# UE23CS352A: Machine Learning

## LAB 3: Decision Tree Classifier - Multi-Dataset Analysis

NAME: BOJJA RAKSHITHA

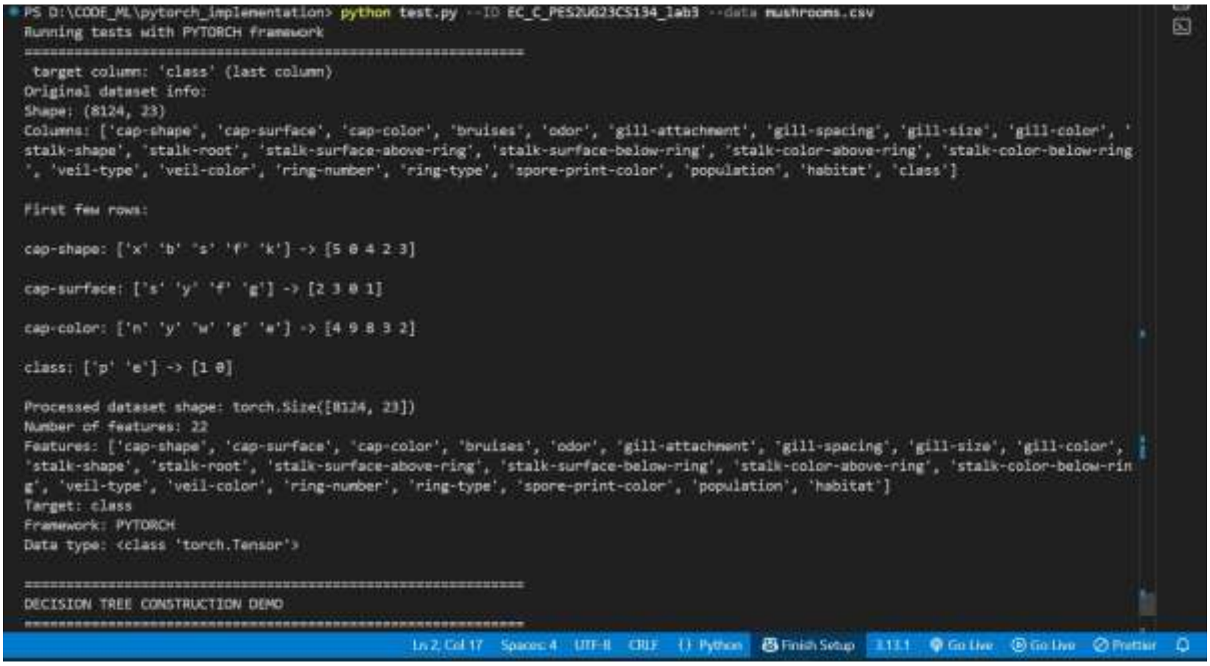
SECTION:C

SRN:PES2UG23CS134

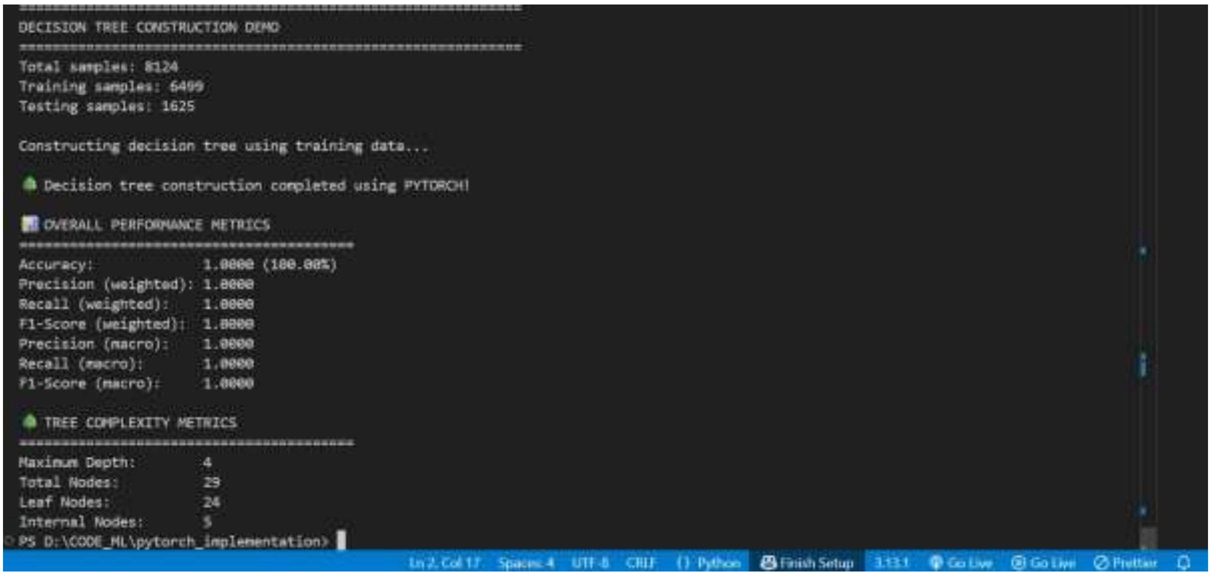
SCREENSHOTS:

Dataset 1: Mushroom Classification

BASIC TESTING



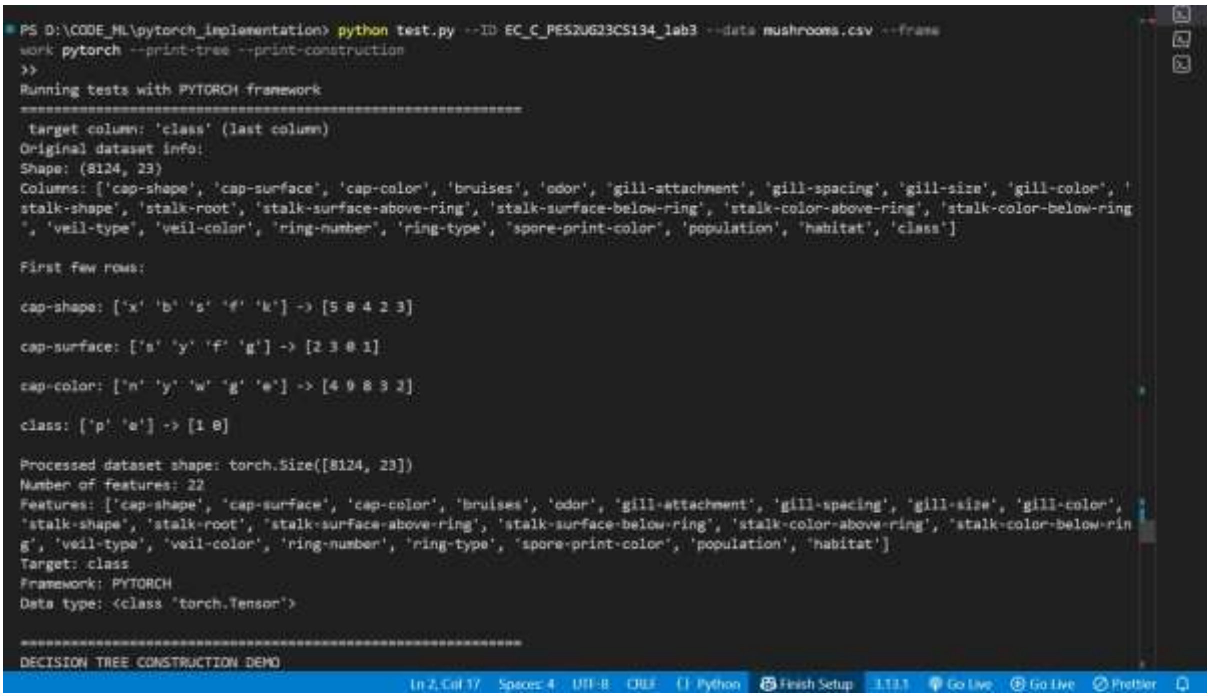
```
PS D:\CODE_ML\pytorch_implementation> python test.py --ID EC_C_PES2UG23CS134_lab3 --data mushrooms.csv
Running tests with PYTORCH framework
================================================================
 target column: 'class' (last column)
Original dataset info:
Shape: (8124, 23)
Columns: ['cap-shape', 'cap-surface', 'cap-color', 'bruises', 'odor', 'gill-attachment', 'gill-spacing', 'gill-size', 'gill-color', '
stalk-shape', 'stalk-root', 'stalk-surface-above-ring', 'stalk-surface-below-ring', 'stalk-color-above-ring', 'stalk-color-below-ring
', 'veil-type', 'veil-color', 'ring-number', 'ring-type', 'spore-print-color', 'population', 'habitat', 'class']

First few rows:

cap-shape: ['x' 'b' 's' 'f' 'k'] -> [5 0 4 2 3]

cap-surface: ['s' 'y' 'f' 'g'] -> [2 3 0 1]

cap-color: ['n' 'y' 'w' 'g' 'e'] -> [4 9 8 3 2]

class: ['p' 'e'] -> [1 0]

Processed dataset shape: torch.Size([8124, 23])
Number of features: 22
Features: ['cap-shape', 'cap-surface', 'cap-color', 'bruises', 'odor', 'gill-attachment', 'gill-spacing', 'gill-size', 'gill-color',
'stalk-shape', 'stalk-root', 'stalk-surface-above-ring', 'stalk-surface-below-ring', 'stalk-color-above-ring', 'stalk-color-below-rin
g', 'veil-type', 'veil-color', 'ring-number', 'ring-type', 'spore-print-color', 'population', 'habitat']
Target: class
Framework: PYTORCH
Data type: <class 'torch.Tensor'>

================================================================
DECISION TREE CONSTRUCTION DEMO
================================================================
```



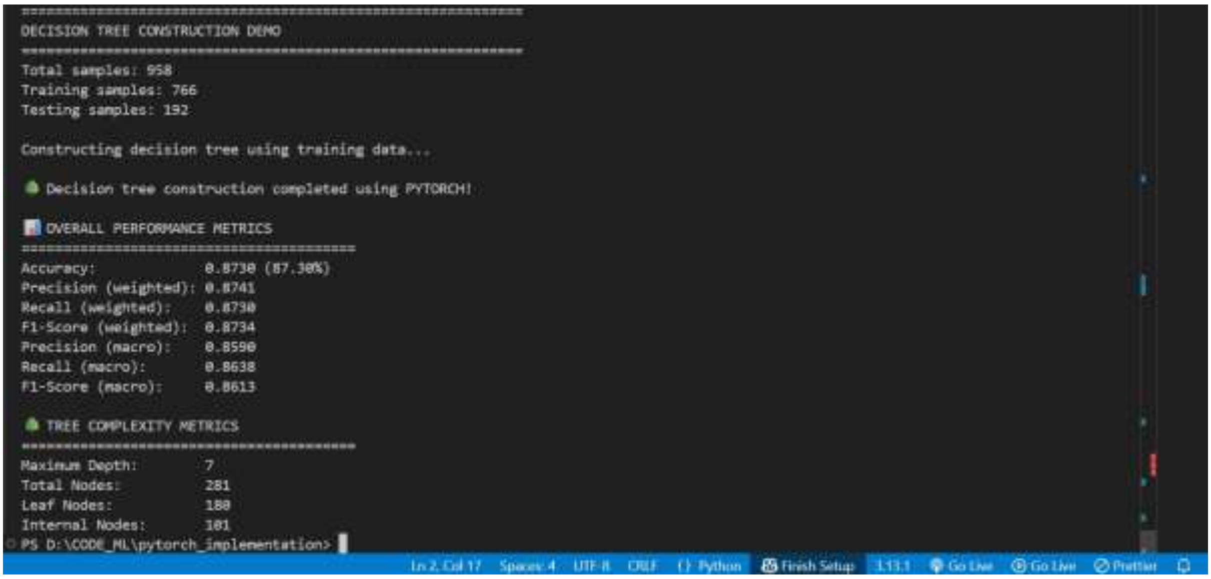
```
============================================================
DECISION TREE CONSTRUCTION DEMO
============================================================
Total samples: 8124
Training samples: 6499
Testing samples: 1625

Constructing decision tree using training data...

🌲 Decision tree construction completed using PYTORCH!

📊 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
PS D:\CODE_ML\pytorch_implementation> 
```

- **Highest Accuracy:** Achieved 100% accuracy using PyTorch and Sklearn.
- Binary classification with highly informative features like odor, which had high information gain and low entropy.
- **Dataset Size Impact:** 8124 samples allowed for good generalization with a shallow tree (Max Depth: 4).
- **Number of Features:** 22 features helped the model make precise splits.

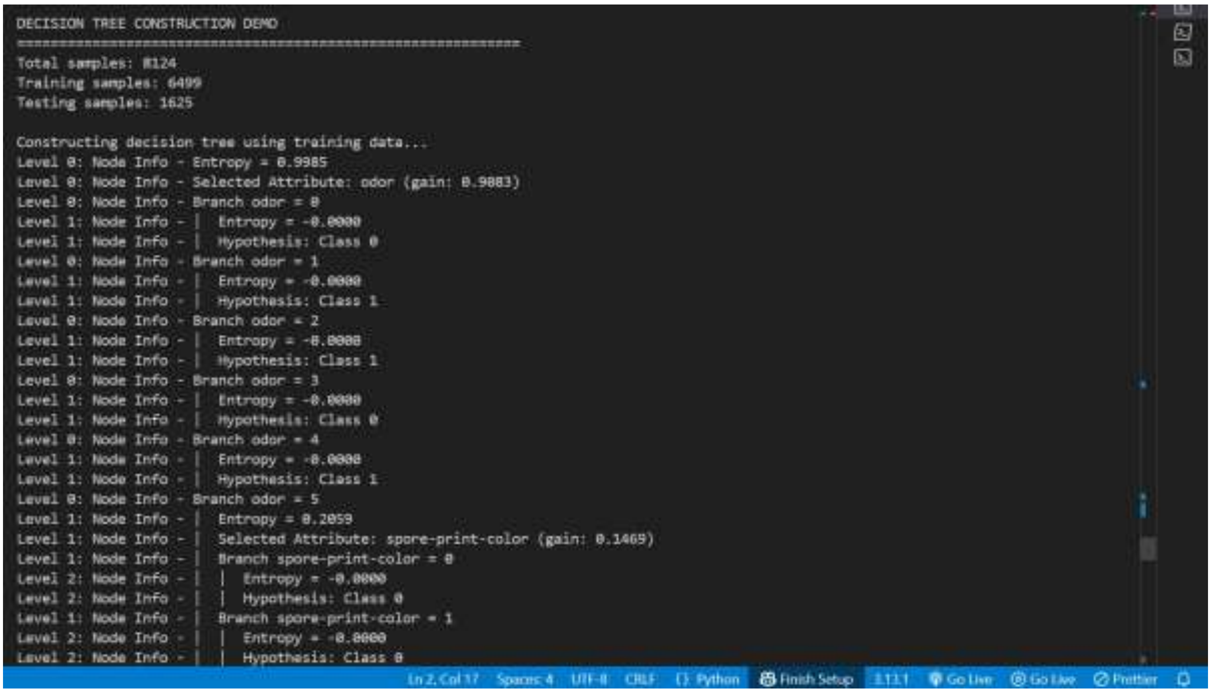
TREE VISUALIZATION



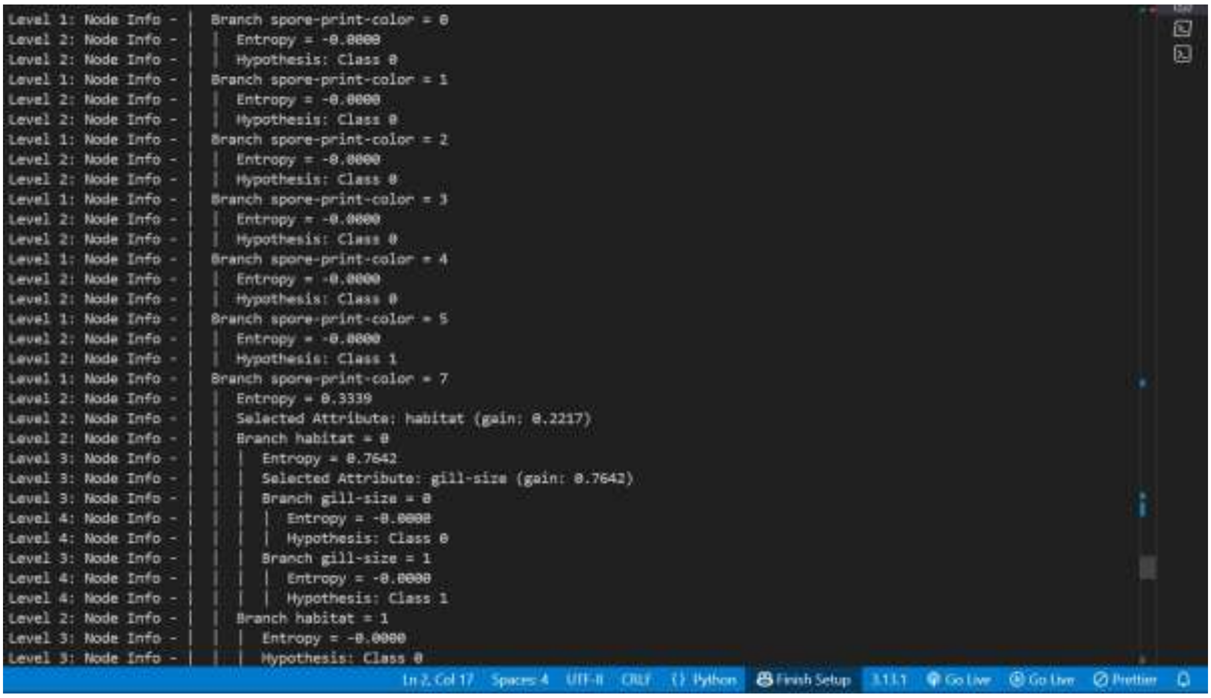
```
PS D:\CODE_ML\pytorch_implementation> python test.py --ID EC_C_PES2UG23CS134_lab3 --data mushrooms.csv --frame
work pytorch --print-tree --print-construction
>>
Running tests with PYTORCH framework
============================================================
 target column: 'class' (last column)
Original dataset info:
Shape: (8124, 23)
Columns: ['cap-shape', 'cap-surface', 'cap-color', 'bruises', 'odor', 'gill-attachment', 'gill-spacing', 'gill-size', 'gill-color', '
stalk-shape', 'stalk-root', 'stalk-surface-above-ring', 'stalk-surface-below-ring', 'stalk-color-above-ring', 'stalk-color-below-ring
', 'veil-type', 'veil-color', 'ring-number', 'ring-type', 'spore-print-color', 'population', 'habitat', 'class']

First few rows:

cap-shape: ['x' 'b' 's' 'f' 'k'] -> [5 0 4 2 3]

cap-surface: ['s' 'y' 'f' 'g'] -> [2 3 0 1]

cap-color: ['n' 'y' 'w' 'g' 'e'] -> [4 9 8 3 2]

class: ['p' 'e'] -> [1 0]

Processed dataset shape: torch.Size([8124, 23])
Number of features: 22
Features: ['cap-shape', 'cap-surface', 'cap-color', 'bruises', 'odor', 'gill-attachment', 'gill-spacing', 'gill-size', 'gill-color',
'stalk-shape', 'stalk-root', 'stalk-surface-above-ring', 'stalk-surface-below-ring', 'stalk-color-above-ring', 'stalk-color-below-rin
g', 'veil-type', 'veil-color', 'ring-number', 'ring-type', 'spore-print-color', 'population', 'habitat']
Target: class
Framework: PYTORCH
Data type: <class 'torch.Tensor'>

============================================================
DECISION TREE CONSTRUCTION DEMO
```



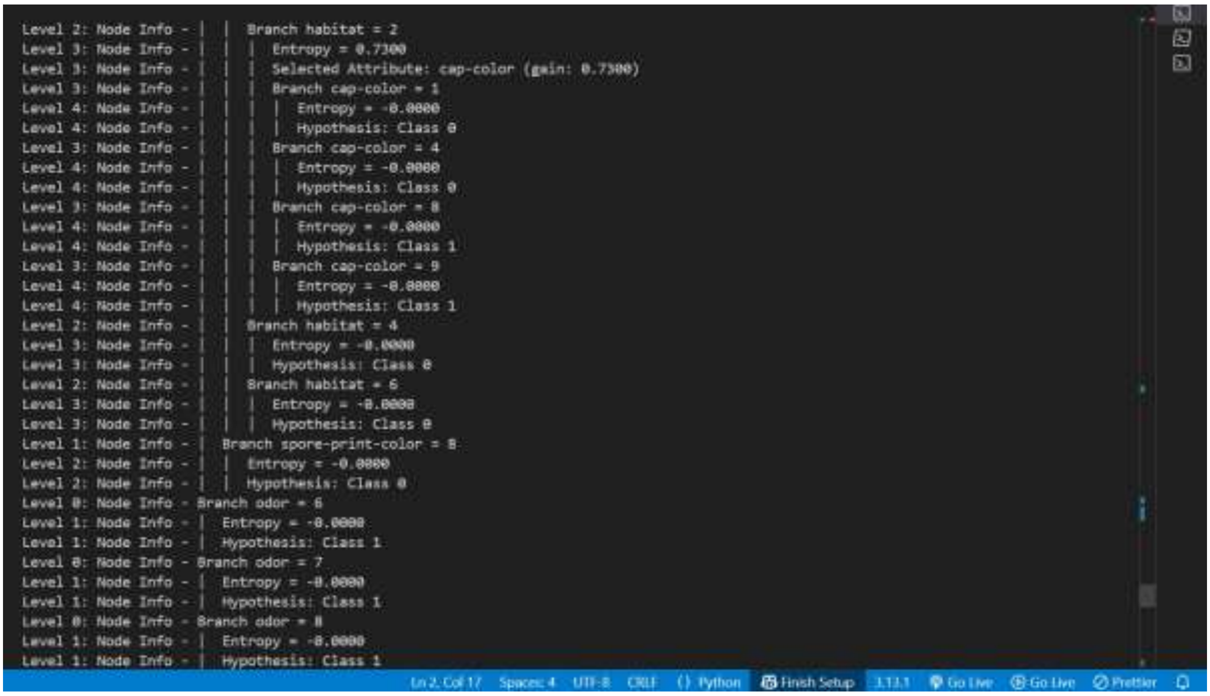
```
=================================================================
DECISION TREE CONSTRUCTION DEMO
=================================================================
Total samples: 958
Training samples: 766
Testing samples: 192

Constructing decision tree using training data...

🌳 Decision tree construction completed using PYTORCH!

📊 OVERALL PERFORMANCE METRICS
============================================
Accuracy:              0.8730 (87.30%)
Precision (weighted): 0.8741
Recall (weighted):    0.8730
F1-Score (weighted):  0.8734
Precision (macro):    0.8590
Recall (macro):       0.8638
F1-Score (macro):     0.8613

🌳 TREE COMPLEXITY METRICS
============================================
Maximum Depth:         7
Total Nodes:           281
Leaf Nodes:            180
Internal Nodes:        101
PS D:\CODE_ML\pytorch_implementation> 
```



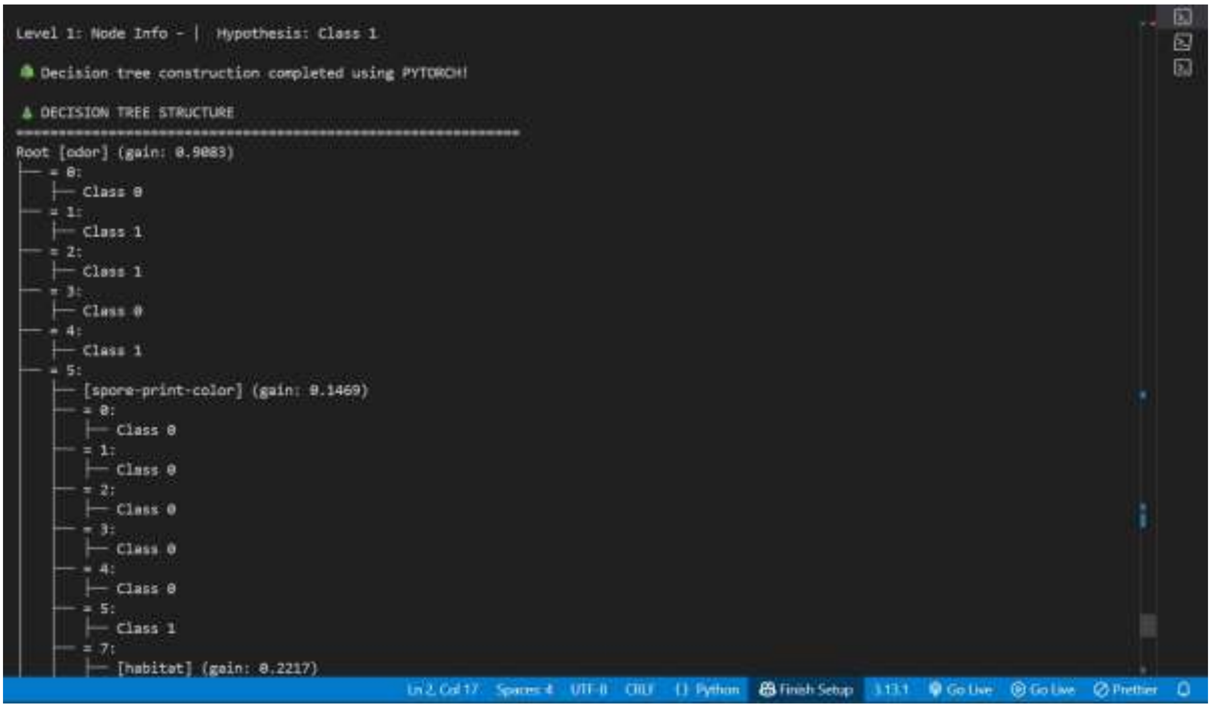
```
DECISION TREE CONSTRUCTION DEMO
=================================================================
Total samples: 8124
Training samples: 6499
Testing samples: 1625

Constructing decision tree using training data...
Level 0: Node Info - Entropy = 0.9985
Level 0: Node Info - Selected Attribute: odor (gain: 0.9083)
Level 0: Node Info - Branch odor = 0
Level 1: Node Info - |  Entropy = -0.0000
Level 1: Node Info - |  Hypothesis: Class 0
Level 0: Node Info - Branch odor = 1
Level 1: Node Info - |  Entropy = -0.0000
Level 1: Node Info - |  Hypothesis: Class 1
Level 0: Node Info - Branch odor = 2
Level 1: Node Info - |  Entropy = -0.0000
Level 1: Node Info - |  Hypothesis: Class 1
Level 0: Node Info - Branch odor = 3
Level 1: Node Info - |  Entropy = -0.0000
Level 1: Node Info - |  Hypothesis: Class 0
Level 0: Node Info - Branch odor = 4
Level 1: Node Info - |  Entropy = -0.0000
Level 1: Node Info - |  Hypothesis: Class 1
Level 0: Node Info - Branch odor = 5
Level 1: Node Info - |  Entropy = 0.2059
Level 1: Node Info - |  Selected Attribute: spore-print-color (gain: 0.1469)
Level 1: Node Info - |  Branch spore-print-color = 0
Level 2: Node Info - |  |  Entropy = -0.0000
Level 2: Node Info - |  |  Hypothesis: Class 0
Level 1: Node Info - |  Branch spore-print-color = 1
Level 2: Node Info - |  |  Entropy = -0.0000
Level 2: Node Info - |  |  Hypothesis: Class 0
```



