social, Finance	High
mushrooms.csv	4	29	Odor, Spore print	Low

Dataset Specific Insights

Dataset	Feature Importance	Class Distribution	Decision Patterns	Overfitting?
tictactoe.csv	Health	Balanced	If the middle square is taken then the tree checks other squares for winning moves.	Likely, due to large number of nodes for relatively small dataset.
nursery.csv	has_nurs	Imbalanced	If Health of student in "not recommended" then the student is immediately rejected.	Likely, due to large number of nodes (983).
mushrooms.csv	Odor	Balanced	If mushroom has certain types of odor (1, 2, 4, 6, 7, 8) then mushroom is immediately classified as poisonous.	Not likely.

Comparative Analysis Report

Algorithm Performance

Highest Accuracy

Mushroom dataset achieved highest accuracy, this is because of features like odor, spore print, which were highly directly correlated with the class labels and had high information gain.

Dataset Size

Dataset size did not directly influence accuracy and other performance metrics, the information gain of features influences these metrics, dataset size did influence complexity of decision tree, which the larger datasets yielding deeper and more complex decision trees.

Number of Features

Number of features can lead to more branches in the decision tree and may lead to better accuracy, in this case more features did not lead to better performance.

Data Characteristics Impact

How does class imbalance affect tree construction?

Decision trees tend to favor majority classes (due to high information gain), therefore if class labels are not well balanced and some are in the minority, then the decision tree may struggle to consider these minority cases.

Feature Types

Binary features are generally better for the ID3 algorithm, as it is known to have a bias towards multi-valued features.

Practical Applications

Which real world scenarios is the dataset type most relevant

1. Mushrooms: For binary classification problems
2. Nursery: Real world applications like school / education systems
3. TicTacToe: Machine learning for games / game bots

What improvements can be made to performance for each dataset

1. Mushrooms: Already pretty good.
2. Nursery: We have to deal with class imbalance, approaches like weighted classes can be used
3. Tic-Tac-Toe: Pruning of decision tree can be done to combat over fitting.