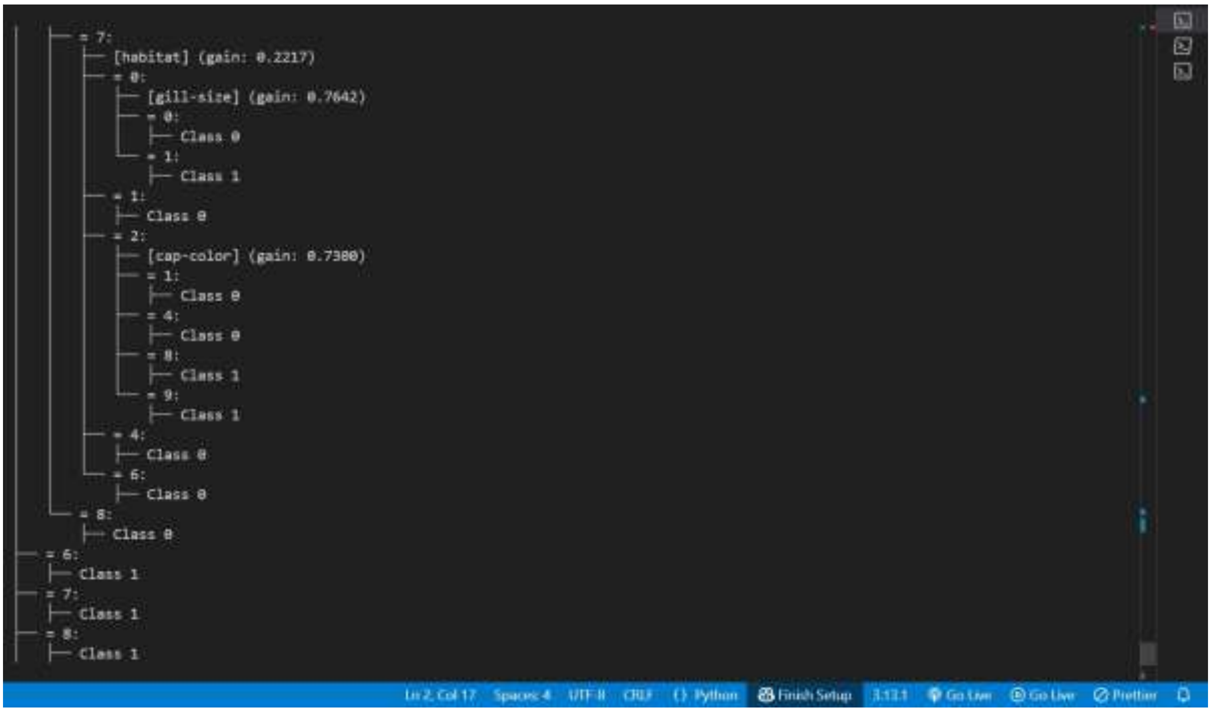
```
Level 1: Node Info - |  Branch spore-print-color = 0
Level 2: Node Info - |  |  Entropy = -0.0000
Level 2: Node Info - |  |  Hypothesis: Class 0
Level 1: Node Info - |  Branch spore-print-color = 1
Level 2: Node Info - |  |  Entropy = -0.0000
Level 2: Node Info - |  |  Hypothesis: Class 0
Level 1: Node Info - |  Branch spore-print-color = 2
Level 2: Node Info - |  |  Entropy = -0.0000
Level 2: Node Info - |  |  Hypothesis: Class 0
Level 1: Node Info - |  Branch spore-print-color = 3
Level 2: Node Info - |  |  Entropy = -0.0000
Level 2: Node Info - |  |  Hypothesis: Class 0
Level 1: Node Info - |  Branch spore-print-color = 4
Level 2: Node Info - |  |  Entropy = -0.0000
Level 2: Node Info - |  |  Hypothesis: Class 0
Level 1: Node Info - |  Branch spore-print-color = 5
Level 2: Node Info - |  |  Entropy = -0.0000
Level 2: Node Info - |  |  Hypothesis: Class 1
Level 1: Node Info - |  Branch spore-print-color = 7
Level 2: Node Info - |  |  Entropy = 0.3339
Level 2: Node Info - |  |  Selected Attribute: habitat (gain: 0.2217)
Level 2: Node Info - |  |  Branch habitat = 0
Level 3: Node Info - |  |  |  Entropy = 0.7642
Level 3: Node Info - |  |  |  Selected Attribute: gill-size (gain: 0.7642)
Level 3: Node Info - |  |  |  Branch gill-size = 0
Level 4: Node Info - |  |  |  |  Entropy = -0.0000
Level 4: Node Info - |  |  |  |  Hypothesis: Class 0
Level 3: Node Info - |  |  |  Branch gill-size = 1
Level 4: Node Info - |  |  |  |  Entropy = -0.0000
Level 4: Node Info - |  |  |  |  Hypothesis: Class 1
Level 2: Node Info - |  |  Branch habitat = 1
Level 3: Node Info - |  |  |  Entropy = -0.0000
Level 3: Node Info - |  |  |  Hypothesis: Class 0
```



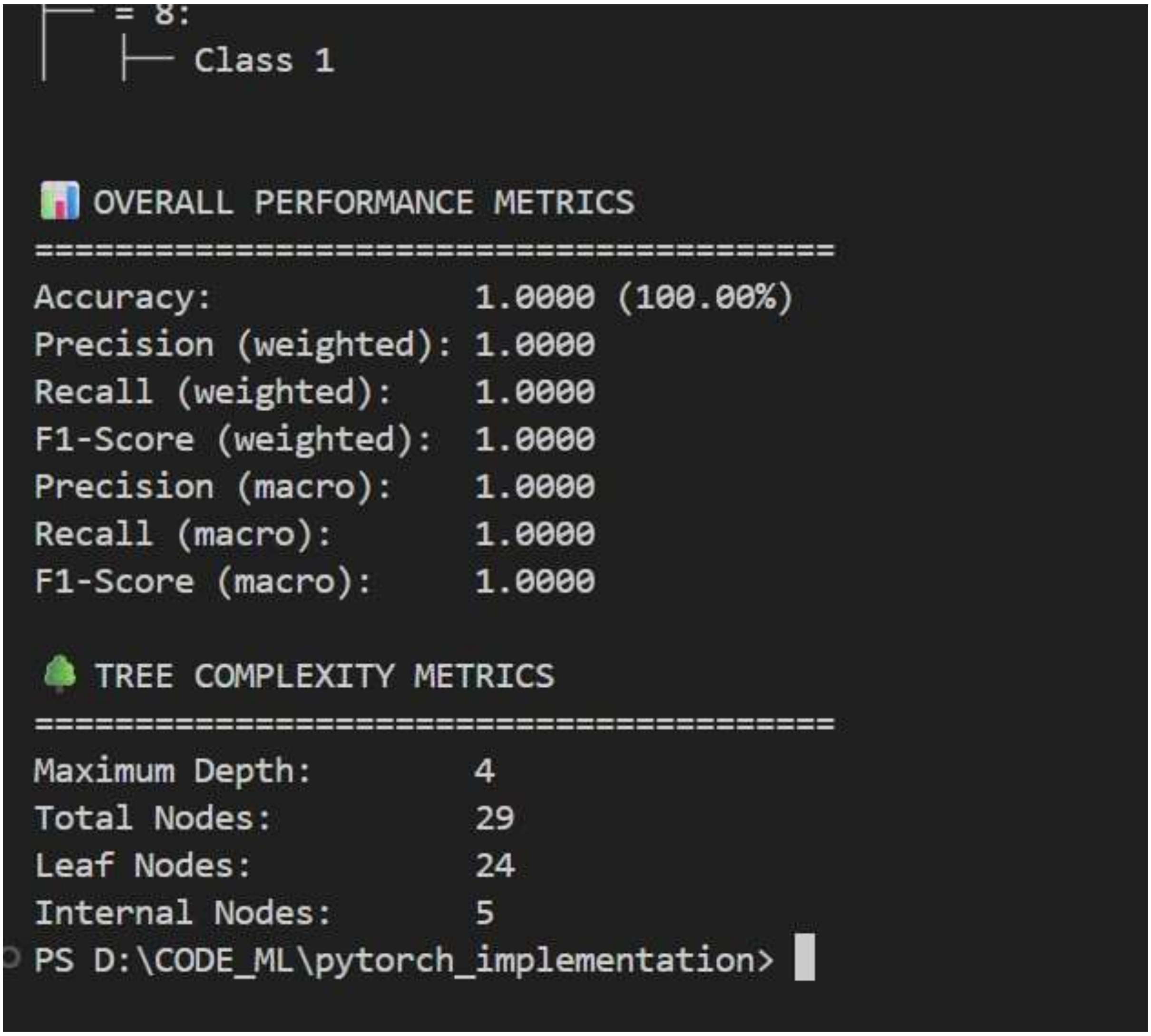
```
Level 2: Node Info - |  |  Branch habitat = 2
Level 3: Node Info - |  |  |  Entropy = 0.7300
Level 3: Node Info - |  |  |  Selected Attribute: cap-color (gain: 0.7300)
Level 3: Node Info - |  |  |  Branch cap-color = 1
Level 4: Node Info - |  |  |  |  Entropy = -0.0000
Level 4: Node Info - |  |  |  |  Hypothesis: Class 0
Level 3: Node Info - |  |  |  Branch cap-color = 4
Level 4: Node Info - |  |  |  |  Entropy = -0.0000
Level 4: Node Info - |  |  |  |  Hypothesis: Class 0
Level 3: Node Info - |  |  |  Branch cap-color = 8
Level 4: Node Info - |  |  |  |  Entropy = -0.0000
Level 4: Node Info - |  |  |  |  Hypothesis: Class 1
Level 3: Node Info - |  |  |  Branch cap-color = 9
Level 4: Node Info - |  |  |  |  Entropy = -0.0000
Level 4: Node Info - |  |  |  |  Hypothesis: Class 1
Level 2: Node Info - |  |  Branch habitat = 4
Level 3: Node Info - |  |  |  Entropy = -0.0000
Level 3: Node Info - |  |  |  Hypothesis: Class 0
Level 2: Node Info - |  |  Branch habitat = 6
Level 3: Node Info - |  |  |  Entropy = -0.0000
Level 3: Node Info - |  |  |  Hypothesis: Class 0
Level 1: Node Info - |  Branch spore-print-color = 8
Level 2: Node Info - |  |  Entropy = -0.0000
Level 2: Node Info - |  |  Hypothesis: Class 0
Level 0: Node Info - Branch odor = 6
Level 1: Node Info - |  Entropy = -0.0000
Level 1: Node Info - |  Hypothesis: Class 1
Level 0: Node Info - Branch odor = 7
Level 1: Node Info - |  Entropy = -0.0000
Level 1: Node Info - |  Hypothesis: Class 1
Level 0: Node Info - Branch odor = 8
Level 1: Node Info - |  Entropy = -0.0000
Level 1: Node Info - |  Hypothesis: Class 1
```



```
Level 1: Node Info - |  Hypothesis: Class 1

🌲 Decision tree construction completed using PYTORCH!

🌲 DECISION TREE STRUCTURE
============================================================
Root [odor] (gain: 0.9083)
├── = 0:
│   ├── Class 0
├── = 1:
│   ├── Class 1
├── = 2:
│   ├── Class 1
├── = 3:
│   ├── Class 0
├── = 4:
│   ├── Class 1
├── = 5:
│   ├── [spore-print-color] (gain: 0.1469)
│   ├── = 0:
│   │   ├── Class 0
│   ├── = 1:
│   │   ├── Class 0
│   ├── = 2:
│   │   ├── Class 0
│   ├── = 3:
│   │   ├── Class 0
│   ├── = 4:
│   │   ├── Class 0
│   ├── = 5:
│   │   ├── Class 1
│   ├── = 7:
│   │   ├── [habitat] (gain: 0.2217)
```



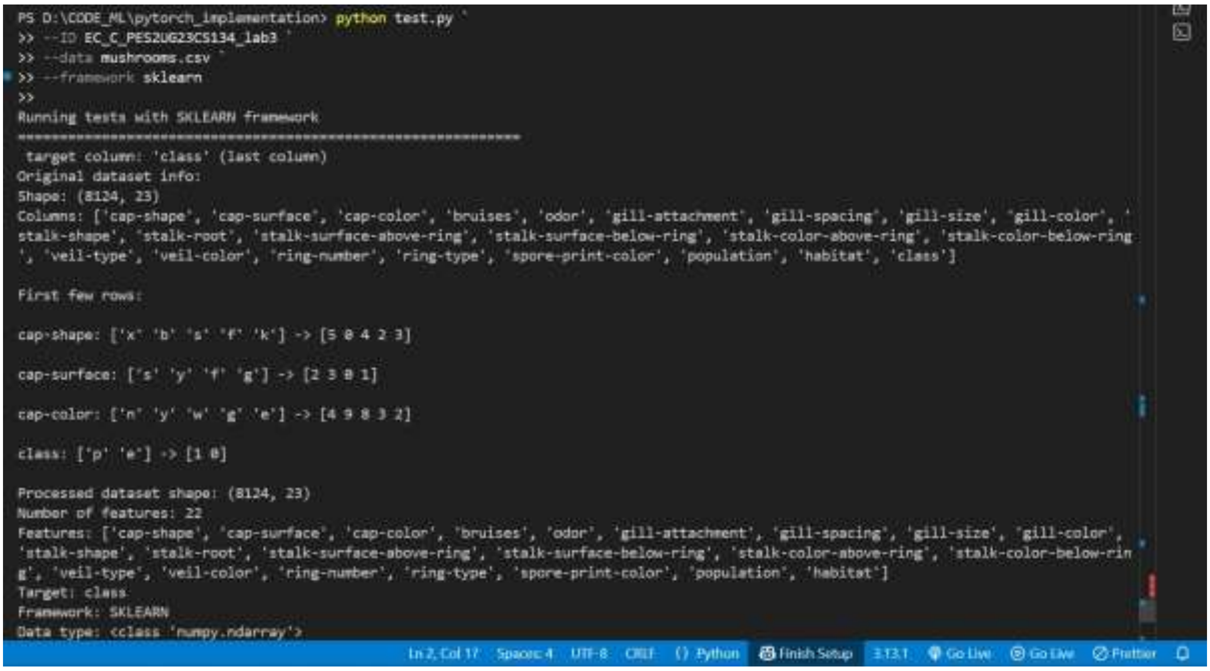
```
│   ├── = 7:
│   │   ├── [habitat] (gain: 0.2217)
│   │   ├── = 0:
│   │   │   ├── [gill-size] (gain: 0.7642)
│   │   │   ├── = 0:
│   │   │   │   ├── Class 0
│   │   │   └── = 1:
│   │   │       ├── Class 1
│   │   ├── = 1:
│   │   │   ├── Class 0
│   │   ├── = 2:
│   │   │   ├── [cap-color] (gain: 0.7380)
│   │   │   ├── = 1:
│   │   │   │   ├── Class 0
│   │   │   ├── = 4:
│   │   │   │   ├── Class 0
│   │   │   ├── = 8:
│   │   │   │   ├── Class 1
│   │   │   └── = 9:
│   │   │       ├── Class 1
│   │   ├── = 4:
│   │   │   ├── Class 0
│   │   └── = 6:
│   │       ├── Class 0
│   └── = 8:
│       ├── Class 0
├── = 6:
│   ├── Class 1
├── = 7:
│   ├── Class 1
├── = 8:
│   ├── Class 1
```



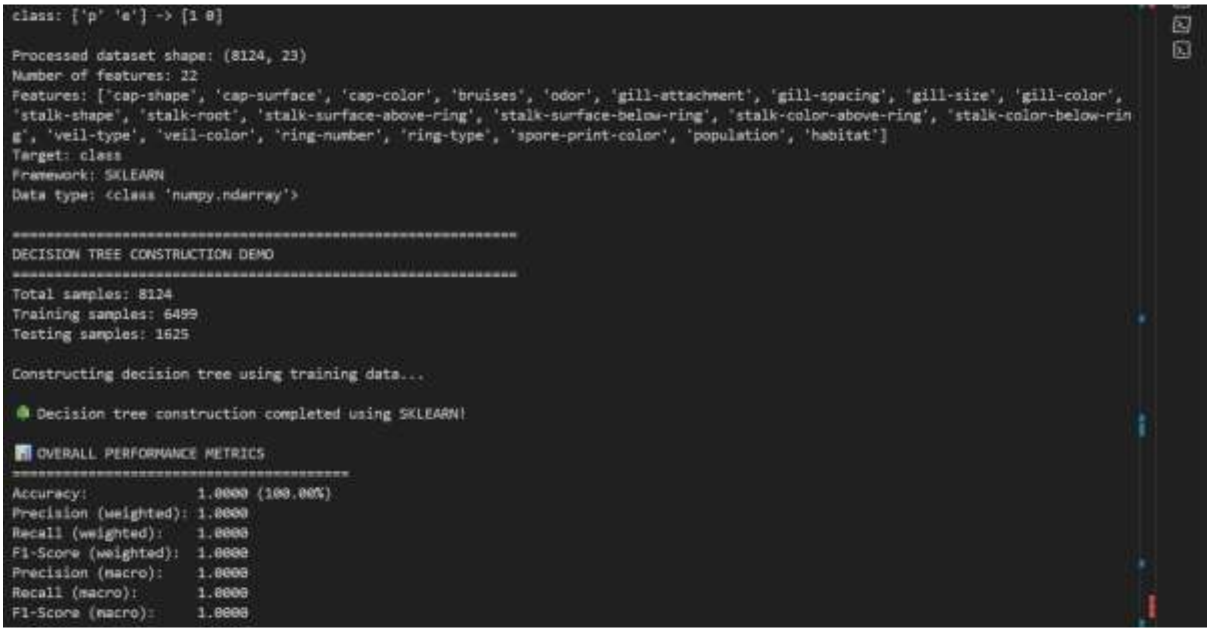
```
├── = 8:
│   ├── Class 1


📊 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
PS D:\CODE_ML\pytorch_implementation> 
```

- **Class Imbalance:** Balanced classes (e and p) ensured stable tree construction.
- **Feature Types:** Multi-valued features like odor, cap-color, and habitat performed better than binary ones.
- **Interpretability:** Very high due to shallow tree and clear splits.

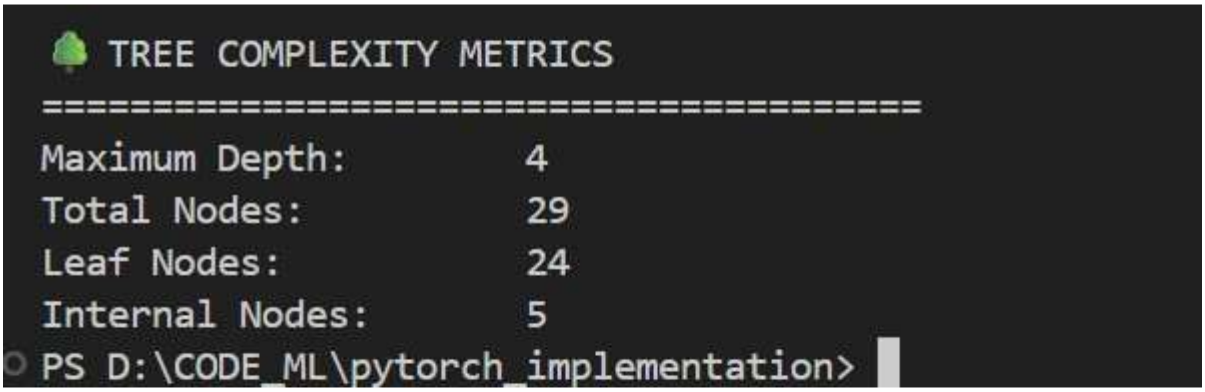
SKLEARN IMPLEMENTATIONS



```
PS D:\CODE_ML\pytorch_implementation> python test.py `
>> --ID EC_C_PES2UG23CS134_lab3 `
>> --data mushrooms.csv `
>> --framework sklearn
>>
Running tests with SKLEARN framework
============================================================
 target column: 'class' (last column)
Original dataset info:
Shape: (8124, 23)
Columns: ['cap-shape', 'cap-surface', 'cap-color', 'bruises', 'odor', 'gill-attachment', 'gill-spacing', 'gill-size', 'gill-color', '
stalk-shape', 'stalk-root', 'stalk-surface-above-ring', 'stalk-surface-below-ring', 'stalk-color-above-ring', 'stalk-color-below-ring
', 'veil-type', 'veil-color', 'ring-number', 'ring-type', 'spore-print-color', 'population', 'habitat', 'class']

First few rows:

cap-shape: ['x' 'b' 's' 'f' 'k'] -> [5 0 4 2 3]

cap-surface: ['s' 'y' 'f' 'g'] -> [2 3 0 1]

cap-color: ['n' 'y' 'w' 'g' 'e'] -> [4 9 8 3 2]

class: ['p' 'e'] -> [1 0]

Processed dataset shape: (8124, 23)
Number of features: 22
Features: ['cap-shape', 'cap-surface', 'cap-color', 'bruises', 'odor', 'gill-attachment', 'gill-spacing', 'gill-size', 'gill-color',
'stalk-shape', 'stalk-root', 'stalk-surface-above-ring', 'stalk-surface-below-ring', 'stalk-color-above-ring', 'stalk-color-below-rin
g', 'veil-type', 'veil-color', 'ring-number', 'ring-type', 'spore-print-color', 'population', 'habitat']
Target: class
Framework: SKLEARN
Data type: <class 'numpy.ndarray'>
```



```
class: ['p' 'e'] -> [1 0]

Processed dataset shape: (8124, 23)
Number of features: 22
Features: ['cap-shape', 'cap-surface', 'cap-color', 'bruises', 'odor', 'gill-attachment', 'gill-spacing', 'gill-size', 'gill-color',
'stalk-shape', 'stalk-root', 'stalk-surface-above-ring', 'stalk-surface-below-ring', 'stalk-color-above-ring', 'stalk-color-below-rin
g', 'veil-type', 'veil-color', 'ring-number', 'ring-type', 'spore-print-color', 'population', 'habitat']
Target: class
Framework: SKLEARN
Data type: <class 'numpy.ndarray'>

============================================================
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
PS D:\CODE_ML\pytorch_implementation>
```

- Useful in food safety, toxicology, and agriculture.

- **Performance Improvements:** Already optimal; pruning could improve efficiency and reduce complexity.

**ANALYSIS OF MUSHROOM DATASET:**

**Mushroom Classification**

**Accuracy**

- Achieved **100% accuracy** using PyTorch.
- Binary classification with clear distinctions (e.g., **odor**) led to highly effective splits.

**Dataset Size & Features**

- **22 features** with strong predictive power.
- Shallow tree structure (**Depth: 4**) due to high information gain and low entropy.

**Class Balance**

- Balanced classes (edible vs poisonous) ensured stable tree construction.

**Feature Types**

- Multi-valued categorical features (e.g., cap-shape, gill-color) enabled rich splits.

**Practical Applications**

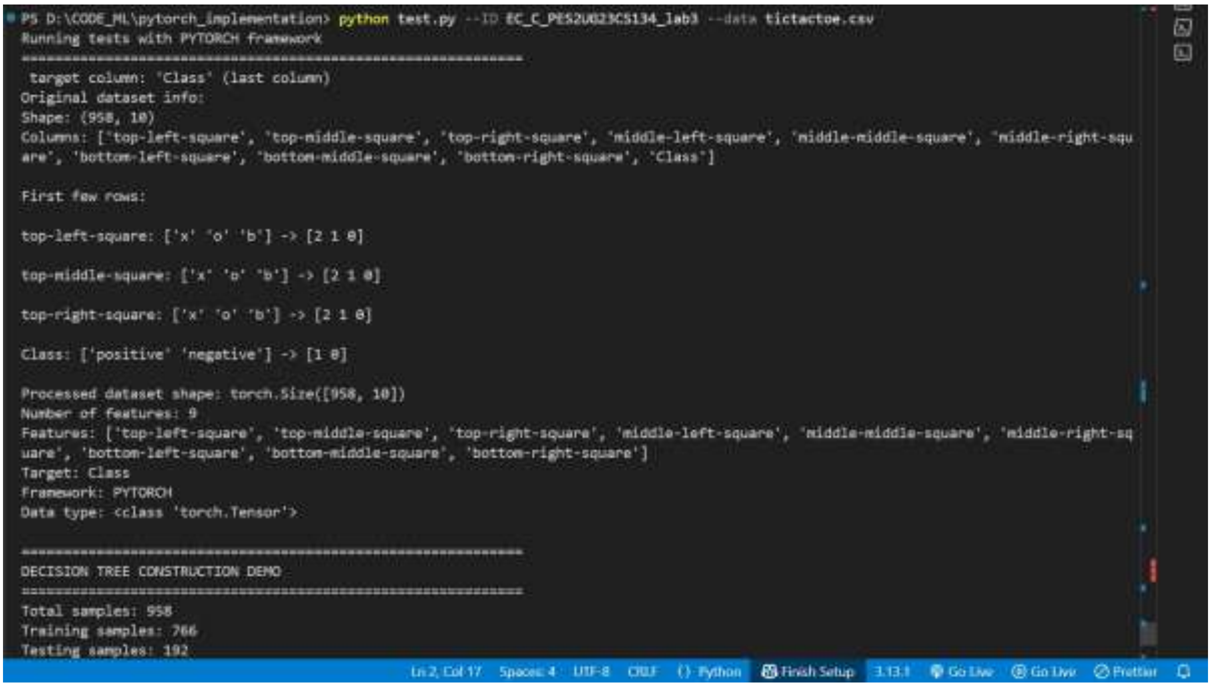
- **Food safety**, **toxicology**, and **agriculture**.
- Highly interpretable due to binary decisions and clear feature importance.

**Improvements**

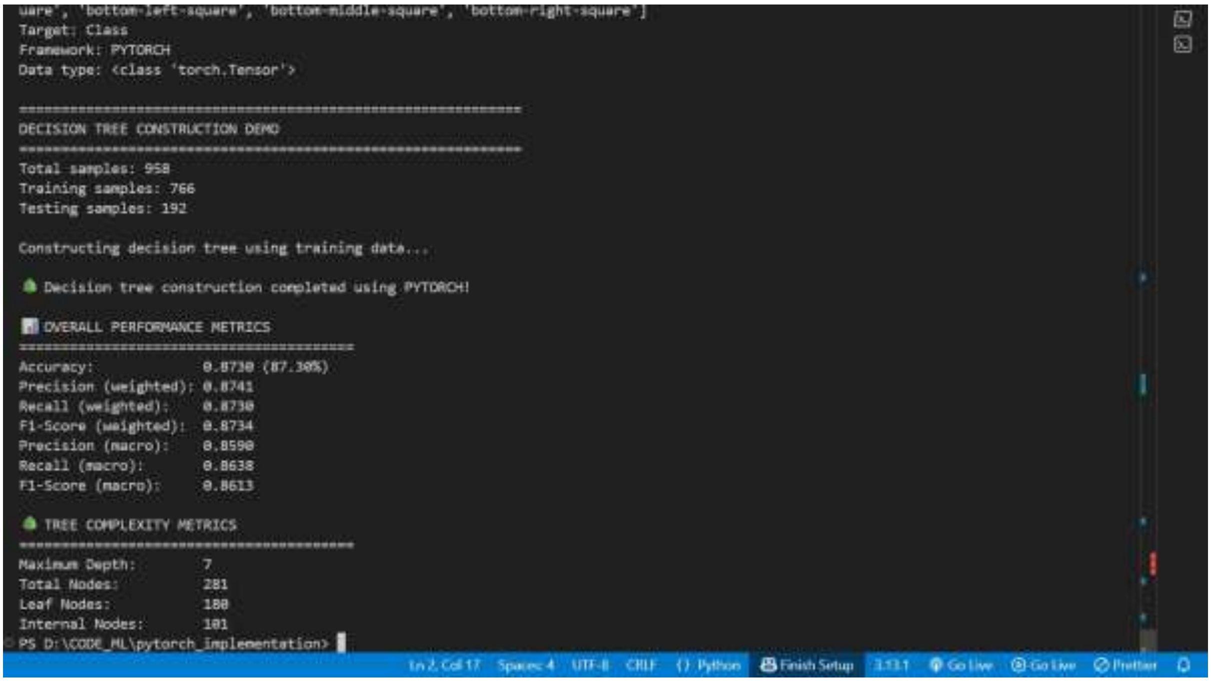
- Already optimal; **pruning** can enhance efficiency without affecting accuracy

## Dataset 2: Tic-Tac-Toe Endgame

# 1.BASIC TESTING



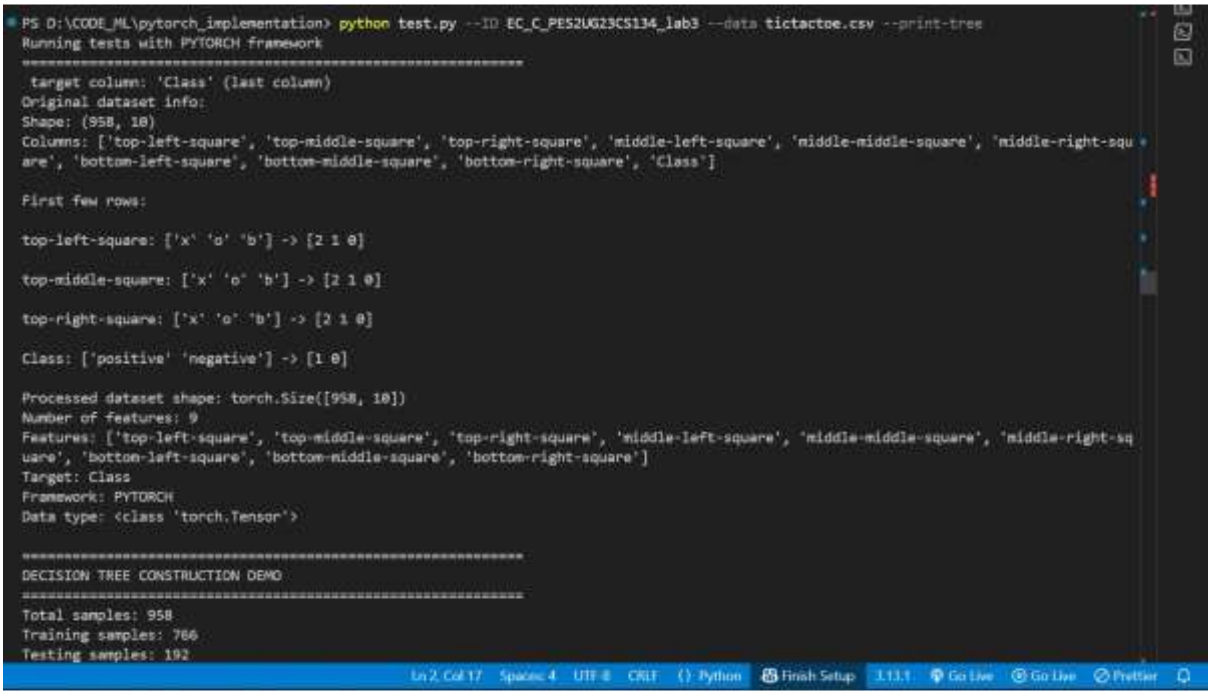
```
PS D:\CODE_ML\pytorch_implementation> python test.py --ID EC_C_PES2UG23CS134_lab3 --data tictactoe.csv
Running tests with PYTORCH framework
============================================================
 target column: 'Class' (last column)
Original dataset info:
Shape: (958, 10)
Columns: ['top-left-square', 'top-middle-square', 'top-right-square', 'middle-left-square', 'middle-middle-square', 'middle-right-squ
are', 'bottom-left-square', 'bottom-middle-square', 'bottom-right-square', 'Class']

First few rows:

top-left-square: ['x' 'o' 'b'] -> [2 1 0]

top-middle-square: ['x' 'o' 'b'] -> [2 1 0]

top-right-square: ['x' 'o' 'b'] -> [2 1 0]

Class: ['positive' 'negative'] -> [1 0]

Processed dataset shape: torch.Size([958, 10])
Number of features: 9
Features: ['top-left-square', 'top-middle-square', 'top-right-square', 'middle-left-square', 'middle-middle-square', 'middle-right-sq
uare', 'bottom-left-square', 'bottom-middle-square', 'bottom-right-square']
Target: Class
Framework: PYTORCH
Data type: <class 'torch.Tensor'>

============================================================
DECISION TREE CONSTRUCTION DEMO
============================================================
Total samples: 958
Training samples: 766
Testing samples: 192
```



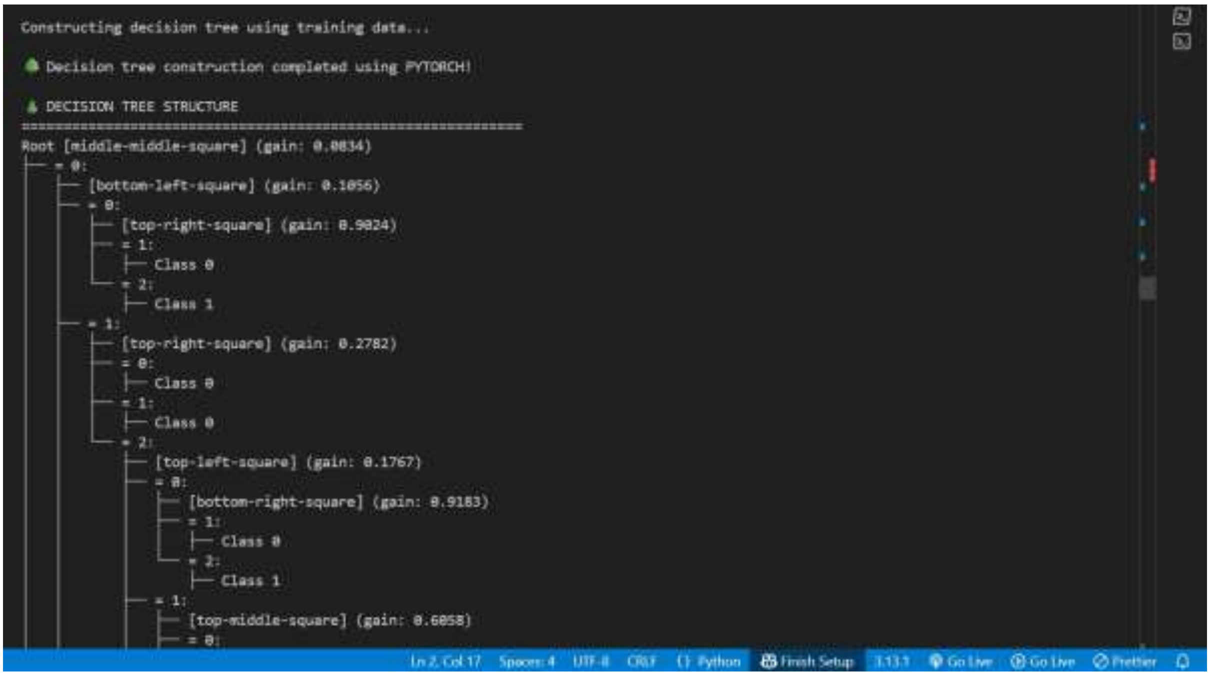
```
uare', 'bottom-left-square', 'bottom-middle-square', 'bottom-right-square']
Target: Class
Framework: PYTORCH
Data type: <class 'torch.Tensor'>

============================================================
DECISION TREE CONSTRUCTION DEMO
============================================================
Total samples: 958
Training samples: 766
Testing samples: 192

Constructing decision tree using training data...

🌳 Decision tree construction completed using PYTORCH!

📊 OVERALL PERFORMANCE METRICS
========================================
Accuracy:             0.8730 (87.30%)
Precision (weighted): 0.8741
Recall (weighted):    0.8730
F1-Score (weighted):  0.8734
Precision (macro):    0.8590
Recall (macro):       0.8638
F1-Score (macro):     0.8613

🌳 TREE COMPLEXITY METRICS
========================================
Maximum Depth:        7
Total Nodes:          281
Leaf Nodes:           180
Internal Nodes:       101
PS D:\CODE_ML\pytorch_implementation> 
```

- **Accuracy:** Achieved 87.30% using PyTorch and Sklearn.
- Binary features representing board positions are less expressive.
- **Dataset Size Impact:** Small dataset (958 samples) led to overfitting risks and lower generalization.
- **Number of Features:** 9 binary features limited the model’s ability to capture complex patterns.

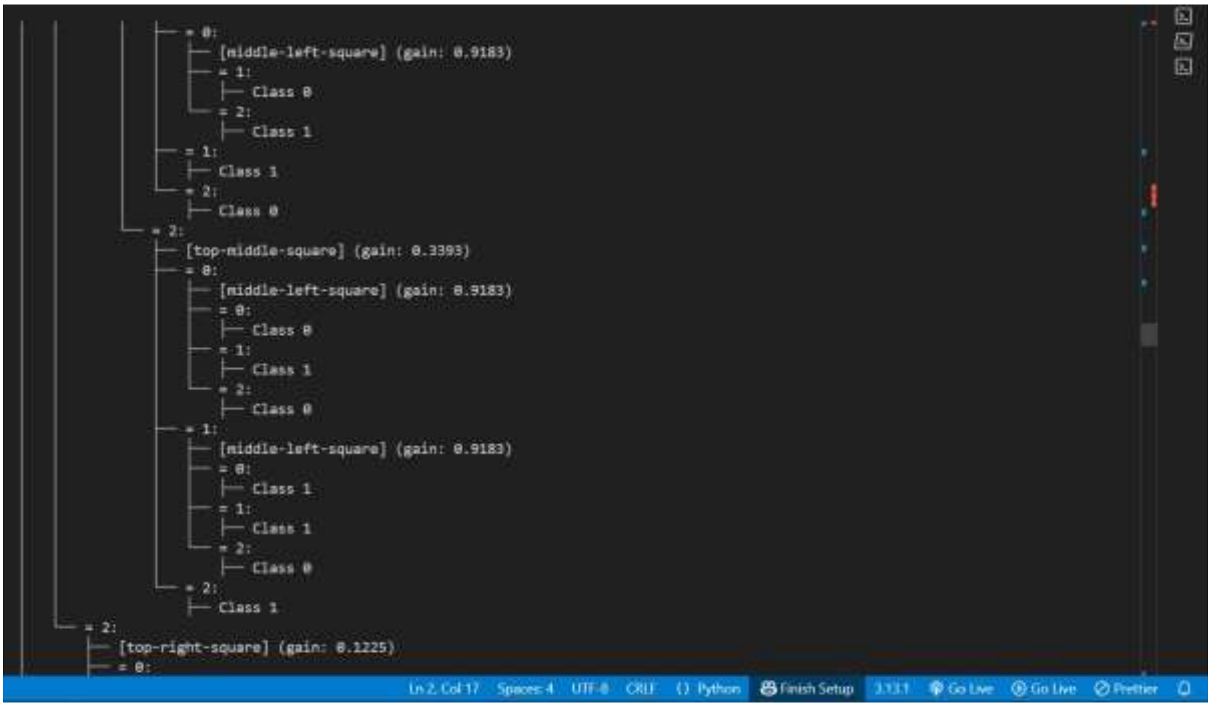
## 2.TREE VISUALIZATIONS



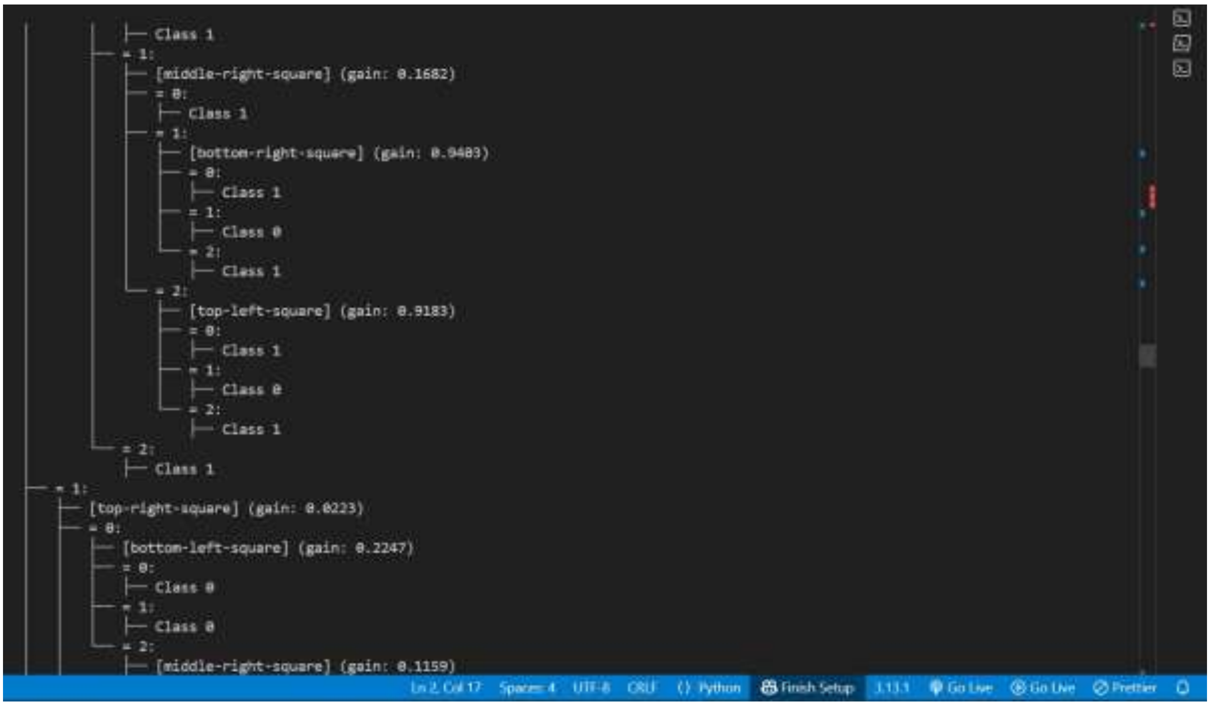
```
PS D:\CODE_ML\pytorch_implementation> python test.py --ID EC_C_PES2UG23CS134_lab3 --data tictactoe.csv --print-tree
Running tests with PYTORCH framework
============================================================
 target column: 'Class' (last column)
Original dataset info:
Shape: (958, 10)
Columns: ['top-left-square', 'top-middle-square', 'top-right-square', 'middle-left-square', 'middle-middle-square', 'middle-right-squ
are', 'bottom-left-square', 'bottom-middle-square', 'bottom-right-square', 'Class']

First few rows:

top-left-square: ['x' 'o' 'b'] -> [2 1 0]

top-middle-square: ['x' 'o' 'b'] -> [2 1 0]

top-right-square: ['x' 'o' 'b'] -> [2 1 0]

Class: ['positive' 'negative'] -> [1 0]

Processed dataset shape: torch.Size([958, 10])
Number of features: 9
Features: ['top-left-square', 'top-middle-square', 'top-right-square', 'middle-left-square', 'middle-middle-square', 'middle-right-sq
uare', 'bottom-left-square', 'bottom-middle-square', 'bottom-right-square']
Target: Class
Framework: PYTORCH
Data type: <class 'torch.Tensor'>

============================================================
DECISION TREE CONSTRUCTION DEMO
============================================================
Total samples: 958
Training samples: 766
Testing samples: 192
```



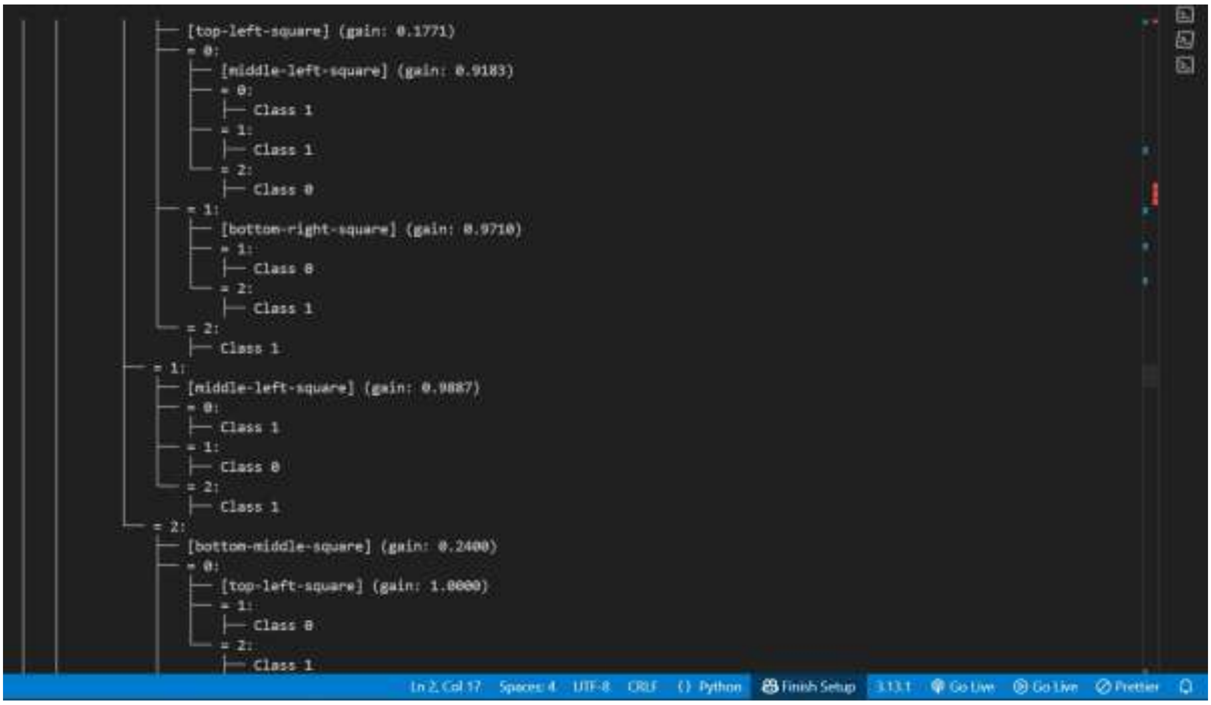
```
Constructing decision tree using training data...

🌳 Decision tree construction completed using PYTORCH!

🌲 DECISION TREE STRUCTURE
============================================================
Root [middle-middle-square] (gain: 0.0834)
├── = 0:
│   ├── [bottom-left-square] (gain: 0.1056)
│   ├── = 0:
│   │   ├── [top-right-square] (gain: 0.9824)
│   │   ├── = 1:
│   │   │   ├── Class 0
│   │   └── = 2:
│   │       ├── Class 1
│   ├── = 1:
│   │   ├── [top-right-square] (gain: 0.2782)
│   │   ├── = 0:
│   │   │   ├── Class 0
│   │   ├── = 1:
│   │   │   ├── Class 0
│   │   └── = 2:
│   │       ├── [top-left-square] (gain: 0.1767)
│   │       ├── = 0:
│   │       │   ├── [bottom-right-square] (gain: 0.9183)
│   │       │   ├── = 1:
│   │       │   │   ├── Class 0
│   │       │   └── = 2:
│   │       │       ├── Class 1
│   │       ├── = 1:
│   │       │   ├── [top-middle-square] (gain: 0.6058)
│   │       │   ├── = 0:
```



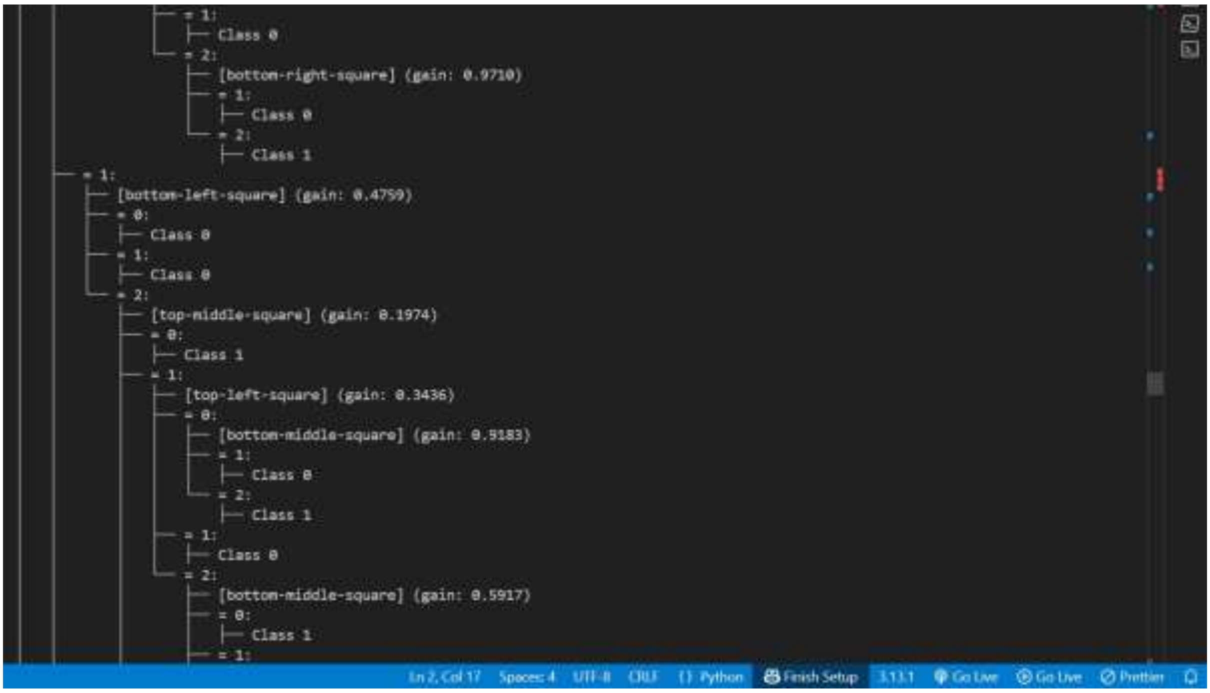
```
│   │       │   ├── = 0:
│   │       │   │   ├── [middle-left-square] (gain: 0.9183)
│   │       │   │   ├── = 1:
│   │       │   │   │   ├── Class 0
│   │       │   │   └── = 2:
│   │       │   │       ├── Class 1
│   │       │   ├── = 1:
│   │       │   │   ├── Class 1
│   │       │   └── = 2:
│   │       │       ├── Class 0
│   │       └── = 2:
│   │           ├── [top-middle-square] (gain: 0.3393)
│   │           ├── = 0:
│   │           │   ├── [middle-left-square] (gain: 0.9183)
│   │           │   ├── = 0:
│   │           │   │   ├── Class 0
│   │           │   ├── = 1:
│   │           │   │   ├── Class 1
│   │           │   └── = 2:
│   │           │       ├── Class 0
│   │           ├── = 1:
│   │           │   ├── [middle-left-square] (gain: 0.9183)
│   │           │   ├── = 0:
│   │           │   │   ├── Class 1
│   │           │   ├── = 1:
│   │           │   │   ├── Class 1
│   │           │   └── = 2:
│   │           │       ├── Class 0
│   │           └── = 2:
│   │               ├── Class 1
│   └── = 2:
│       ├── [top-right-square] (gain: 0.1225)
│       ├── = 0:
```



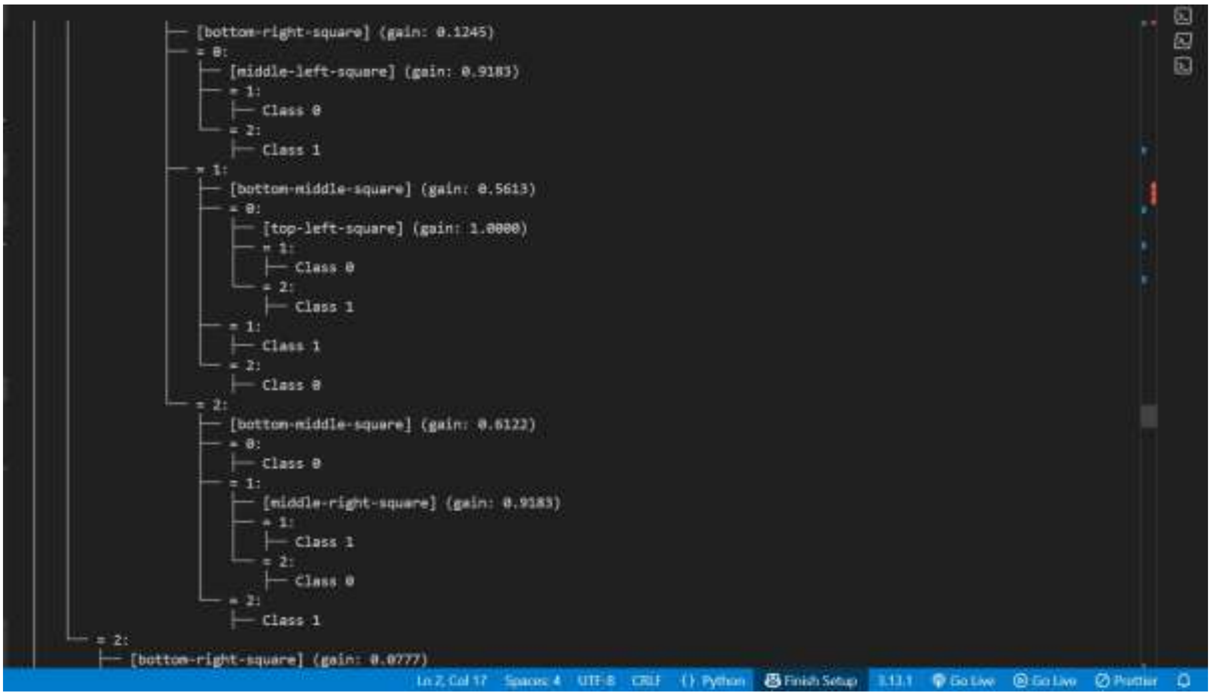
```
│       │   ├── Class 1
│       ├── = 1:
│       │   ├── [middle-right-square] (gain: 0.1682)
│       │   ├── = 0:
│       │   │   ├── Class 1
│       │   ├── = 1:
│       │   │   ├── [bottom-right-square] (gain: 0.9403)
│       │   │   ├── = 0:
│       │   │   │   ├── Class 1
│       │   │   ├── = 1:
│       │   │   │   ├── Class 0
│       │   │   └── = 2:
│       │   │       ├── Class 1
│       │   └── = 2:
│       │       ├── [top-left-square] (gain: 0.9183)
│       │       ├── = 0:
│       │       │   ├── Class 1
│       │       ├── = 1:
│       │       │   ├── Class 0
│       │       └── = 2:
│       │           ├── Class 1
│       └── = 2:
│           ├── Class 1
├── = 1:
│   ├── [top-right-square] (gain: 0.0223)
│   ├── = 0:
│   │   ├── [bottom-left-square] (gain: 0.2247)
│   │   ├── = 0:
│   │   │   ├── Class 0
│   │   ├── = 1:
│   │   │   ├── Class 0
│   │   └── = 2:
│   │       ├── [middle-right-square] (gain: 0.1159)
```



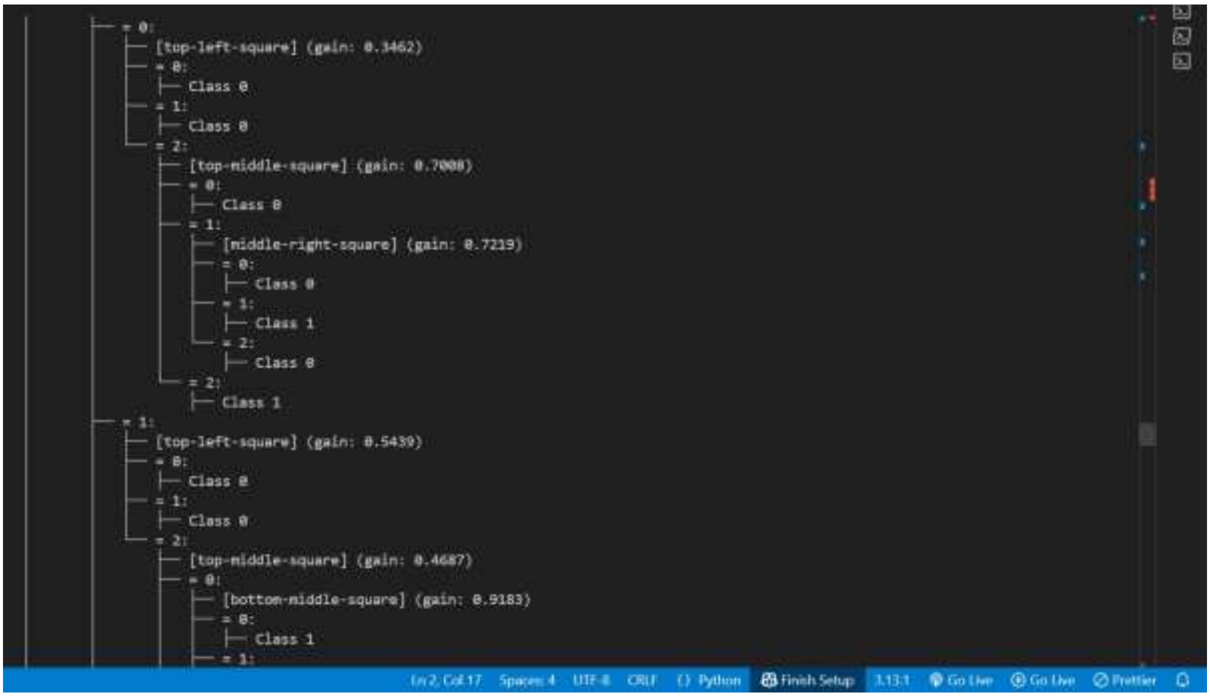
```
│   │       │   ├── [top-left-square] (gain: 0.1771)
│   │       │   ├── = 0:
│   │       │   │   ├── [middle-left-square] (gain: 0.9183)
│   │       │   │   ├── = 0:
│   │       │   │   │   ├── Class 1
│   │       │   │   ├── = 1:
│   │       │   │   │   ├── Class 1
│   │       │   │   └── = 2:
│   │       │   │       ├── Class 0
│   │       │   ├── = 1:
│   │       │   │   ├── [bottom-right-square] (gain: 0.9710)
│   │       │   │   ├── = 1:
│   │       │   │   │   ├── Class 0
│   │       │   │   └── = 2:
│   │       │   │       ├── Class 1
│   │       │   └── = 2:
│   │       │       ├── Class 1
│   │       ├── = 1:
│   │       │   ├── [middle-left-square] (gain: 0.9887)
│   │       │   ├── = 0:
│   │       │   │   ├── Class 1
│   │       │   ├── = 1:
│   │       │   │   ├── Class 0
│   │       │   └── = 2:
│   │       │       ├── Class 1
│   │       └── = 2:
│   │           ├── [bottom-middle-square] (gain: 0.2400)
│   │           ├── = 0:
│   │           │   ├── [top-left-square] (gain: 1.0000)
│   │           │   ├── = 1:
│   │           │   │   ├── Class 0
│   │           │   └── = 2:
│   │           │       ├── Class 1
```



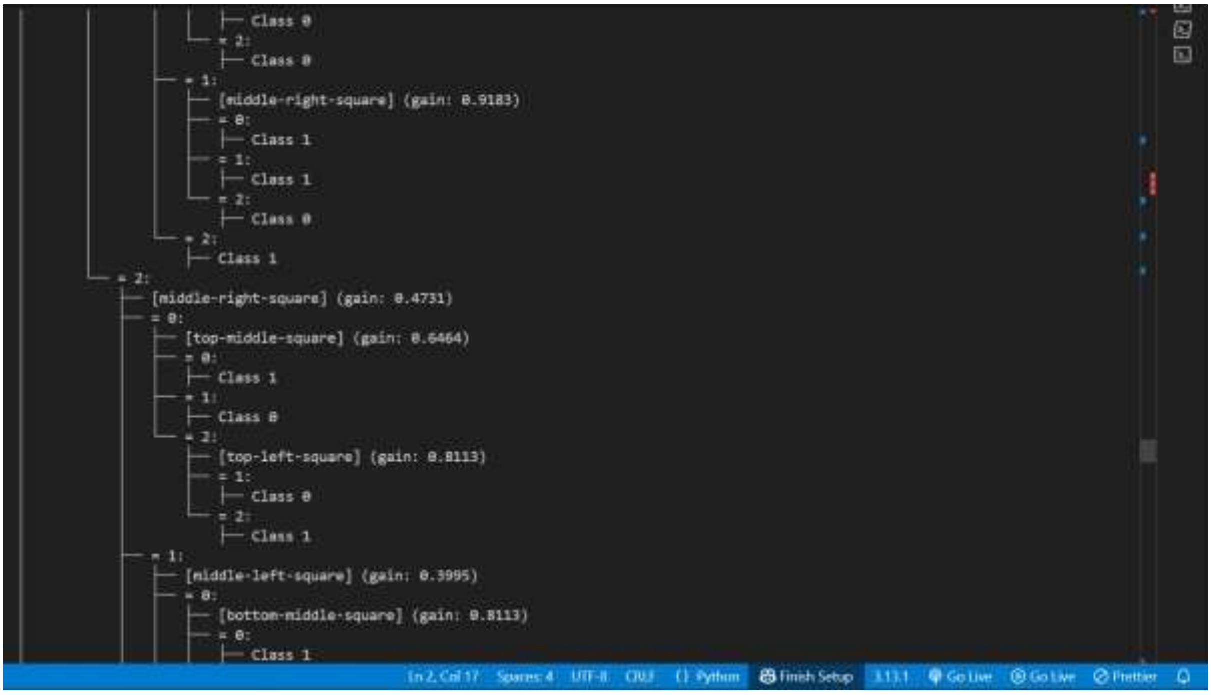
```
│   │           ├── = 1:
│   │           │   ├── Class 0
│   │           └── = 2:
│   │               ├── [bottom-right-square] (gain: 0.9710)
│   │               ├── = 1:
│   │               │   ├── Class 0
│   │               └── = 2:
│   │                   ├── Class 1
│   ├── = 1:
│   │   ├── [bottom-left-square] (gain: 0.4759)
│   │   ├── = 0:
│   │   │   ├── Class 0
│   │   ├── = 1:
│   │   │   ├── Class 0
│   │   └── = 2:
│   │       ├── [top-middle-square] (gain: 0.1974)
│   │       ├── = 0:
│   │       │   ├── Class 1
│   │       ├── = 1:
│   │       │   ├── [top-left-square] (gain: 0.3436)
│   │       │   ├── = 0:
│   │       │   │   ├── [bottom-middle-square] (gain: 0.9183)
│   │       │   │   ├── = 1:
│   │       │   │   │   ├── Class 0
│   │       │   │   └── = 2:
│   │       │   │       ├── Class 1
│   │       │   ├── = 1:
│   │       │   │   ├── Class 0
│   │       │   └── = 2:
│   │       │       ├── [bottom-middle-square] (gain: 0.5917)
│   │       │       ├── = 0:
│   │       │       │   ├── Class 1
│   │       │       ├── = 1:
```



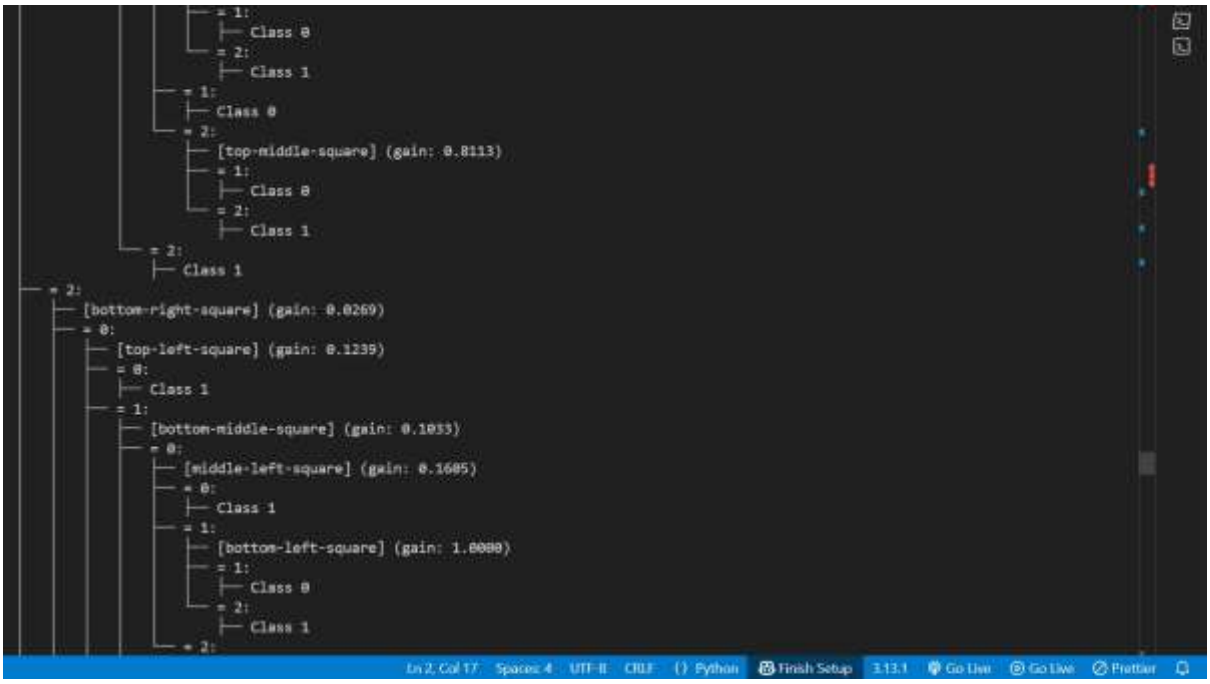
```
│   │           ├── [bottom-right-square] (gain: 0.1245)
│   │           ├── = 0:
│   │           │   ├── [middle-left-square] (gain: 0.9183)
│   │           │   ├── = 1:
│   │           │   │   ├── Class 0
│   │           │   └── = 2:
│   │           │       ├── Class 1
│   │           ├── = 1:
│   │           │   ├── [bottom-middle-square] (gain: 0.5613)
│   │           │   ├── = 0:
│   │           │   │   ├── [top-left-square] (gain: 1.0000)
│   │           │   │   ├── = 1:
│   │           │   │   │   ├── Class 0
│   │           │   │   └── = 2:
│   │           │   │       ├── Class 1
│   │           │   ├── = 1:
│   │           │   │   ├── Class 1
│   │           │   └── = 2:
│   │           │       ├── Class 0
│   │           └── = 2:
│   │               ├── [bottom-middle-square] (gain: 0.6122)
│   │               ├── = 0:
│   │               │   ├── Class 0
│   │               ├── = 1:
│   │               │   ├── [middle-right-square] (gain: 0.9183)
│   │               │   ├── = 1:
│   │               │   │   ├── Class 1
│   │               │   └── = 2:
│   │               │       ├── Class 0
│   │               └── = 2:
│   │                   ├── Class 1
│   └── = 2:
│       ├── [bottom-right-square] (gain: 0.0777)
```



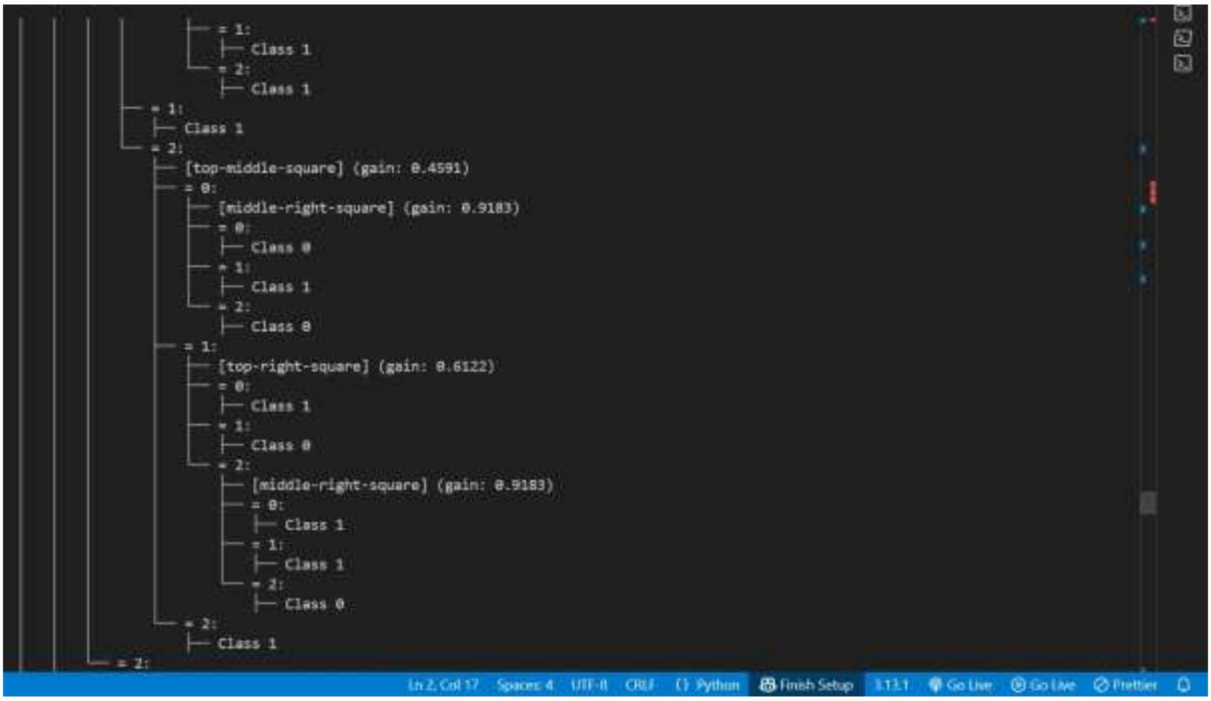
```
│       ├── = 0:
│       │   ├── [top-left-square] (gain: 0.3462)
│       │   ├── = 0:
│       │   │   ├── Class 0
│       │   ├── = 1:
│       │   │   ├── Class 0
│       │   └── = 2:
│       │       ├── [top-middle-square] (gain: 0.7000)
│       │       ├── = 0:
│       │       │   ├── Class 0
│       │       ├── = 1:
│       │       │   ├── [middle-right-square] (gain: 0.7219)
│       │       │   ├── = 0:
│       │       │   │   ├── Class 0
│       │       │   ├── = 1:
│       │       │   │   ├── Class 1
│       │       │   └── = 2:
│       │       │       ├── Class 0
│       │       └── = 2:
│       │           ├── Class 1
│       ├── = 1:
│       │   ├── [top-left-square] (gain: 0.5439)
│       │   ├── = 0:
│       │   │   ├── Class 0
│       │   ├── = 1:
│       │   │   ├── Class 0
│       │   └── = 2:
│       │       ├── [top-middle-square] (gain: 0.4687)
│       │       ├── = 0:
│       │       │   ├── [bottom-middle-square] (gain: 0.9183)
│       │       │   ├── = 0:
│       │       │   │   ├── Class 1
│       │       │   ├── = 1:
```



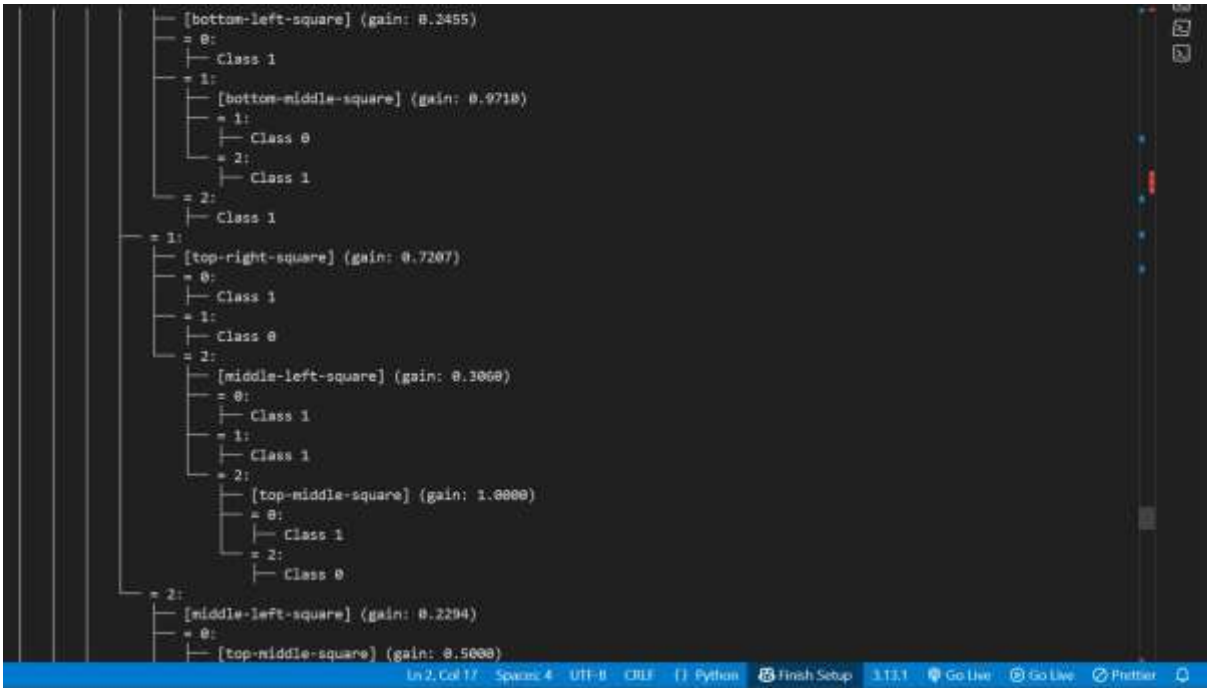
```
│       │       │   │   ├── Class 0
│       │       │   └── = 2:
│       │       │       ├── Class 0
│       │       ├── = 1:
│       │       │   ├── [middle-right-square] (gain: 0.9183)
│       │       │   ├── = 0:
│       │       │   │   ├── Class 1
│       │       │   ├── = 1:
│       │       │   │   ├── Class 1
│       │       │   └── = 2:
│       │       │       ├── Class 0
│       │       └── = 2:
│       │           ├── Class 1
│       └── = 2:
│           ├── [middle-right-square] (gain: 0.4731)
│           ├── = 0:
│           │   ├── [top-middle-square] (gain: 0.6464)
│           │   ├── = 0:
│           │   │   ├── Class 1
│           │   ├── = 1:
│           │   │   ├── Class 0
│           │   └── = 2:
│           │       ├── [top-left-square] (gain: 0.8113)
│           │       ├── = 1:
│           │       │   ├── Class 0
│           │       └── = 2:
│           │           ├── Class 1
│           ├── = 1:
│           │   ├── [middle-left-square] (gain: 0.3995)
│           │   ├── = 0:
│           │   │   ├── [bottom-middle-square] (gain: 0.8113)
│           │   │   ├── = 0:
│           │   │   │   ├── Class 1
```



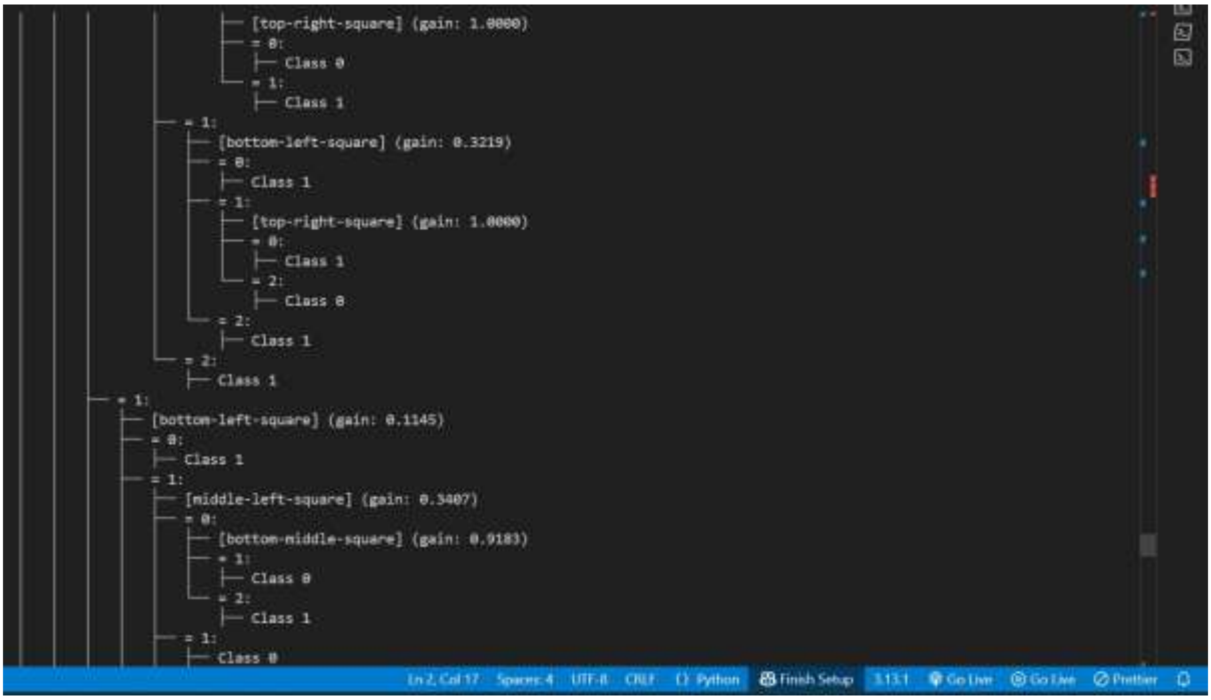
```
│           │   │   ├── = 1:
│           │   │   │   ├── Class 0
│           │   │   └── = 2:
│           │   │       ├── Class 1
│           │   ├── = 1:
│           │   │   ├── Class 0
│           │   └── = 2:
│           │       ├── [top-middle-square] (gain: 0.8113)
│           │       ├── = 1:
│           │       │   ├── Class 0
│           │       └── = 2:
│           │           ├── Class 1
│           └── = 2:
│               ├── Class 1
├── = 2:
│   ├── [bottom-right-square] (gain: 0.0269)
│   ├── = 0:
│   │   ├── [top-left-square] (gain: 0.1239)
│   │   ├── = 0:
│   │   │   ├── Class 1
│   │   ├── = 1:
│   │   │   ├── [bottom-middle-square] (gain: 0.1033)
│   │   │   ├── = 0:
│   │   │   │   ├── [middle-left-square] (gain: 0.1605)
│   │   │   │   ├── = 0:
│   │   │   │   │   ├── Class 1
│   │   │   │   ├── = 1:
│   │   │   │   │   ├── [bottom-left-square] (gain: 1.0000)
│   │   │   │   │   ├── = 1:
│   │   │   │   │   │   ├── Class 0
│   │   │   │   │   └── = 2:
│   │   │   │   │       ├── Class 1
│   │   │   │   └── = 2:
```



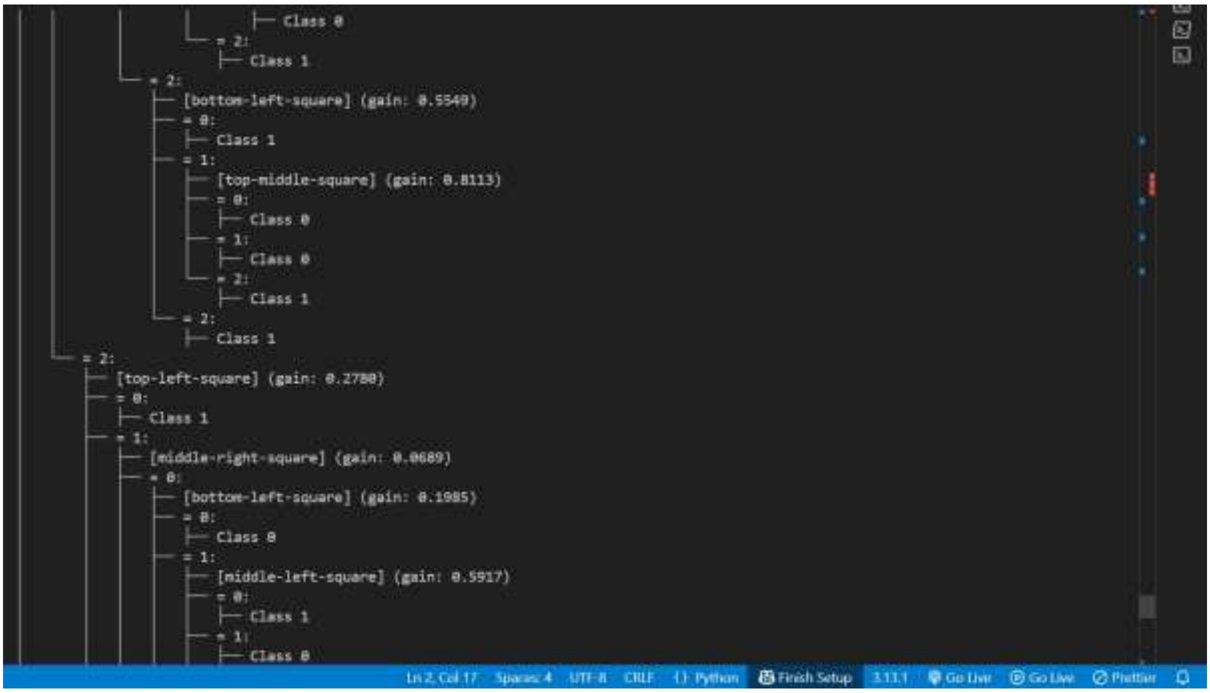
```
│   │   │   │       ├── = 1:
│   │   │   │       │   ├── Class 1
│   │   │   │       └── = 2:
│   │   │   │           ├── Class 1
│   │   │   ├── = 1:
│   │   │   │   ├── Class 1
│   │   │   └── = 2:
│   │   │       ├── [top-middle-square] (gain: 0.4591)
│   │   │       ├── = 0:
│   │   │       │   ├── [middle-right-square] (gain: 0.9183)
│   │   │       │   ├── = 0:
│   │   │       │   │   ├── Class 0
│   │   │       │   ├── = 1:
│   │   │       │   │   ├── Class 1
│   │   │       │   └── = 2:
│   │   │       │       ├── Class 0
│   │   │       ├── = 1:
│   │   │       │   ├── [top-right-square] (gain: 0.6122)
│   │   │       │   ├── = 0:
│   │   │       │   │   ├── Class 1
│   │   │       │   ├── = 1:
│   │   │       │   │   ├── Class 0
│   │   │       │   └── = 2:
│   │   │       │       ├── [middle-right-square] (gain: 0.9183)
│   │   │       │       ├── = 0:
│   │   │       │       │   ├── Class 1
│   │   │       │       ├── = 1:
│   │   │       │       │   ├── Class 1
│   │   │       │       └── = 2:
│   │   │       │           ├── Class 0
│   │   │       └── = 2:
│   │   │           ├── Class 1
│   │   └── = 2:
```



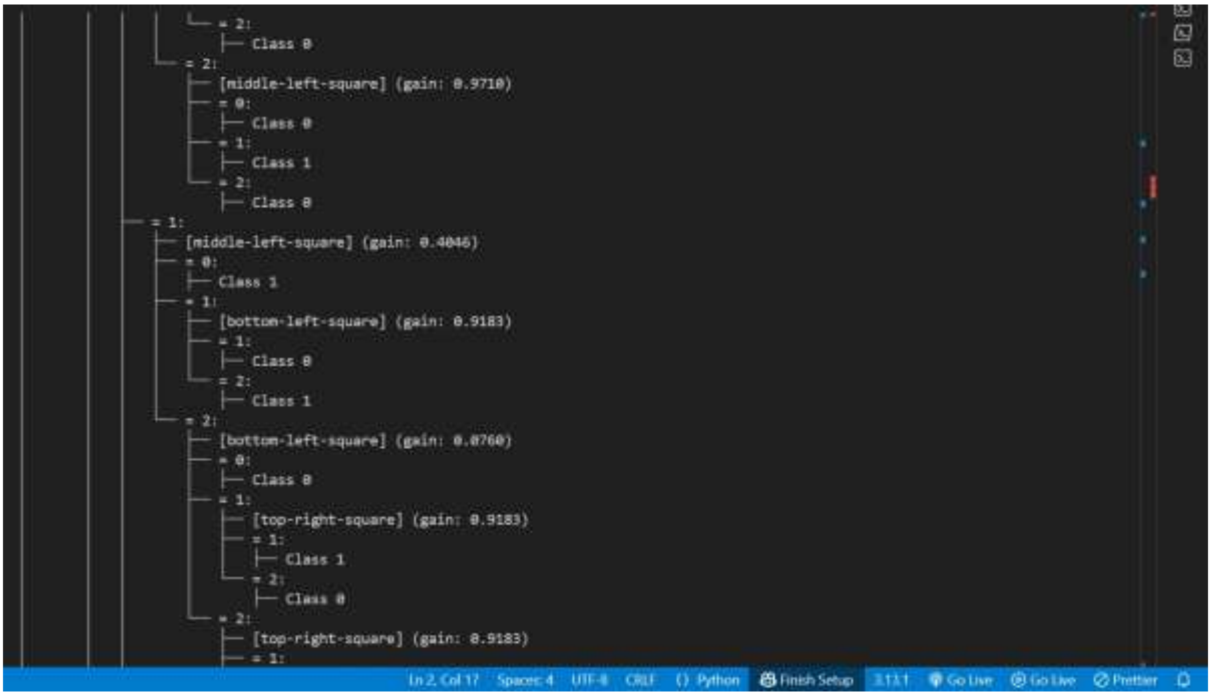
```
│   │   │   │   ├── [bottom-left-square] (gain: 0.2455)
│   │   │   │   ├── = 0:
│   │   │   │   │   ├── Class 1
│   │   │   │   ├── = 1:
│   │   │   │   │   ├── [bottom-middle-square] (gain: 0.9710)
│   │   │   │   │   ├── = 1:
│   │   │   │   │   │   ├── Class 0
│   │   │   │   │   └── = 2:
│   │   │   │   │       ├── Class 1
│   │   │   │   └── = 2:
│   │   │   │       ├── Class 1
│   │   │   ├── = 1:
│   │   │   │   ├── [top-right-square] (gain: 0.7207)
│   │   │   │   ├── = 0:
│   │   │   │   │   ├── Class 1
│   │   │   │   ├── = 1:
│   │   │   │   │   ├── Class 0
│   │   │   │   └── = 2:
│   │   │   │       ├── [middle-left-square] (gain: 0.3060)
│   │   │   │       ├── = 0:
│   │   │   │       │   ├── Class 1
│   │   │   │       ├── = 1:
│   │   │   │       │   ├── Class 1
│   │   │   │       └── = 2:
│   │   │   │           ├── [top-middle-square] (gain: 1.0000)
│   │   │   │           ├── = 0:
│   │   │   │           │   ├── Class 1
│   │   │   │           └── = 2:
│   │   │   │               ├── Class 0
│   │   │   └── = 2:
│   │   │       ├── [middle-left-square] (gain: 0.2294)
│   │   │       ├── = 0:
│   │   │       │   ├── [top-middle-square] (gain: 0.5000)
```



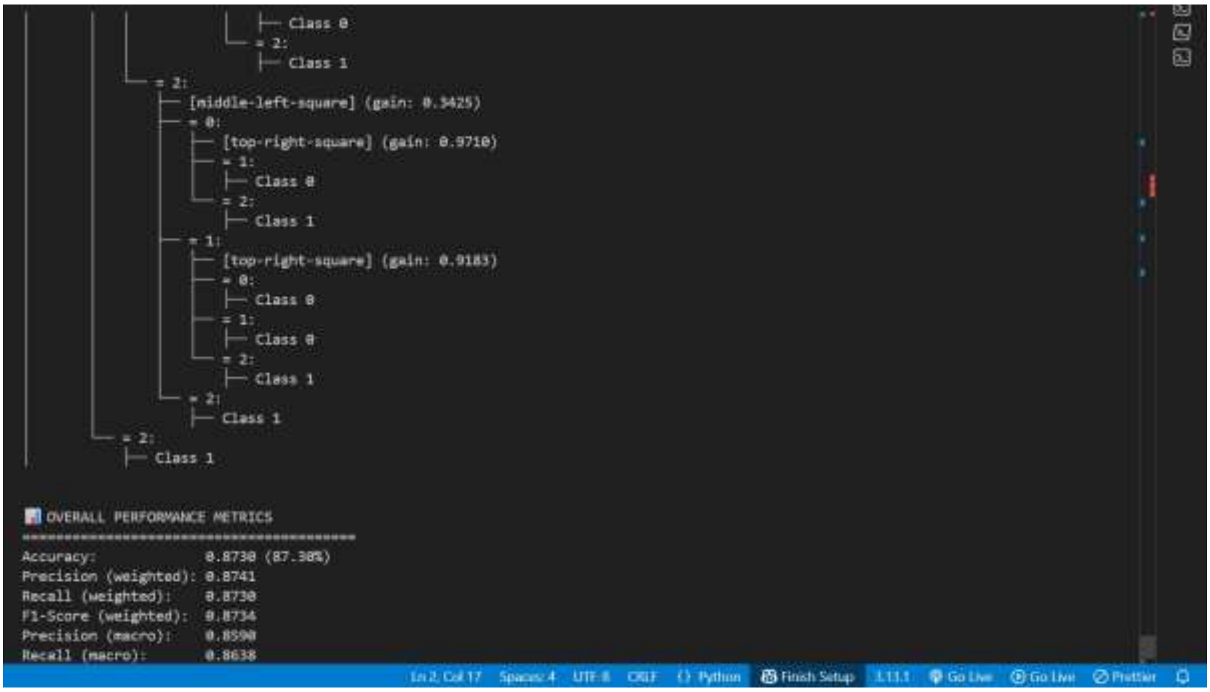
```
│   │   │       │       ├── [top-right-square] (gain: 1.0000)
│   │   │       │       ├── = 0:
│   │   │       │       │   ├── Class 0
│   │   │       │       └── = 1:
│   │   │       │           ├── Class 1
│   │   │       ├── = 1:
│   │   │       │   ├── [bottom-left-square] (gain: 0.3219)
│   │   │       │   ├── = 0:
│   │   │       │   │   ├── Class 1
│   │   │       │   ├── = 1:
│   │   │       │   │   ├── [top-right-square] (gain: 1.0000)
│   │   │       │   │   ├── = 0:
│   │   │       │   │   │   ├── Class 1
│   │   │       │   │   └── = 2:
│   │   │       │   │       ├── Class 0
│   │   │       │   └── = 2:
│   │   │       │       ├── Class 1
│   │   │       └── = 2:
│   │   │           ├── Class 1
│   │   ├── = 1:
│   │   │   ├── [bottom-left-square] (gain: 0.1145)
│   │   │   ├── = 0:
│   │   │   │   ├── Class 1
│   │   │   ├── = 1:
│   │   │   │   ├── [middle-left-square] (gain: 0.3407)
│   │   │   │   ├── = 0:
│   │   │   │   │   ├── [bottom-middle-square] (gain: 0.9183)
│   │   │   │   │   ├── = 1:
│   │   │   │   │   │   ├── Class 0
│   │   │   │   │   └── = 2:
│   │   │   │   │       ├── Class 1
│   │   │   │   ├── = 1:
│   │   │   │   │   ├── Class 0
```



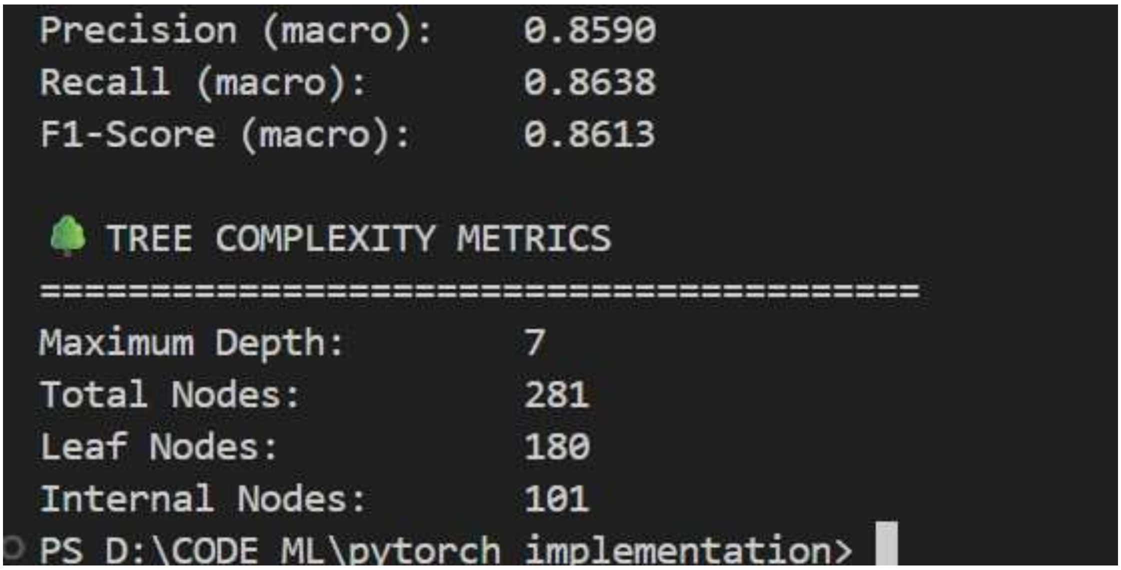
```
│   │       │       │       ├── Class 0
│   │       │       └── = 2:
│   │       │           ├── Class 1
│   │       └── = 2:
│   │           ├── [bottom-left-square] (gain: 0.5549)
│   │           ├── = 0:
│   │           │   ├── Class 1
│   │           ├── = 1:
│   │           │   ├── [top-middle-square] (gain: 0.8113)
│   │           │   ├── = 0:
│   │           │   │   ├── Class 0
│   │           │   ├── = 1:
│   │           │   │   ├── Class 0
│   │           │   └── = 2:
│   │           │       ├── Class 1
│   │           └── = 2:
│   │               ├── Class 1
│   └── = 2:
│       ├── [top-left-square] (gain: 0.2780)
│       ├── = 0:
│       │   ├── Class 1
│       ├── = 1:
│       │   ├── [middle-right-square] (gain: 0.0689)
│       │   ├── = 0:
│       │   │   ├── [bottom-left-square] (gain: 0.1985)
│       │   │   ├── = 0:
│       │   │   │   ├── Class 0
│       │   │   ├── = 1:
│       │   │   │   ├── [middle-left-square] (gain: 0.5917)
│       │   │   │   ├── = 0:
│       │   │   │   │   ├── Class 1
│       │   │   │   ├── = 1:
│       │   │   │   │   ├── Class 0
```



```
  │       │   │   │   └── = 2:
  │       │   │   │       ├── Class 0
  │       │   │   └── = 2:
  │       │   │       ├── [middle-left-square] (gain: 0.9710)
  │       │   │       ├── = 0:
  │       │   │       │   ├── Class 0
  │       │   │       ├── = 1:
  │       │   │       │   ├── Class 1
  │       │   │       └── = 2:
  │       │   │           ├── Class 0
  │       │   ├── = 1:
  │       │   │   ├── [middle-left-square] (gain: 0.4046)
  │       │   │   ├── = 0:
  │       │   │   │   ├── Class 1
  │       │   │   ├── = 1:
  │       │   │   │   ├── [bottom-left-square] (gain: 0.9183)
  │       │   │   │   ├── = 1:
  │       │   │   │   │   ├── Class 0
  │       │   │   │   └── = 2:
  │       │   │   │       ├── Class 1
  │       │   │   └── = 2:
  │       │   │       ├── [bottom-left-square] (gain: 0.0760)
  │       │   │       ├── = 0:
  │       │   │       │   ├── Class 0
  │       │   │       ├── = 1:
  │       │   │       │   ├── [top-right-square] (gain: 0.9183)
  │       │   │       │   ├── = 1:
  │       │   │       │   │   ├── Class 1
  │       │   │       │   └── = 2:
  │       │   │       │       ├── Class 0
  │       │   │       └── = 2:
  │       │   │           ├── [top-right-square] (gain: 0.9183)
  │       │   │           ├── = 1:
```



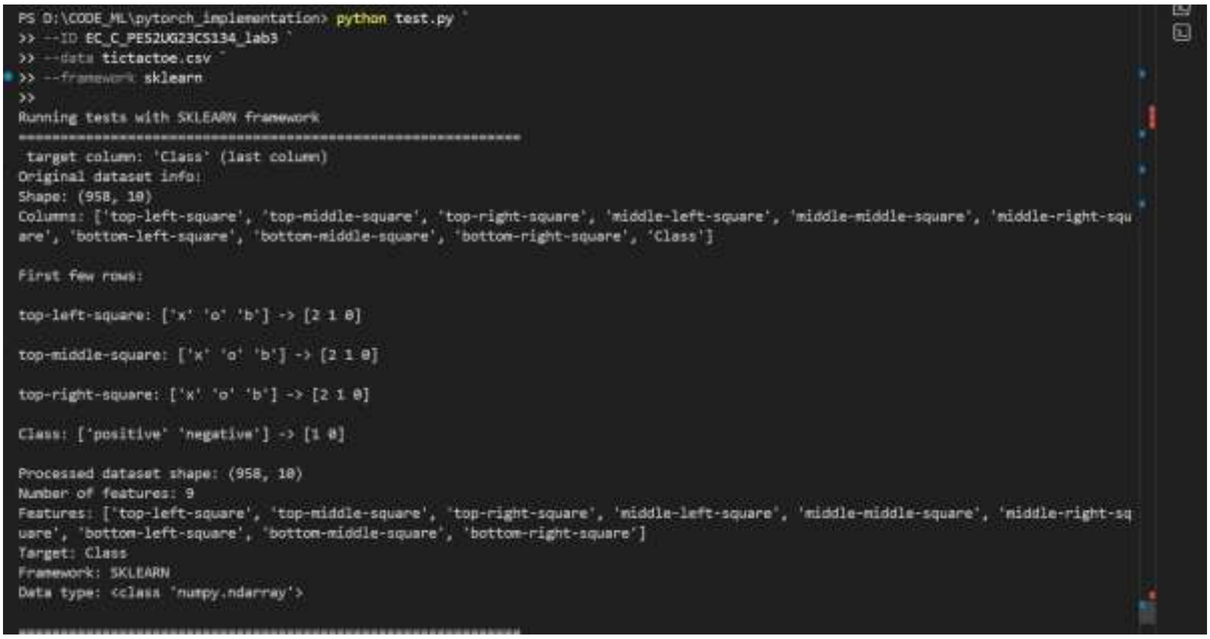
```
  │       │   │               │   ├── Class 0
  │       │   │               └── = 2:
  │       │   │                   ├── Class 1
  │       │   └── = 2:
  │       │       ├── [middle-left-square] (gain: 0.3425)
  │       │       ├── = 0:
  │       │       │   ├── [top-right-square] (gain: 0.9710)
  │       │       │   ├── = 1:
  │       │       │   │   ├── Class 0
  │       │       │   └── = 2:
  │       │       │       ├── Class 1
  │       │       ├── = 1:
  │       │       │   ├── [top-right-square] (gain: 0.9183)
  │       │       │   ├── = 0:
  │       │       │   │   ├── Class 0
  │       │       │   ├── = 1:
  │       │       │   │   ├── Class 0
  │       │       │   └── = 2:
  │       │       │       ├── Class 1
  │       │       └── = 2:
  │       │           ├── Class 1
  │       └── = 2:
  │           ├── Class 1

📊 OVERALL PERFORMANCE METRICS
========================================
Accuracy:              0.8730 (87.30%)
Precision (weighted): 0.8741
Recall (weighted):     0.8730
F1-Score (weighted):  0.8734
Precision (macro):     0.8590
Recall (macro):        0.8638
```



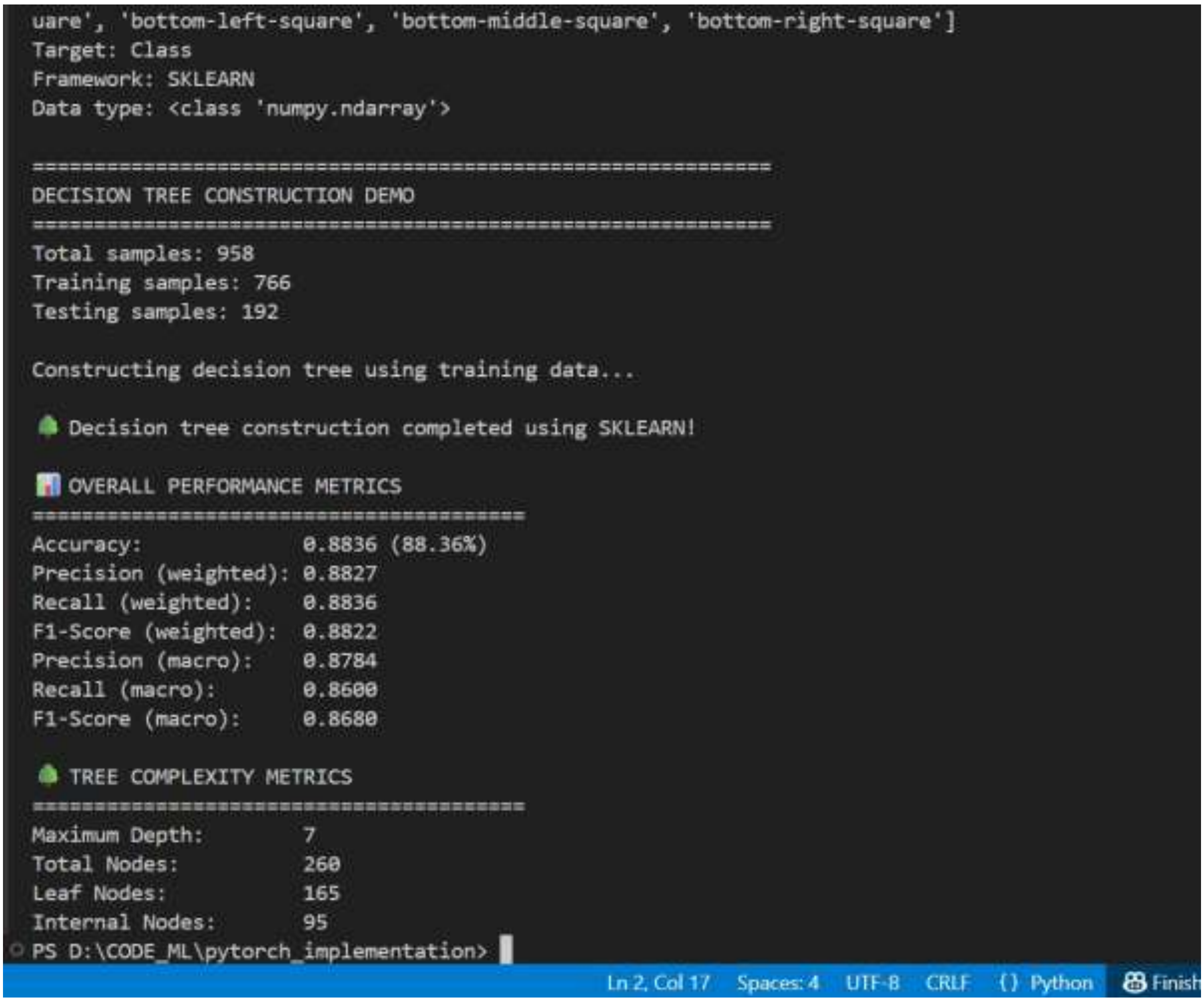
```
Precision (macro):    0.8590
Recall (macro):       0.8638
F1-Score (macro):     0.8613

🌳 TREE COMPLEXITY METRICS
========================================
Maximum Depth:        7
Total Nodes:          281
Leaf Nodes:           180
Internal Nodes:       101
PS D:\CODE ML\pytorch implementation>
```

- **Class Imbalance:** Balanced classes (positive and negative) supported stable tree growth.
- **Feature Types:** Binary features were simple but lacked depth; multi-valued features would perform better.
- **Interpretability:** Moderate; tree depth increased complexity.

## 3. SKLEARN IMPLEMENTATIONS



```
PS D:\CODE_ML\pytorch_implementation> python test.py `
>> --ID EC_C_PES2UG23CS134_lab3 `
>> --data tictactoe.csv `
>> --framework sklearn
>>
Running tests with SKLEARN framework
============================================================
 target column: 'Class' (last column)
Original dataset info:
Shape: (958, 10)
Columns: ['top-left-square', 'top-middle-square', 'top-right-square', 'middle-left-square', 'middle-middle-square', 'middle-right-squ
are', 'bottom-left-square', 'bottom-middle-square', 'bottom-right-square', 'Class']

First few rows:

top-left-square: ['x' 'o' 'b'] -> [2 1 0]

top-middle-square: ['x' 'o' 'b'] -> [2 1 0]

top-right-square: ['x' 'o' 'b'] -> [2 1 0]

Class: ['positive' 'negative'] -> [1 0]

Processed dataset shape: (958, 10)
Number of features: 9
Features: ['top-left-square', 'top-middle-square', 'top-right-square', 'middle-left-square', 'middle-middle-square', 'middle-right-sq
uare', 'bottom-left-square', 'bottom-middle-square', 'bottom-right-square']
Target: Class
Framework: SKLEARN
Data type: <class 'numpy.ndarray'>

============================================================
```



```
uare', 'bottom-left-square', 'bottom-middle-square', 'bottom-right-square']
Target: Class
Framework: SKLEARN
Data type: <class 'numpy.ndarray'>

============================================================
DECISION TREE CONSTRUCTION DEMO
============================================================
Total samples: 958
Training samples: 766
Testing samples: 192

Constructing decision tree using training data...

🌳 Decision tree construction completed using SKLEARN!

📊 OVERALL PERFORMANCE METRICS
========================================
Accuracy:              0.8836 (88.36%)
Precision (weighted): 0.8827
Recall (weighted):     0.8836
F1-Score (weighted):   0.8822
Precision (macro):     0.8784
Recall (macro):        0.8600
F1-Score (macro):      0.8680

🌳 TREE COMPLEXITY METRICS
========================================
Maximum Depth:         7
Total Nodes:           260
Leaf Nodes:            165
Internal Nodes:        95
PS D:\CODE_ML\pytorch_implementation> 
```

- **Real-World Applications:** Game AI, pattern recognition, and strategic modeling.
- **Performance Improvements:** Feature engineering or ensemble methods (e.g., Random Forest) could improve accuracy.

ANALYSIS OF TICTACTOE DATASET:

**Accuracy**

- Achieved **87.30% accuracy**.
- Lower than Mushroom due to complex decision paths and fewer samples.

**Dataset Size & Features**

- **958 samples**, **9 binary features** representing board positions.
- Tree had **281 nodes**, indicating complexity despite fewer features.

**Class Balance**

- Balanced win/loss classes supported stable tree growth.

**Feature Types**

- Binary features ('x', 'o', 'b') are simple but less expressive.
- Limited feature diversity affects decision tree depth and accuracy.

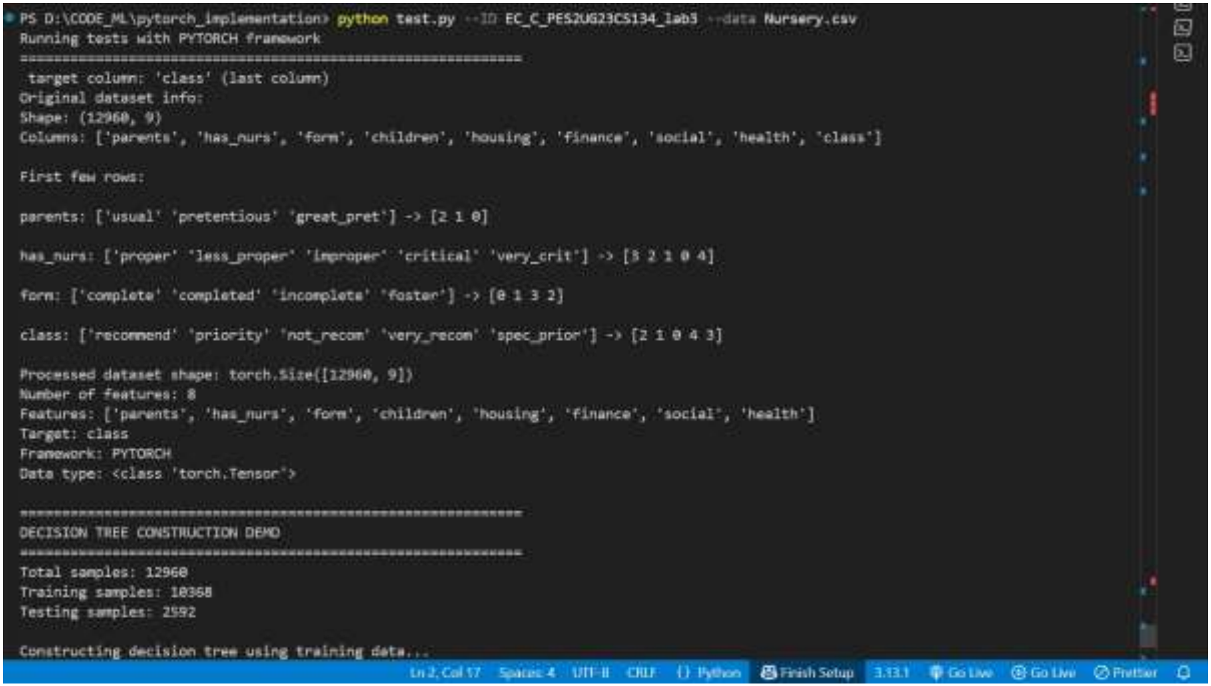
**Practical Applications**

- Useful in **game AI**, **pattern recognition**, and **strategy modeling**.

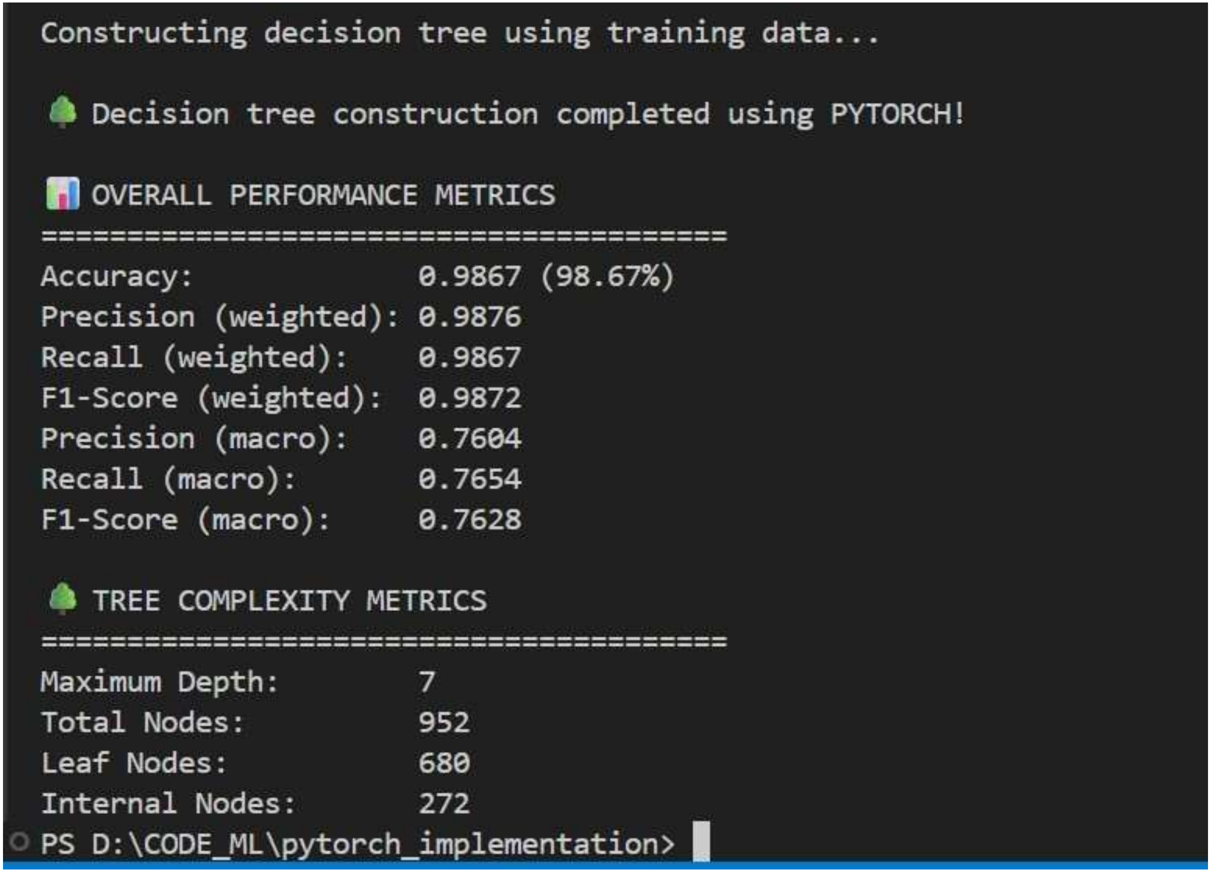
**Improvements**

- Apply **feature engineering** or **ensemble methods** to improve accuracy.

Dataset 3: Nursery School 1.BASIC TESTING



```
PS D:\CODE_ML\pytorch_implementation> python test.py --ID EC_C_PES2UG23CS134_lab3 --data Nursery.csv
Running tests with PYTORCH framework
============================================================
 target column: 'class' (last column)
Original dataset info:
Shape: (12960, 9)
Columns: ['parents', 'has_nurs', 'form', 'children', 'housing', 'finance', 'social', 'health', 'class']

First few rows:

parents: ['usual' 'pretentious' 'great_pret'] -> [2 1 0]

has_nurs: ['proper' 'less_proper' 'improper' 'critical' 'very_crit'] -> [3 2 1 0 4]

form: ['complete' 'completed' 'incomplete' 'foster'] -> [0 1 3 2]

class: ['recommend' 'priority' 'not_recom' 'very_recom' 'spec_prior'] -> [2 1 0 4 3]

Processed dataset shape: torch.Size([12960, 9])
Number of features: 8
Features: ['parents', 'has_nurs', 'form', 'children', 'housing', 'finance', 'social', 'health']
Target: class
Framework: PYTORCH
Data type: <class 'torch.Tensor'>

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

Constructing decision tree using training data...
```

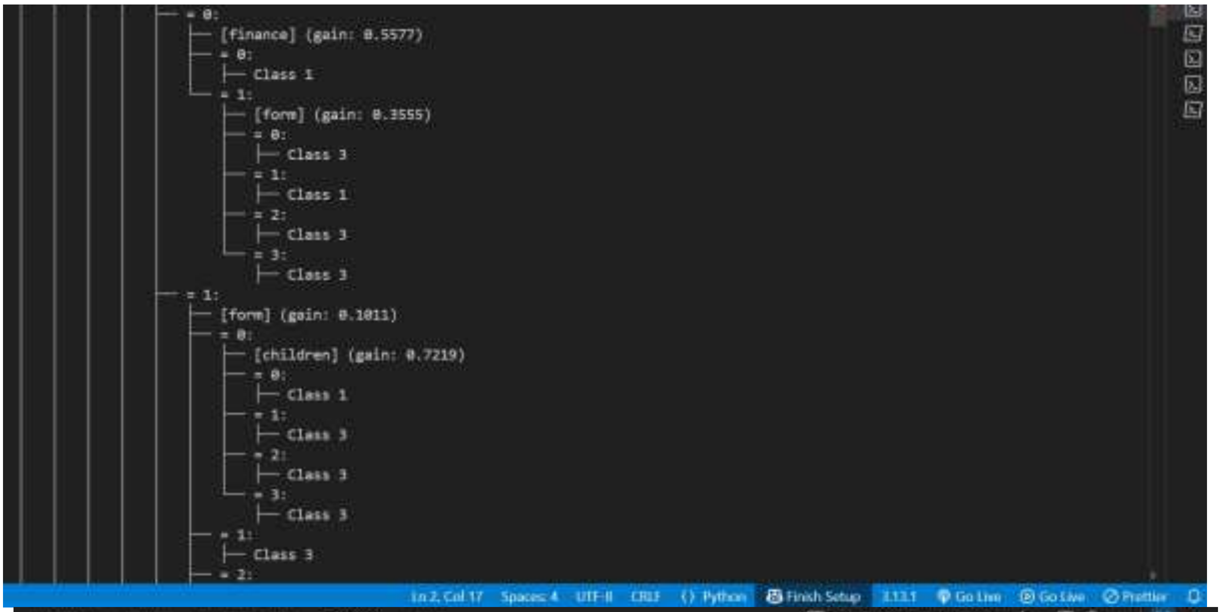


```
Constructing decision tree using training data...

🌳 Decision tree construction completed using PYTORCH!

📊 OVERALL PERFORMANCE METRICS
========================================
Accuracy:             0.9867 (98.67%)
Precision (weighted): 0.9876
Recall (weighted):    0.9867
F1-Score (weighted):  0.9872
Precision (macro):    0.7604
Recall (macro):       0.7654
F1-Score (macro):     0.7628

🌳 TREE COMPLEXITY METRICS
========================================
Maximum Depth:        7
Total Nodes:          952
Leaf Nodes:           680
Internal Nodes:       272
PS D:\CODE_ML\pytorch_implementation>
```

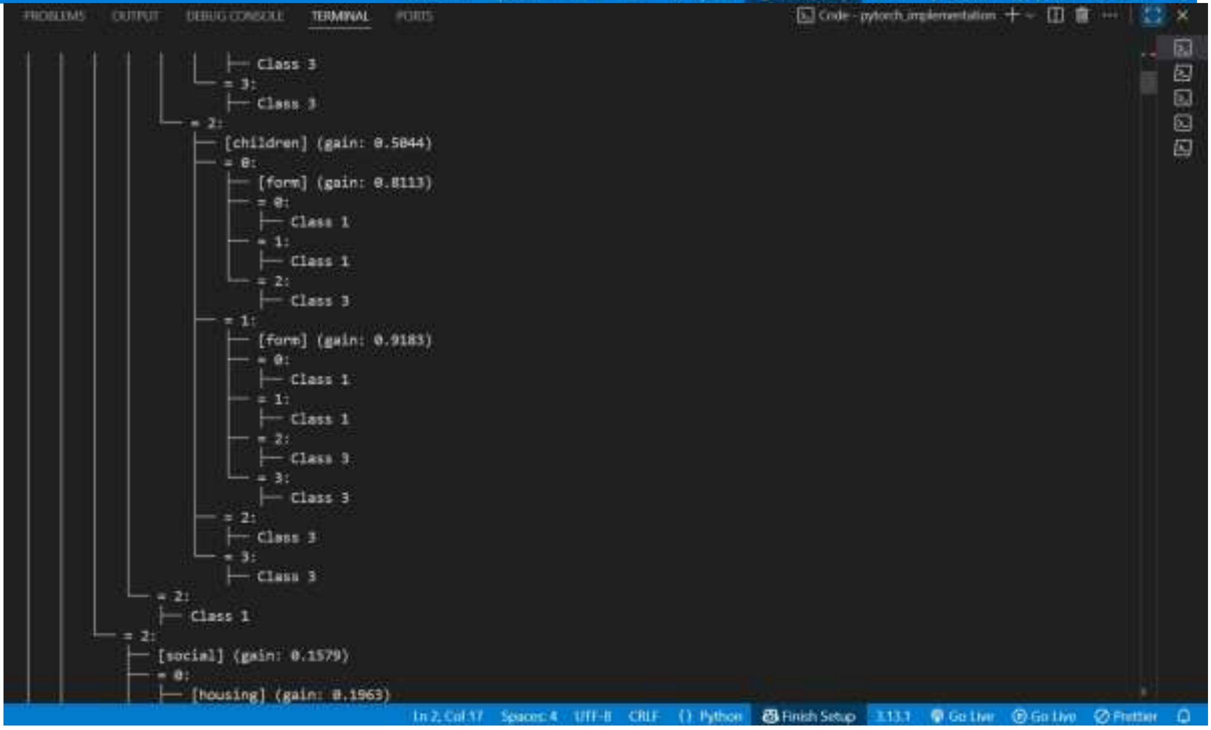
- **Accuracy:** Achieved 98.67% (PyTorch) and 98.87% (Sklearn).

- **Why:** Multi-class classification with rich features like form, finance, and children.
- **Dataset Size Impact:** Large dataset (12,960 samples) required deeper trees (Max Depth: 7) but improved generalization.
- **Number of Features:** 8 multi-valued features provided strong splits.

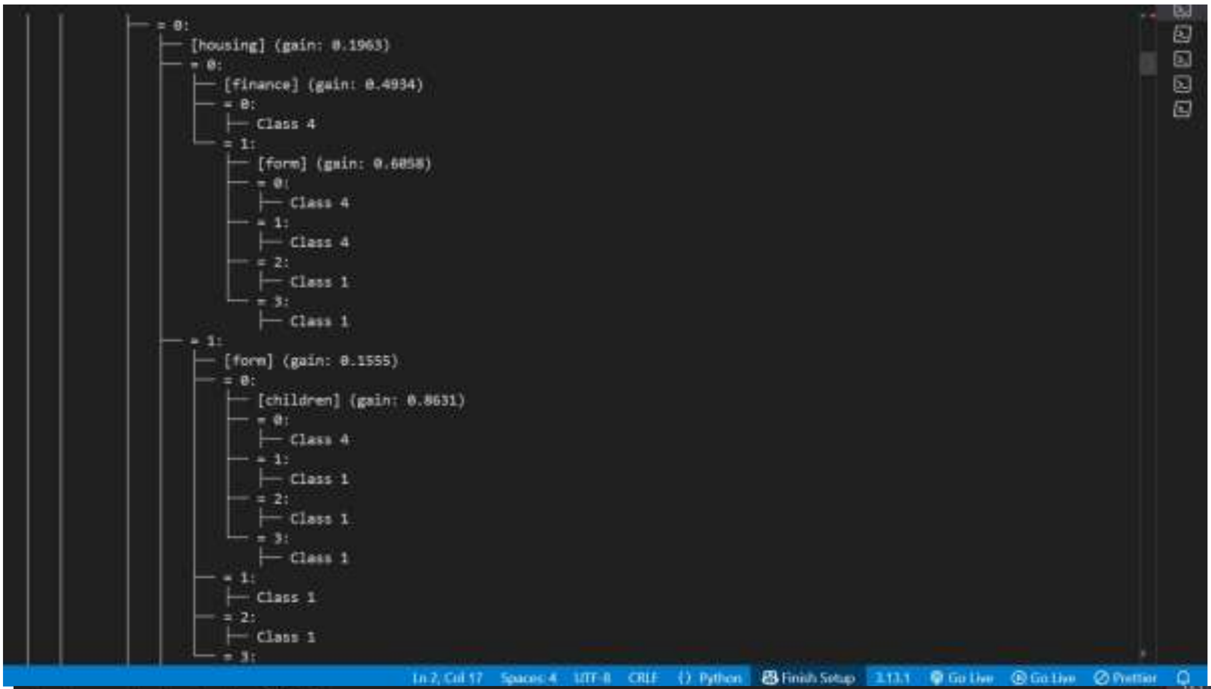
## 2.TREE VISUALIZATION



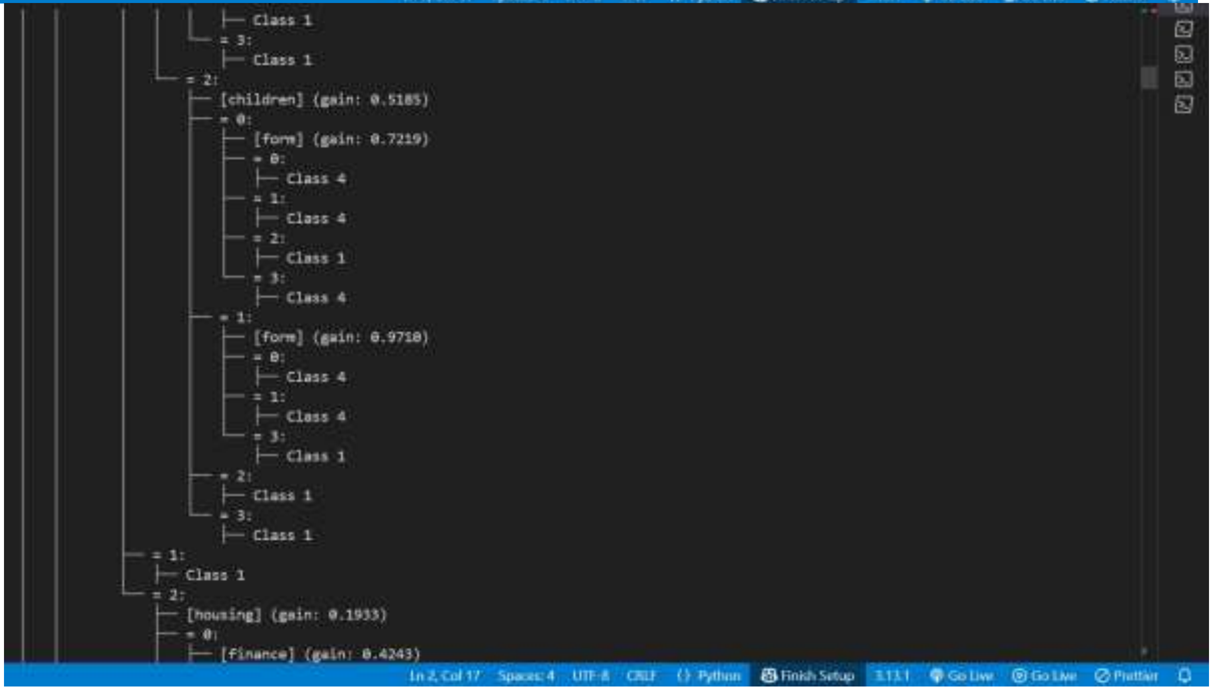
```
│   │   │   │   ├── = 0:
│   │   │   │   │   ├── [finance] (gain: 0.5577)
│   │   │   │   │   ├── = 0:
│   │   │   │   │   │   ├── Class 1
│   │   │   │   │   └── = 1:
│   │   │   │   │       ├── [form] (gain: 0.3555)
│   │   │   │   │       ├── = 0:
│   │   │   │   │       │   ├── Class 3
│   │   │   │   │       ├── = 1:
│   │   │   │   │       │   ├── Class 1
│   │   │   │   │       ├── = 2:
│   │   │   │   │       │   ├── Class 3
│   │   │   │   │       └── = 3:
│   │   │   │   │           ├── Class 3
│   │   │   │   ├── = 1:
│   │   │   │   │   ├── [form] (gain: 0.1011)
│   │   │   │   │   ├── = 0:
│   │   │   │   │   │   ├── [children] (gain: 0.7219)
│   │   │   │   │   │   ├── = 0:
│   │   │   │   │   │   │   ├── Class 1
│   │   │   │   │   │   ├── = 1:
│   │   │   │   │   │   │   ├── Class 3
│   │   │   │   │   │   ├── = 2:
│   │   │   │   │   │   │   ├── Class 3
│   │   │   │   │   │   └── = 3:
│   │   │   │   │   │       ├── Class 3
│   │   │   │   │   ├── = 1:
│   │   │   │   │   │   ├── Class 3
│   │   │   │   │   ├── = 2:
```



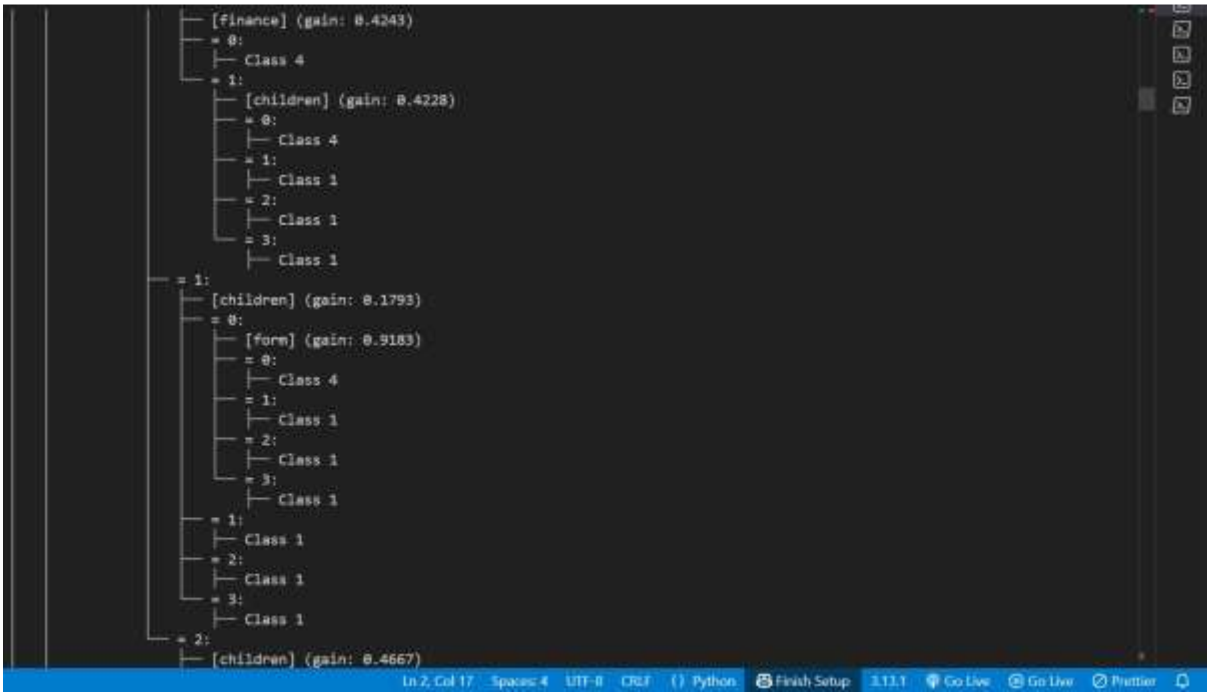
```
│   │   │   │   │   │   ├── Class 3
│   │   │   │   │   └── = 3:
│   │   │   │   │       ├── Class 3
│   │   │   │   └── = 2:
│   │   │   │       ├── [children] (gain: 0.5844)
│   │   │   │       ├── = 0:
│   │   │   │       │   ├── [form] (gain: 0.8113)
│   │   │   │       │   ├── = 0:
│   │   │   │       │   │   ├── Class 1
│   │   │   │       │   ├── = 1:
│   │   │   │       │   │   ├── Class 1
│   │   │   │       │   └── = 2:
│   │   │   │       │       ├── Class 3
│   │   │   │       ├── = 1:
│   │   │   │       │   ├── [form] (gain: 0.9183)
│   │   │   │       │   ├── = 0:
│   │   │   │       │   │   ├── Class 1
│   │   │   │       │   ├── = 1:
│   │   │   │       │   │   ├── Class 1
│   │   │   │       │   ├── = 2:
│   │   │   │       │   │   ├── Class 3
│   │   │   │       │   └── = 3:
│   │   │   │       │       ├── Class 3
│   │   │   │       ├── = 2:
│   │   │   │       │   ├── Class 3
│   │   │   │       └── = 3:
│   │   │   │           ├── Class 3
│   │   │   └── = 2:
│   │   │       ├── Class 1
│   │   └── = 2:
│   │       ├── [social] (gain: 0.1579)
│   │       ├── = 0:
│   │       │   ├── [housing] (gain: 0.1963)
```



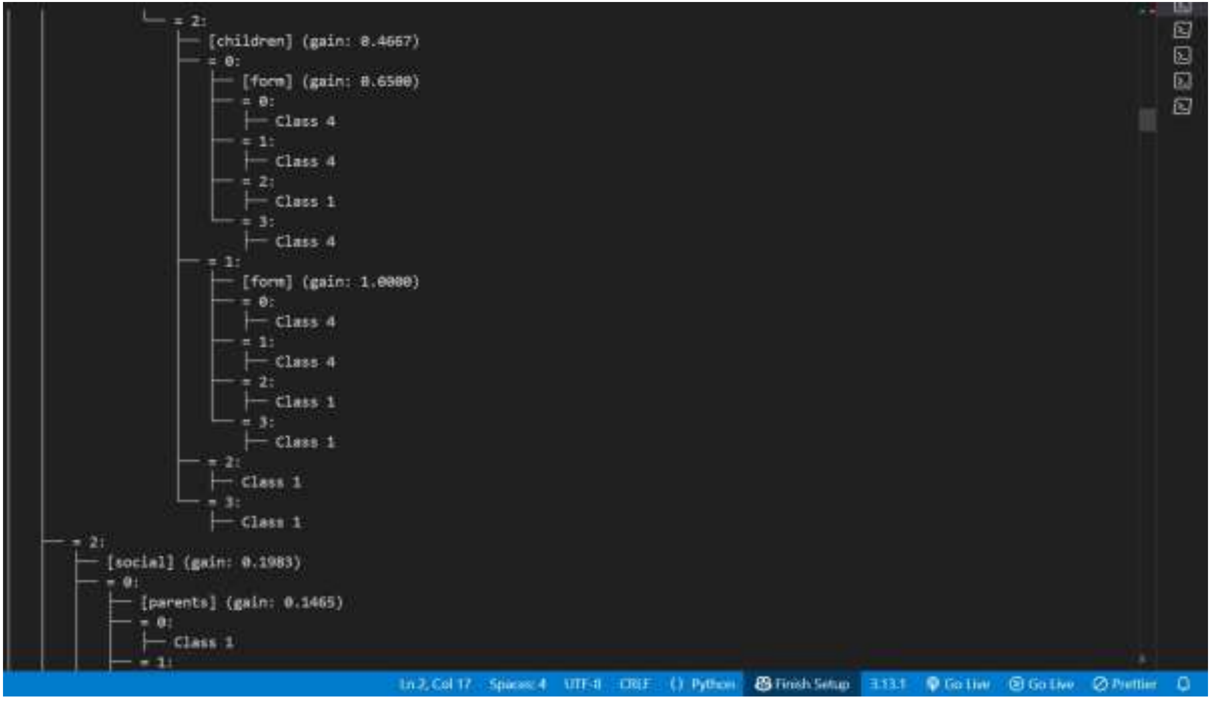
```
│   │       ├── = 0:
│   │       │   ├── [housing] (gain: 0.1963)
│   │       │   ├── = 0:
│   │       │   │   ├── [finance] (gain: 0.4934)
│   │       │   │   ├── = 0:
│   │       │   │   │   ├── Class 4
│   │       │   │   └── = 1:
│   │       │   │       ├── [form] (gain: 0.6058)
│   │       │   │       ├── = 0:
│   │       │   │       │   ├── Class 4
│   │       │   │       ├── = 1:
│   │       │   │       │   ├── Class 4
│   │       │   │       ├── = 2:
│   │       │   │       │   ├── Class 1
│   │       │   │       └── = 3:
│   │       │   │           ├── Class 1
│   │       │   ├── = 1:
│   │       │   │   ├── [form] (gain: 0.1555)
│   │       │   │   ├── = 0:
│   │       │   │   │   ├── [children] (gain: 0.8631)
│   │       │   │   │   ├── = 0:
│   │       │   │   │   │   ├── Class 4
│   │       │   │   │   ├── = 1:
│   │       │   │   │   │   ├── Class 1
│   │       │   │   │   ├── = 2:
│   │       │   │   │   │   ├── Class 1
│   │       │   │   │   └── = 3:
│   │       │   │   │       ├── Class 1
│   │       │   │   ├── = 1:
│   │       │   │   │   ├── Class 1
│   │       │   │   ├── = 2:
│   │       │   │   │   ├── Class 1
│   │       │   │   └── = 3:
```



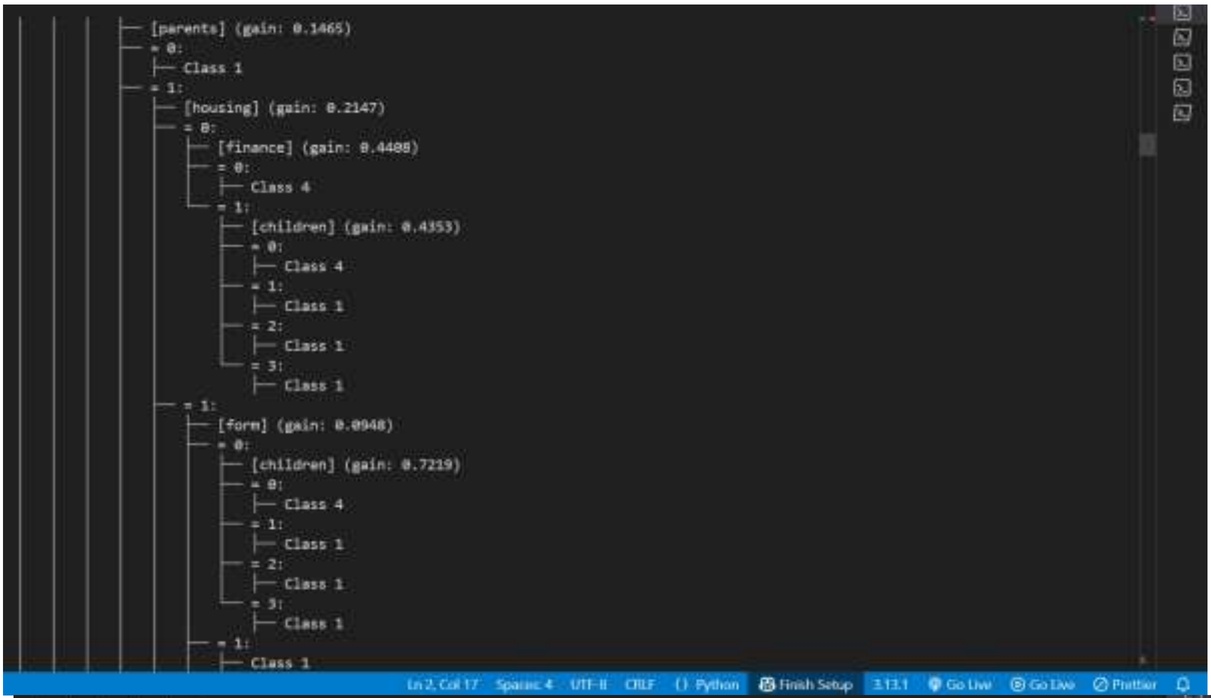
```
│   │       │   │   │   ├── Class 1
│   │       │   │   └── = 3:
│   │       │   │       ├── Class 1
│   │       │   └── = 2:
│   │       │       ├── [children] (gain: 0.5185)
│   │       │       ├── = 0:
│   │       │       │   ├── [form] (gain: 0.7219)
│   │       │       │   ├── = 0:
│   │       │       │   │   ├── Class 4
│   │       │       │   ├── = 1:
│   │       │       │   │   ├── Class 4
│   │       │       │   ├── = 2:
│   │       │       │   │   ├── Class 1
│   │       │       │   └── = 3:
│   │       │       │       ├── Class 4
│   │       │       ├── = 1:
│   │       │       │   ├── [form] (gain: 0.9710)
│   │       │       │   ├── = 0:
│   │       │       │   │   ├── Class 4
│   │       │       │   ├── = 1:
│   │       │       │   │   ├── Class 4
│   │       │       │   └── = 3:
│   │       │       │       ├── Class 1
│   │       │       ├── = 2:
│   │       │       │   ├── Class 1
│   │       │       └── = 3:
│   │       │           ├── Class 1
│   │       ├── = 1:
│   │       │   ├── Class 1
│   │       └── = 2:
│   │           ├── [housing] (gain: 0.1933)
│   │           ├── = 0:
│   │           │   ├── [finance] (gain: 0.4243)
```



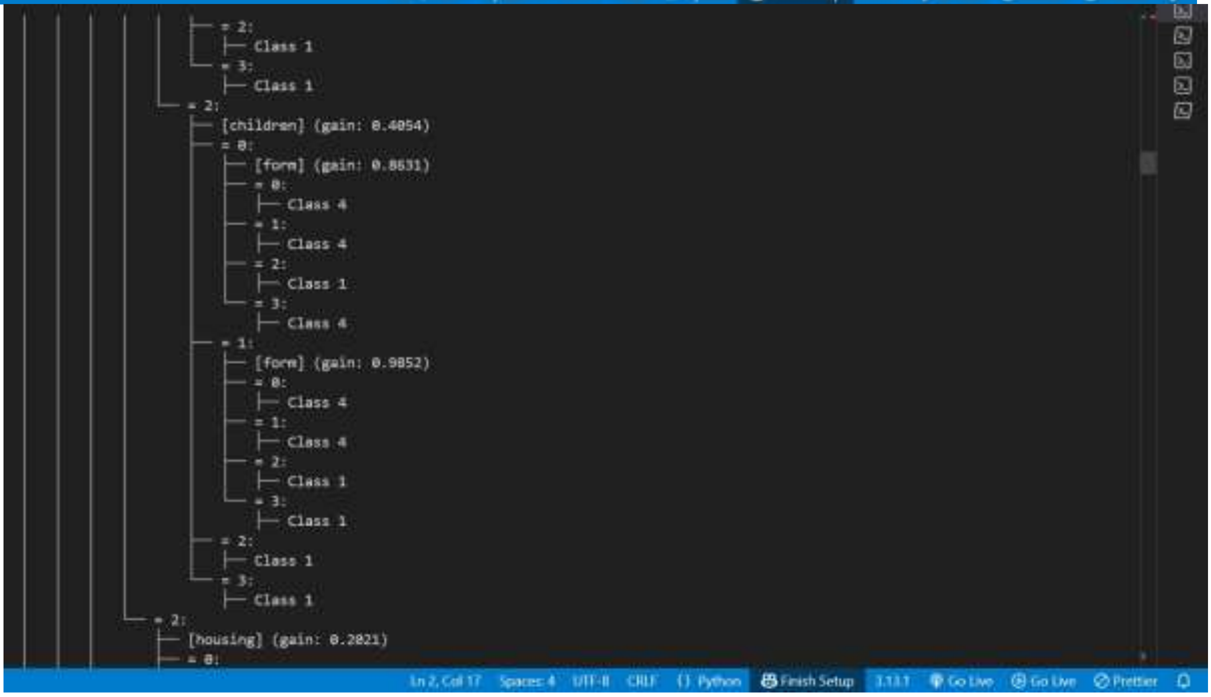
```
│   │           │   ├── [finance] (gain: 0.4243)
│   │           │   ├── = 0:
│   │           │   │   ├── Class 4
│   │           │   └── = 1:
│   │           │       ├── [children] (gain: 0.4228)
│   │           │       ├── = 0:
│   │           │       │   ├── Class 4
│   │           │       ├── = 1:
│   │           │       │   ├── Class 1
│   │           │       ├── = 2:
│   │           │       │   ├── Class 1
│   │           │       └── = 3:
│   │           │           ├── Class 1
│   │           ├── = 1:
│   │           │   ├── [children] (gain: 0.1793)
│   │           │   ├── = 0:
│   │           │   │   ├── [form] (gain: 0.9183)
│   │           │   │   ├── = 0:
│   │           │   │   │   ├── Class 4
│   │           │   │   ├── = 1:
│   │           │   │   │   ├── Class 1
│   │           │   │   ├── = 2:
│   │           │   │   │   ├── Class 1
│   │           │   │   └── = 3:
│   │           │   │       ├── Class 1
│   │           │   ├── = 1:
│   │           │   │   ├── Class 1
│   │           │   ├── = 2:
│   │           │   │   ├── Class 1
│   │           │   └── = 3:
│   │           │       ├── Class 1
│   │           └── = 2:
│   │               ├── [children] (gain: 0.4667)
```



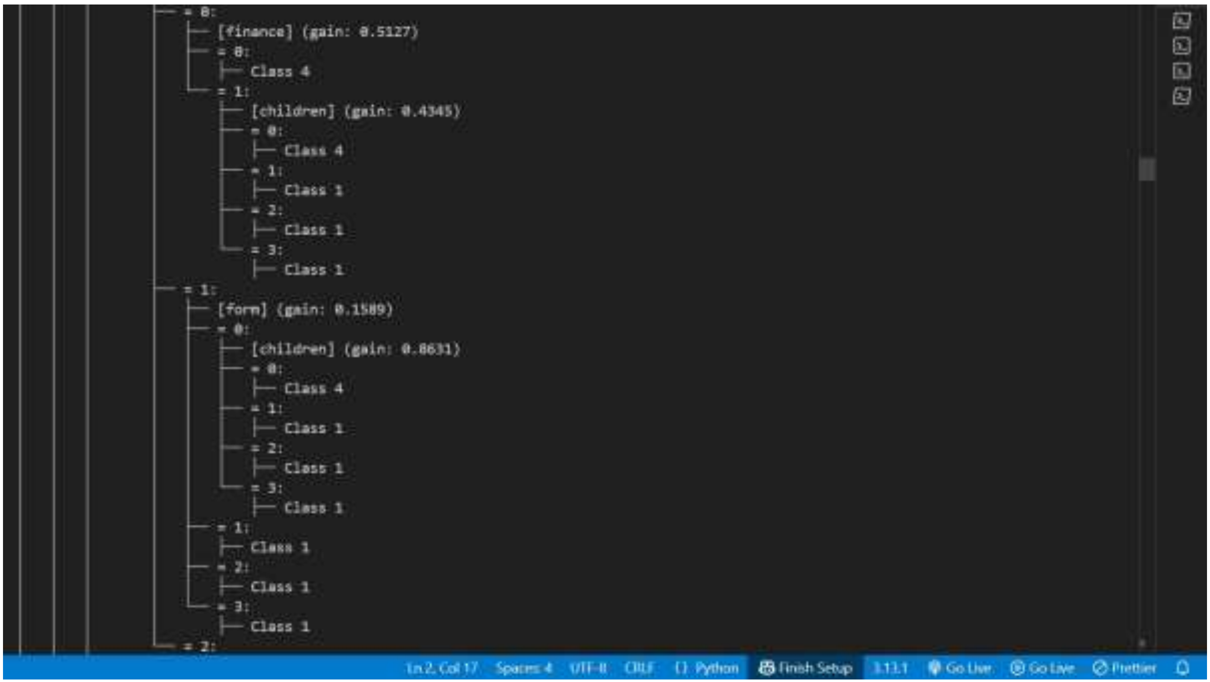
```
│   │           └── = 2:
│   │               ├── [children] (gain: 0.4667)
│   │               ├── = 0:
│   │               │   ├── [form] (gain: 0.6500)
│   │               │   ├── = 0:
│   │               │   │   ├── Class 4
│   │               │   ├── = 1:
│   │               │   │   ├── Class 4
│   │               │   ├── = 2:
│   │               │   │   ├── Class 1
│   │               │   └── = 3:
│   │               │       ├── Class 4
│   │               ├── = 1:
│   │               │   ├── [form] (gain: 1.0000)
│   │               │   ├── = 0:
│   │               │   │   ├── Class 4
│   │               │   ├── = 1:
│   │               │   │   ├── Class 4
│   │               │   ├── = 2:
│   │               │   │   ├── Class 1
│   │               │   └── = 3:
│   │               │       ├── Class 1
│   │               ├── = 2:
│   │               │   ├── Class 1
│   │               └── = 3:
│   │                   ├── Class 1
│   ├── = 2:
│   │   ├── [social] (gain: 0.1983)
│   │   ├── = 0:
│   │   │   ├── [parents] (gain: 0.1465)
│   │   │   ├── = 0:
│   │   │   │   ├── Class 1
│   │   │   ├── = 1:
```



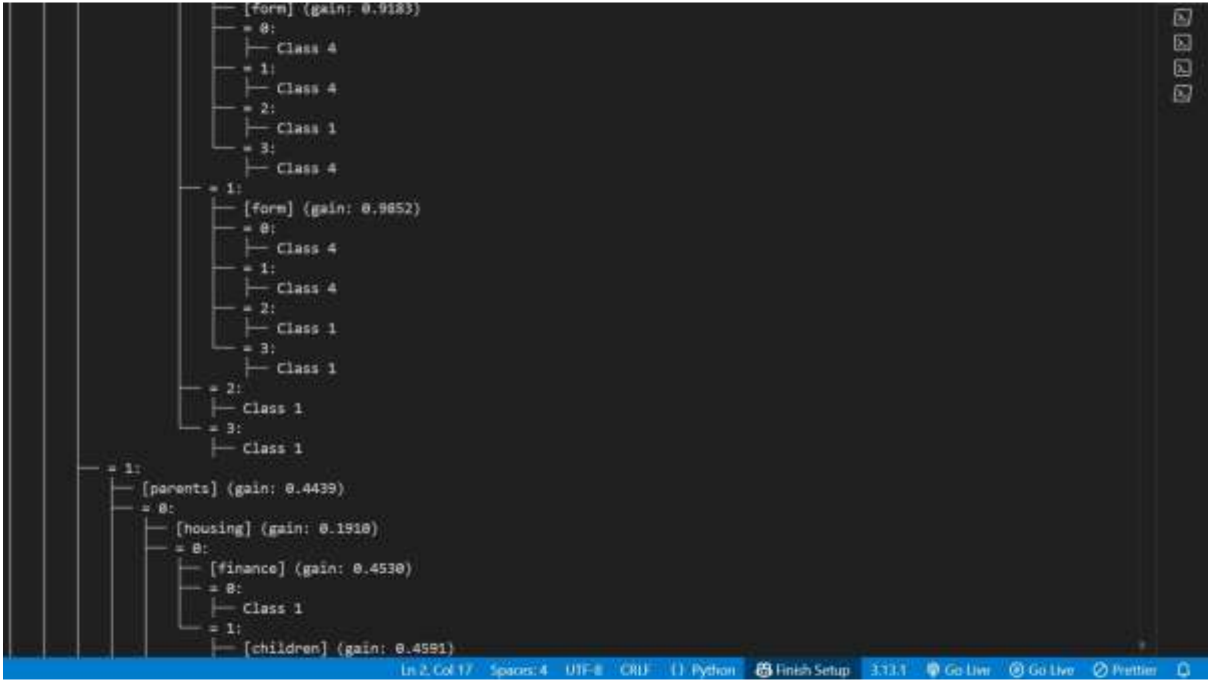
```
│   │   │   ├── [parents] (gain: 0.1465)
│   │   │   ├── = 0:
│   │   │   │   ├── Class 1
│   │   │   ├── = 1:
│   │   │   │   ├── [housing] (gain: 0.2147)
│   │   │   │   ├── = 0:
│   │   │   │   │   ├── [finance] (gain: 0.4408)
│   │   │   │   │   ├── = 0:
│   │   │   │   │   │   ├── Class 4
│   │   │   │   │   └── = 1:
│   │   │   │   │       ├── [children] (gain: 0.4353)
│   │   │   │   │       ├── = 0:
│   │   │   │   │       │   ├── Class 4
│   │   │   │   │       ├── = 1:
│   │   │   │   │       │   ├── Class 1
│   │   │   │   │       ├── = 2:
│   │   │   │   │       │   ├── Class 1
│   │   │   │   │       └── = 3:
│   │   │   │   │           ├── Class 1
│   │   │   │   ├── = 1:
│   │   │   │   │   ├── [form] (gain: 0.0948)
│   │   │   │   │   ├── = 0:
│   │   │   │   │   │   ├── [children] (gain: 0.7219)
│   │   │   │   │   │   ├── = 0:
│   │   │   │   │   │   │   ├── Class 4
│   │   │   │   │   │   ├── = 1:
│   │   │   │   │   │   │   ├── Class 1
│   │   │   │   │   │   ├── = 2:
│   │   │   │   │   │   │   ├── Class 1
│   │   │   │   │   │   └── = 3:
│   │   │   │   │   │       ├── Class 1
│   │   │   │   │   ├── = 1:
│   │   │   │   │   │   ├── Class 1
```



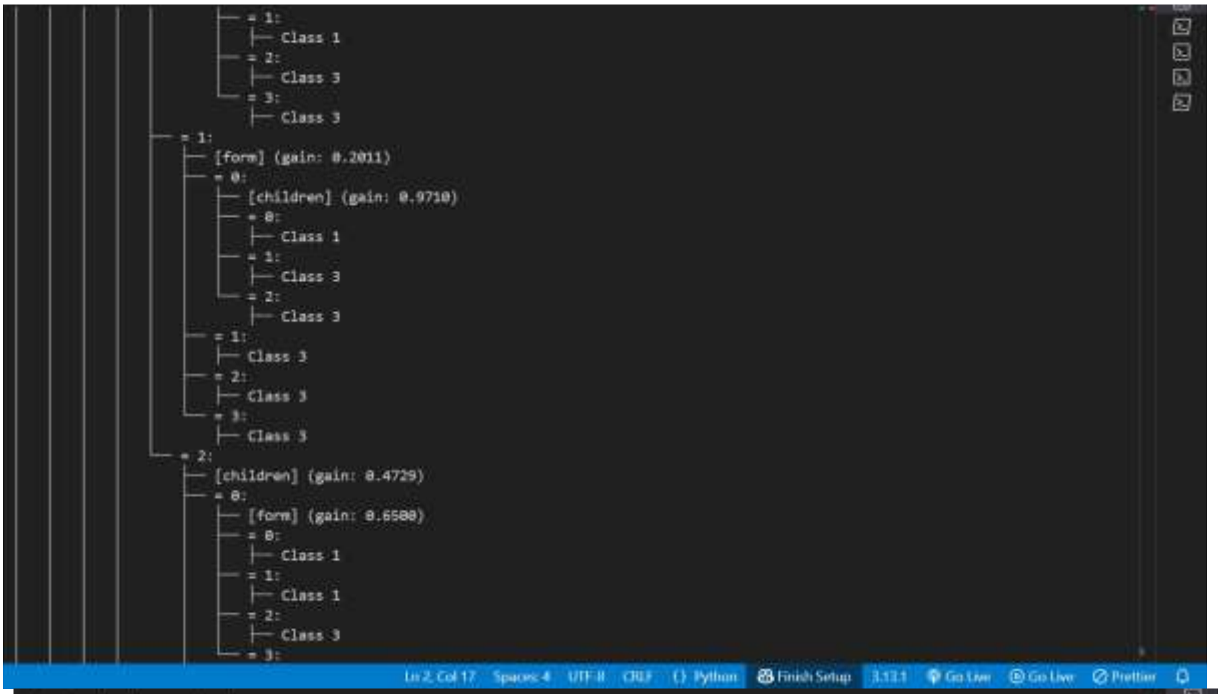
```
│   │   │   │   │   ├── = 2:
│   │   │   │   │   │   ├── Class 1
│   │   │   │   │   └── = 3:
│   │   │   │   │       ├── Class 1
│   │   │   │   └── = 2:
│   │   │   │       ├── [children] (gain: 0.4054)
│   │   │   │       ├── = 0:
│   │   │   │       │   ├── [form] (gain: 0.8631)
│   │   │   │       │   ├── = 0:
│   │   │   │       │   │   ├── Class 4
│   │   │   │       │   ├── = 1:
│   │   │   │       │   │   ├── Class 4
│   │   │   │       │   ├── = 2:
│   │   │   │       │   │   ├── Class 1
│   │   │   │       │   └── = 3:
│   │   │   │       │       ├── Class 4
│   │   │   │       ├── = 1:
│   │   │   │       │   ├── [form] (gain: 0.9852)
│   │   │   │       │   ├── = 0:
│   │   │   │       │   │   ├── Class 4
│   │   │   │       │   ├── = 1:
│   │   │   │       │   │   ├── Class 4
│   │   │   │       │   ├── = 2:
│   │   │   │       │   │   ├── Class 1
│   │   │   │       │   └── = 3:
│   │   │   │       │       ├── Class 1
│   │   │   │       ├── = 2:
│   │   │   │       │   ├── Class 1
│   │   │   │       └── = 3:
│   │   │   │           ├── Class 1
│   │   │   └── = 2:
│   │   │       ├── [housing] (gain: 0.2821)
│   │   │       ├── = 0:
```



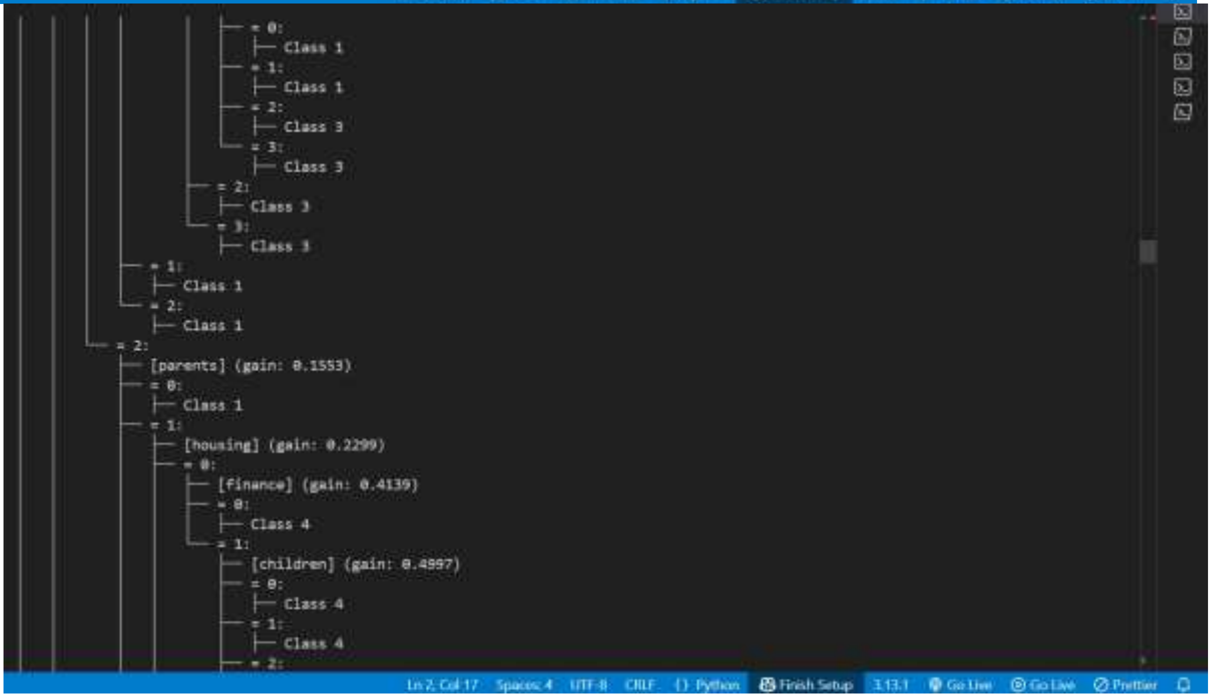
```
│   │   │       ├── = 0:
│   │   │       │   ├── [finance] (gain: 0.5127)
│   │   │       │   ├── = 0:
│   │   │       │   │   ├── Class 4
│   │   │       │   └── = 1:
│   │   │       │       ├── [children] (gain: 0.4345)
│   │   │       │       ├── = 0:
│   │   │       │       │   ├── Class 4
│   │   │       │       ├── = 1:
│   │   │       │       │   ├── Class 1
│   │   │       │       ├── = 2:
│   │   │       │       │   ├── Class 1
│   │   │       │       └── = 3:
│   │   │       │           ├── Class 1
│   │   │       ├── = 1:
│   │   │       │   ├── [form] (gain: 0.1589)
│   │   │       │   ├── = 0:
│   │   │       │   │   ├── [children] (gain: 0.8631)
│   │   │       │   │   ├── = 0:
│   │   │       │   │   │   ├── Class 4
│   │   │       │   │   ├── = 1:
│   │   │       │   │   │   ├── Class 1
│   │   │       │   │   ├── = 2:
│   │   │       │   │   │   ├── Class 1
│   │   │       │   │   └── = 3:
│   │   │       │   │       ├── Class 1
│   │   │       │   ├── = 1:
│   │   │       │   │   ├── Class 1
│   │   │       │   ├── = 2:
│   │   │       │   │   ├── Class 1
│   │   │       │   └── = 3:
│   │   │       │       ├── Class 1
│   │   │       └── = 2:
```



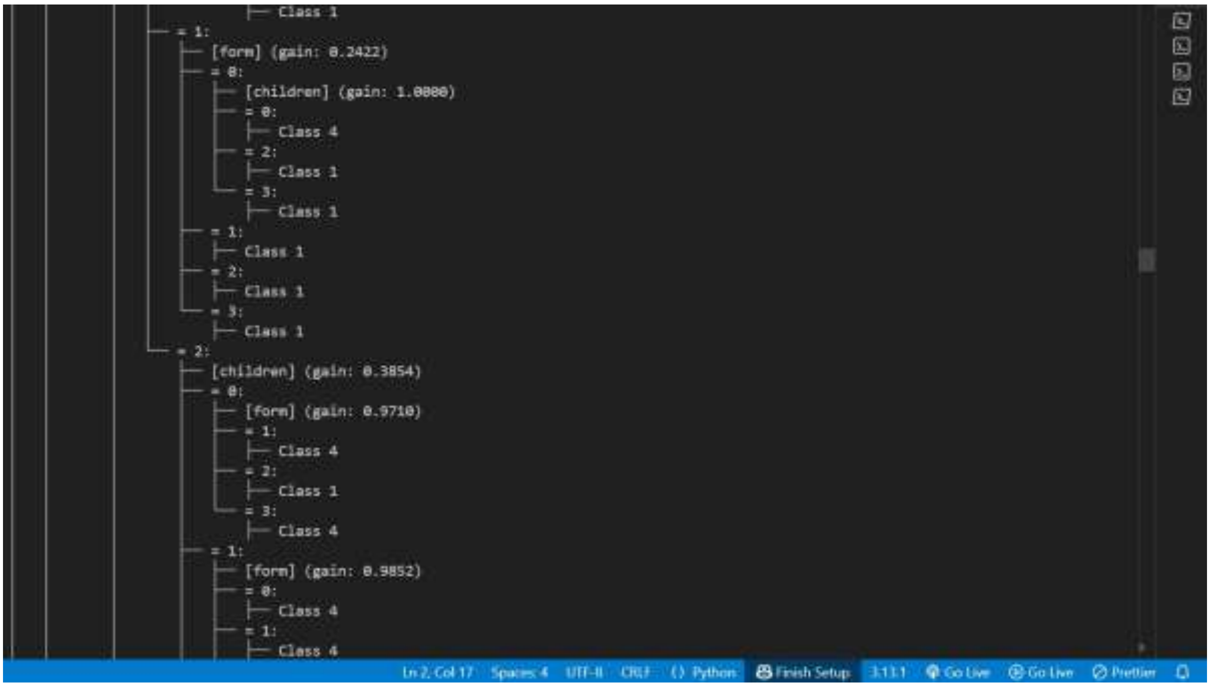
```
│   │   │           │   ├── [form] (gain: 0.9183)
│   │   │           │   ├── = 0:
│   │   │           │   │   ├── Class 4
│   │   │           │   ├── = 1:
│   │   │           │   │   ├── Class 4
│   │   │           │   ├── = 2:
│   │   │           │   │   ├── Class 1
│   │   │           │   └── = 3:
│   │   │           │       ├── Class 4
│   │   │           ├── = 1:
│   │   │           │   ├── [form] (gain: 0.9852)
│   │   │           │   ├── = 0:
│   │   │           │   │   ├── Class 4
│   │   │           │   ├── = 1:
│   │   │           │   │   ├── Class 4
│   │   │           │   ├── = 2:
│   │   │           │   │   ├── Class 1
│   │   │           │   └── = 3:
│   │   │           │       ├── Class 1
│   │   │           ├── = 2:
│   │   │           │   ├── Class 1
│   │   │           └── = 3:
│   │   │               ├── Class 1
│   │   ├── = 1:
│   │   │   ├── [parents] (gain: 0.4439)
│   │   │   ├── = 0:
│   │   │   │   ├── [housing] (gain: 0.1910)
│   │   │   │   ├── = 0:
│   │   │   │   │   ├── [finance] (gain: 0.4530)
│   │   │   │   │   ├── = 0:
│   │   │   │   │   │   ├── Class 1
│   │   │   │   │   └── = 1:
│   │   │   │   │       ├── [children] (gain: 0.4591)
```



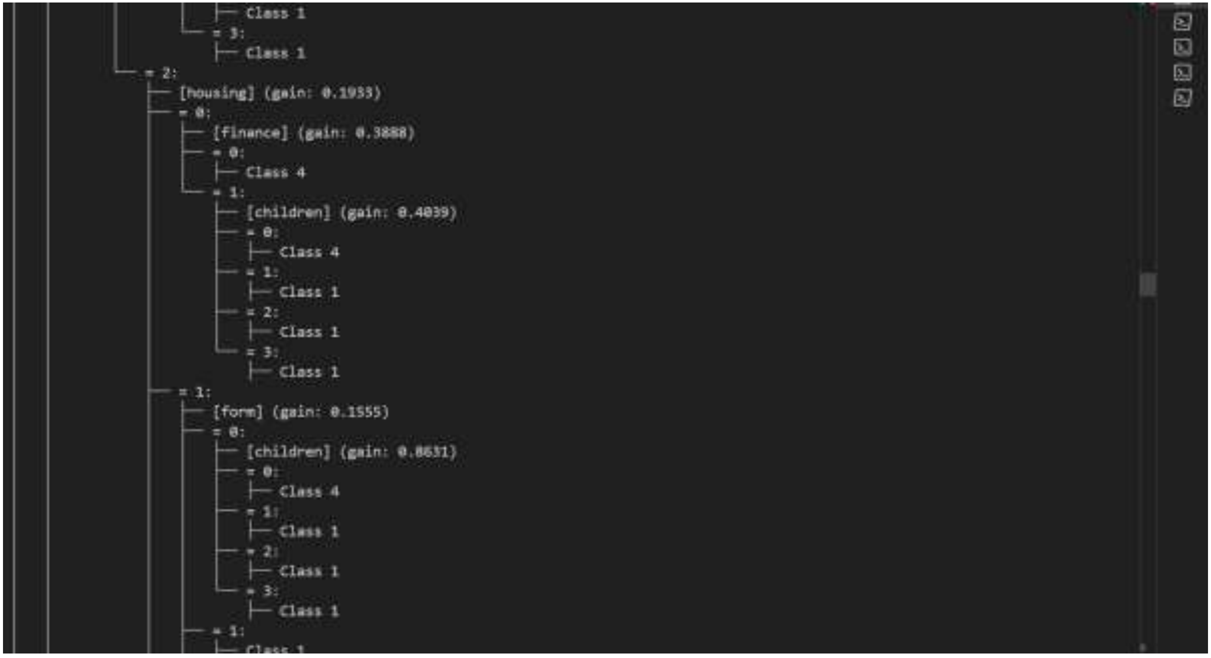
```
│   │   │   │   │       ├── = 1:
│   │   │   │   │       │   ├── Class 1
│   │   │   │   │       ├── = 2:
│   │   │   │   │       │   ├── Class 3
│   │   │   │   │       └── = 3:
│   │   │   │   │           ├── Class 3
│   │   │   │   ├── = 1:
│   │   │   │   │   ├── [form] (gain: 0.2011)
│   │   │   │   │   ├── = 0:
│   │   │   │   │   │   ├── [children] (gain: 0.9710)
│   │   │   │   │   │   ├── = 0:
│   │   │   │   │   │   │   ├── Class 1
│   │   │   │   │   │   ├── = 1:
│   │   │   │   │   │   │   ├── Class 3
│   │   │   │   │   │   └── = 2:
│   │   │   │   │   │       ├── Class 3
│   │   │   │   │   ├── = 1:
│   │   │   │   │   │   ├── Class 3
│   │   │   │   │   ├── = 2:
│   │   │   │   │   │   ├── Class 3
│   │   │   │   │   └── = 3:
│   │   │   │   │       ├── Class 3
│   │   │   │   └── = 2:
│   │   │   │       ├── [children] (gain: 0.4729)
│   │   │   │       ├── = 0:
│   │   │   │       │   ├── [form] (gain: 0.6500)
│   │   │   │       │   ├── = 0:
│   │   │   │       │   │   ├── Class 1
│   │   │   │       │   ├── = 1:
│   │   │   │       │   │   ├── Class 1
│   │   │   │       │   ├── = 2:
│   │   │   │       │   │   ├── Class 3
│   │   │   │       │   └── = 3:
```



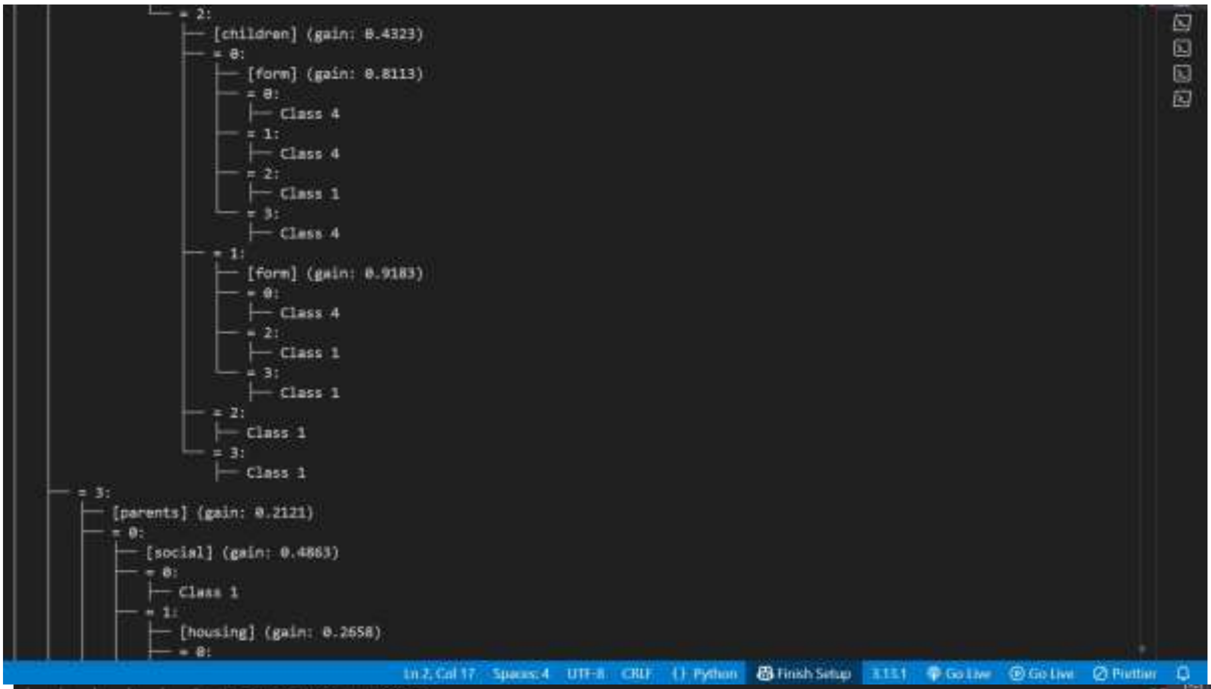
```
│   │   │   │       │   ├── = 0:
│   │   │   │       │   │   ├── Class 1
│   │   │   │       │   ├── = 1:
│   │   │   │       │   │   ├── Class 1
│   │   │   │       │   ├── = 2:
│   │   │   │       │   │   ├── Class 3
│   │   │   │       │   └── = 3:
│   │   │   │       │       ├── Class 3
│   │   │   │       ├── = 2:
│   │   │   │       │   ├── Class 3
│   │   │   │       └── = 3:
│   │   │   │           ├── Class 3
│   │   │   ├── = 1:
│   │   │   │   ├── Class 1
│   │   │   └── = 2:
│   │   │       ├── Class 1
│   │   └── = 2:
│   │       ├── [parents] (gain: 0.1553)
│   │       ├── = 0:
│   │       │   ├── Class 1
│   │       ├── = 1:
│   │       │   ├── [housing] (gain: 0.2299)
│   │       │   ├── = 0:
│   │       │   │   ├── [finance] (gain: 0.4139)
│   │       │   │   ├── = 0:
│   │       │   │   │   ├── Class 4
│   │       │   │   └── = 1:
│   │       │   │       ├── [children] (gain: 0.4997)
│   │       │   │       ├── = 0:
│   │       │   │       │   ├── Class 4
│   │       │   │       ├── = 1:
│   │       │   │       │   ├── Class 4
│   │       │   │       ├── = 2:
```



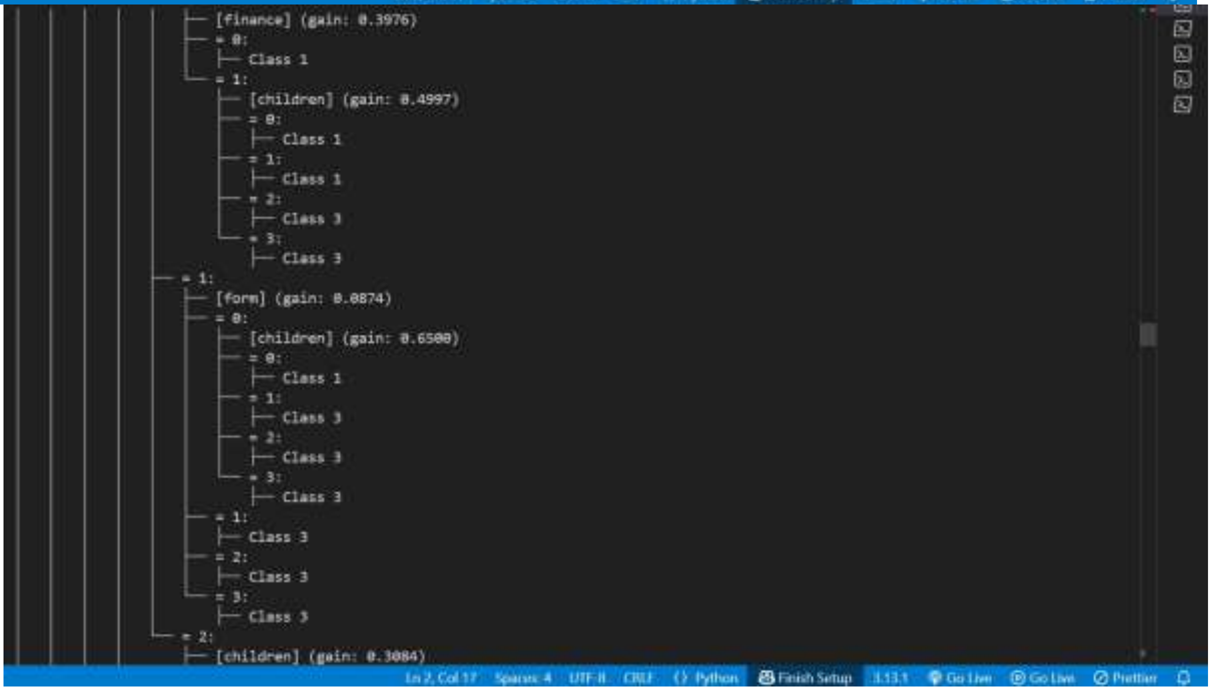
```
│   │       │   │           ├── Class 1
│   │       │   ├── = 1:
│   │       │   │   ├── [form] (gain: 0.2422)
│   │       │   │   ├── = 0:
│   │       │   │   │   ├── [children] (gain: 1.0000)
│   │       │   │   │   ├── = 0:
│   │       │   │   │   │   ├── Class 4
│   │       │   │   │   ├── = 2:
│   │       │   │   │   │   ├── Class 1
│   │       │   │   │   └── = 3:
│   │       │   │   │       ├── Class 1
│   │       │   │   ├── = 1:
│   │       │   │   │   ├── Class 1
│   │       │   │   ├── = 2:
│   │       │   │   │   ├── Class 1
│   │       │   │   └── = 3:
│   │       │   │       ├── Class 1
│   │       │   └── = 2:
│   │       │       ├── [children] (gain: 0.3854)
│   │       │       ├── = 0:
│   │       │       │   ├── [form] (gain: 0.9710)
│   │       │       │   ├── = 1:
│   │       │       │   │   ├── Class 4
│   │       │       │   ├── = 2:
│   │       │       │   │   ├── Class 1
│   │       │       │   └── = 3:
│   │       │       │       ├── Class 4
│   │       │       ├── = 1:
│   │       │       │   ├── [form] (gain: 0.9852)
│   │       │       │   ├── = 0:
│   │       │       │   │   ├── Class 4
│   │       │       │   ├── = 1:
│   │       │       │   │   ├── Class 4
```



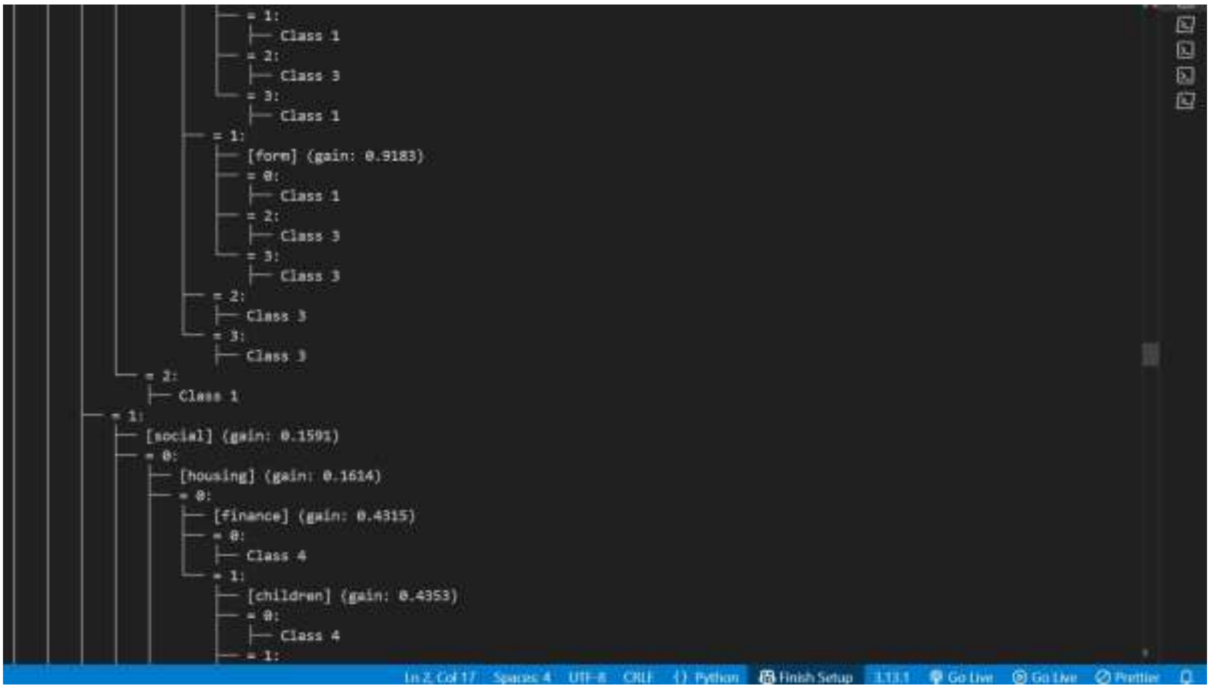
```
│   │       │       │   ├── Class 1
│   │       │       └── = 3:
│   │       │           ├── Class 1
│   │       └── = 2:
│   │           ├── [housing] (gain: 0.1933)
│   │           ├── = 0:
│   │           │   ├── [finance] (gain: 0.3888)
│   │           │   ├── = 0:
│   │           │   │   ├── Class 4
│   │           │   └── = 1:
│   │           │       ├── [children] (gain: 0.4039)
│   │           │       ├── = 0:
│   │           │       │   ├── Class 4
│   │           │       ├── = 1:
│   │           │       │   ├── Class 1
│   │           │       ├── = 2:
│   │           │       │   ├── Class 1
│   │           │       └── = 3:
│   │           │           ├── Class 1
│   │           ├── = 1:
│   │           │   ├── [form] (gain: 0.1555)
│   │           │   ├── = 0:
│   │           │   │   ├── [children] (gain: 0.8631)
│   │           │   │   ├── = 0:
│   │           │   │   │   ├── Class 4
│   │           │   │   ├── = 1:
│   │           │   │   │   ├── Class 1
│   │           │   │   ├── = 2:
│   │           │   │   │   ├── Class 1
│   │           │   │   └── = 3:
│   │           │   │       ├── Class 1
│   │           │   ├── = 1:
│   │           │   │   ├── Class 1
```



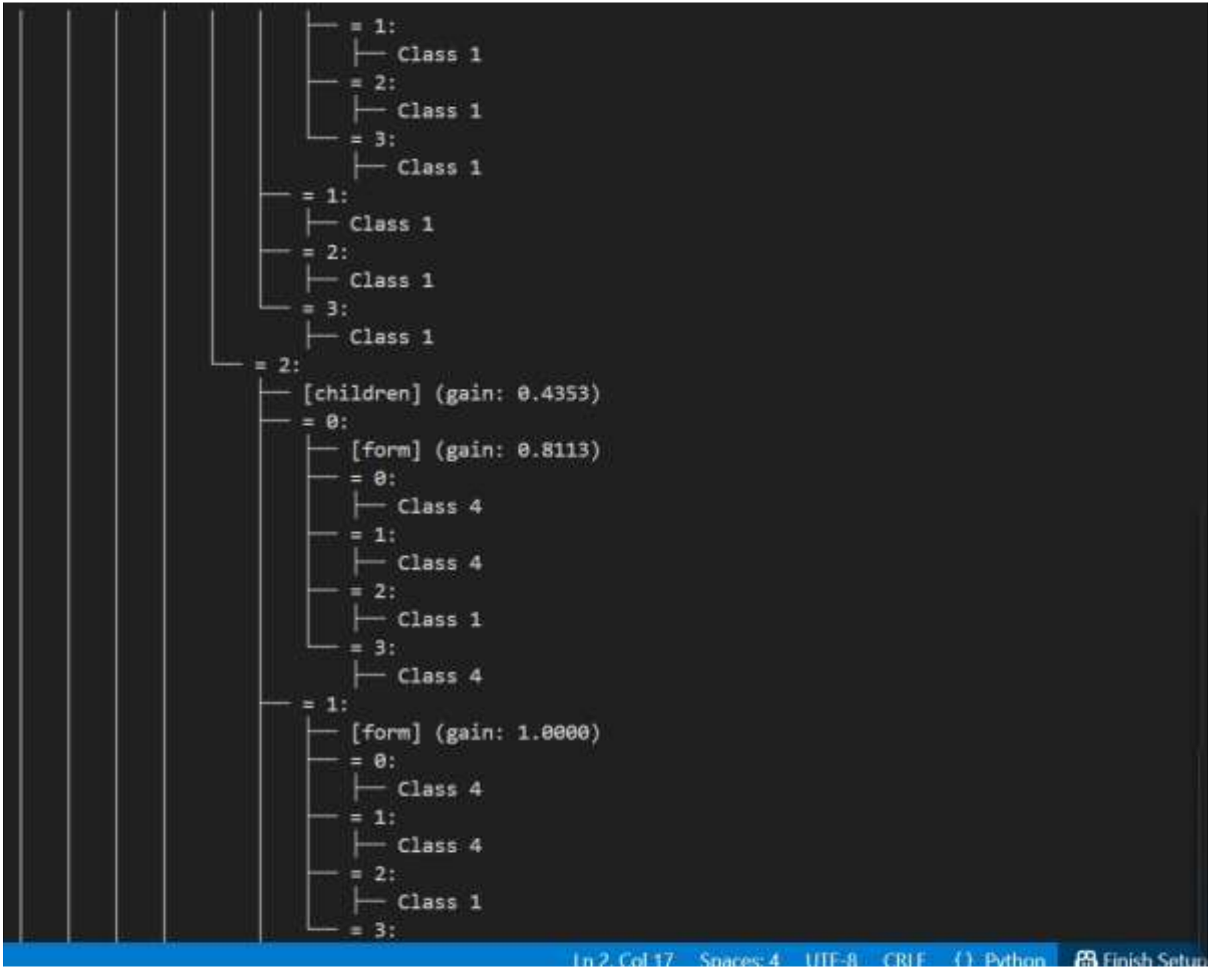
```
│   │           └── = 2:
│   │               ├── [children] (gain: 0.4323)
│   │               ├── = 0:
│   │               │   ├── [form] (gain: 0.8113)
│   │               │   ├── = 0:
│   │               │   │   ├── Class 4
│   │               │   ├── = 1:
│   │               │   │   ├── Class 4
│   │               │   ├── = 2:
│   │               │   │   ├── Class 1
│   │               │   └── = 3:
│   │               │       ├── Class 4
│   │               ├── = 1:
│   │               │   ├── [form] (gain: 0.9183)
│   │               │   ├── = 0:
│   │               │   │   ├── Class 4
│   │               │   ├── = 2:
│   │               │   │   ├── Class 1
│   │               │   └── = 3:
│   │               │       ├── Class 1
│   │               ├── = 2:
│   │               │   ├── Class 1
│   │               └── = 3:
│   │                   ├── Class 1
│   ├── = 3:
│   │   ├── [parents] (gain: 0.2121)
│   │   ├── = 0:
│   │   │   ├── [social] (gain: 0.4863)
│   │   │   ├── = 0:
│   │   │   │   ├── Class 1
│   │   │   ├── = 1:
│   │   │   │   ├── [housing] (gain: 0.2658)
│   │   │   │   ├── = 0:
```



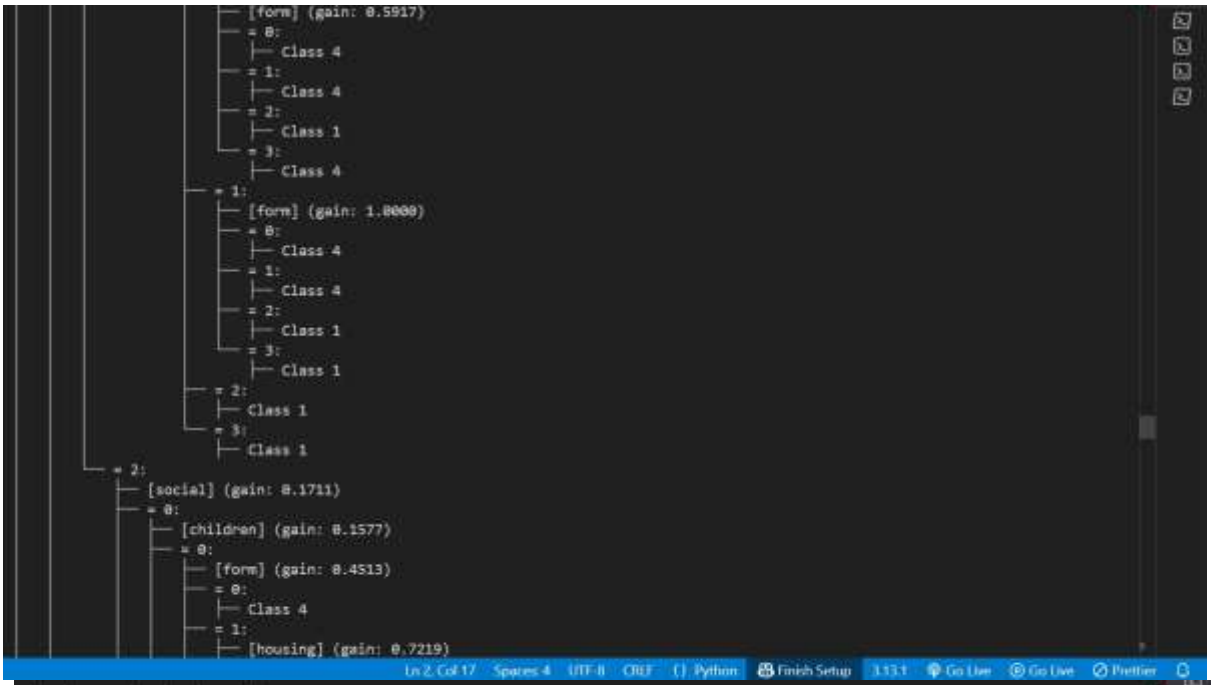
```
│   │   │   │   │   ├── [finance] (gain: 0.3976)
│   │   │   │   │   ├── = 0:
│   │   │   │   │   │   ├── Class 1
│   │   │   │   │   └── = 1:
│   │   │   │   │       ├── [children] (gain: 0.4997)
│   │   │   │   │       ├── = 0:
│   │   │   │   │       │   ├── Class 1
│   │   │   │   │       ├── = 1:
│   │   │   │   │       │   ├── Class 1
│   │   │   │   │       ├── = 2:
│   │   │   │   │       │   ├── Class 3
│   │   │   │   │       └── = 3:
│   │   │   │   │           ├── Class 3
│   │   │   │   ├── = 1:
│   │   │   │   │   ├── [form] (gain: 0.0874)
│   │   │   │   │   ├── = 0:
│   │   │   │   │   │   ├── [children] (gain: 0.6500)
│   │   │   │   │   │   ├── = 0:
│   │   │   │   │   │   │   ├── Class 1
│   │   │   │   │   │   ├── = 1:
│   │   │   │   │   │   │   ├── Class 3
│   │   │   │   │   │   ├── = 2:
│   │   │   │   │   │   │   ├── Class 3
│   │   │   │   │   │   └── = 3:
│   │   │   │   │   │       ├── Class 3
│   │   │   │   │   ├── = 1:
│   │   │   │   │   │   ├── Class 3
│   │   │   │   │   ├── = 2:
│   │   │   │   │   │   ├── Class 3
│   │   │   │   │   └── = 3:
│   │   │   │   │       ├── Class 3
│   │   │   │   └── = 2:
│   │   │   │       ├── [children] (gain: 0.3084)
```



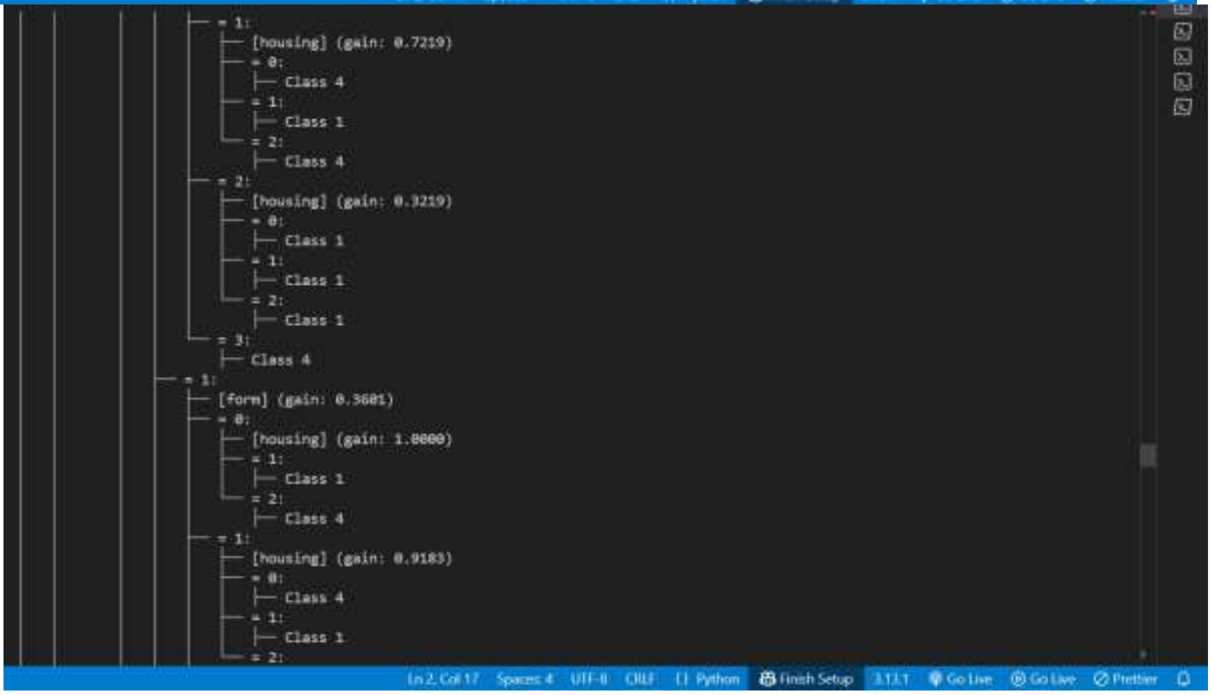
```
│   │   │   │       │   ├── = 1:
│   │   │   │       │   │   ├── Class 1
│   │   │   │       │   ├── = 2:
│   │   │   │       │   │   ├── Class 3
│   │   │   │       │   └── = 3:
│   │   │   │       │       ├── Class 1
│   │   │   │       ├── = 1:
│   │   │   │       │   ├── [form] (gain: 0.9183)
│   │   │   │       │   ├── = 0:
│   │   │   │       │   │   ├── Class 1
│   │   │   │       │   ├── = 2:
│   │   │   │       │   │   ├── Class 3
│   │   │   │       │   └── = 3:
│   │   │   │       │       ├── Class 3
│   │   │   │       ├── = 2:
│   │   │   │       │   ├── Class 3
│   │   │   │       └── = 3:
│   │   │   │           ├── Class 3
│   │   │   └── = 2:
│   │   │       ├── Class 1
│   │   ├── = 1:
│   │   │   ├── [social] (gain: 0.1591)
│   │   │   ├── = 0:
│   │   │   │   ├── [housing] (gain: 0.1614)
│   │   │   │   ├── = 0:
│   │   │   │   │   ├── [finance] (gain: 0.4315)
│   │   │   │   │   ├── = 0:
│   │   │   │   │   │   ├── Class 4
│   │   │   │   │   └── = 1:
│   │   │   │   │       ├── [children] (gain: 0.4353)
│   │   │   │   │       ├── = 0:
│   │   │   │   │       │   ├── Class 4
│   │   │   │   │       ├── = 1:
```



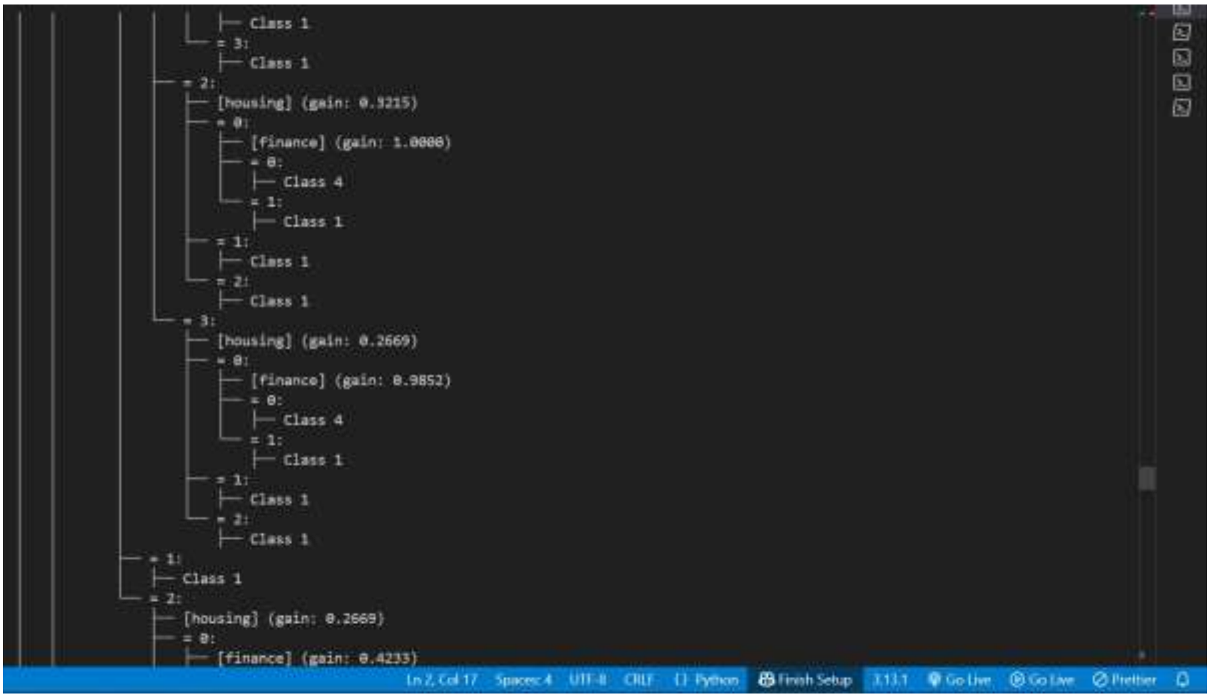
```
│   │   │   │   │   │   ├── = 1:
│   │   │   │   │   │   │   ├── Class 1
│   │   │   │   │   │   ├── = 2:
│   │   │   │   │   │   │   ├── Class 1
│   │   │   │   │   │   └── = 3:
│   │   │   │   │   │       ├── Class 1
│   │   │   │   │   ├── = 1:
│   │   │   │   │   │   ├── Class 1
│   │   │   │   │   ├── = 2:
│   │   │   │   │   │   ├── Class 1
│   │   │   │   │   └── = 3:
│   │   │   │   │       ├── Class 1
│   │   │   │   └── = 2:
│   │   │   │       ├── [children] (gain: 0.4353)
│   │   │   │       ├── = 0:
│   │   │   │       │   ├── [form] (gain: 0.8113)
│   │   │   │       │   ├── = 0:
│   │   │   │       │   │   ├── Class 4
│   │   │   │       │   ├── = 1:
│   │   │   │       │   │   ├── Class 4
│   │   │   │       │   ├── = 2:
│   │   │   │       │   │   ├── Class 1
│   │   │   │       │   └── = 3:
│   │   │   │       │       ├── Class 4
│   │   │   │       ├── = 1:
│   │   │   │       │   ├── [form] (gain: 1.0000)
│   │   │   │       │   ├── = 0:
│   │   │   │       │   │   ├── Class 4
│   │   │   │       │   ├── = 1:
│   │   │   │       │   │   ├── Class 4
│   │   │   │       │   ├── = 2:
│   │   │   │       │   │   ├── Class 1
│   │   │   │       │   └── = 3:
```



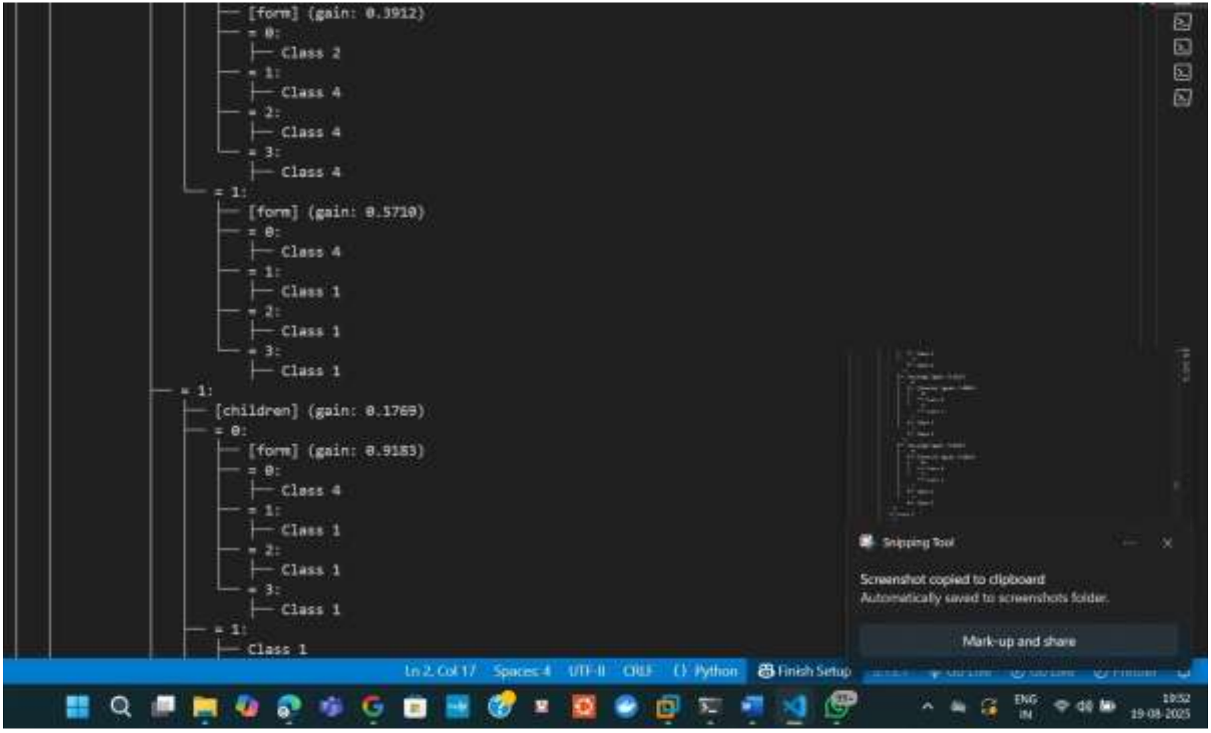
```
│   │   │           │   ├── [form] (gain: 0.5917)
│   │   │           │   ├── = 0:
│   │   │           │   │   ├── Class 4
│   │   │           │   ├── = 1:
│   │   │           │   │   ├── Class 4
│   │   │           │   ├── = 2:
│   │   │           │   │   ├── Class 1
│   │   │           │   └── = 3:
│   │   │           │       ├── Class 4
│   │   │           ├── = 1:
│   │   │           │   ├── [form] (gain: 1.0000)
│   │   │           │   ├── = 0:
│   │   │           │   │   ├── Class 4
│   │   │           │   ├── = 1:
│   │   │           │   │   ├── Class 4
│   │   │           │   ├── = 2:
│   │   │           │   │   ├── Class 1
│   │   │           │   └── = 3:
│   │   │           │       ├── Class 1
│   │   │           ├── = 2:
│   │   │           │   ├── Class 1
│   │   │           └── = 3:
│   │   │               ├── Class 1
│   │   └── = 2:
│   │       ├── [social] (gain: 0.1711)
│   │       ├── = 0:
│   │       │   ├── [children] (gain: 0.1577)
│   │       │   ├── = 0:
│   │       │   │   ├── [form] (gain: 0.4513)
│   │       │   │   ├── = 0:
│   │       │   │   │   ├── Class 4
│   │       │   │   ├── = 1:
│   │       │   │   │   ├── [housing] (gain: 0.7219)
```



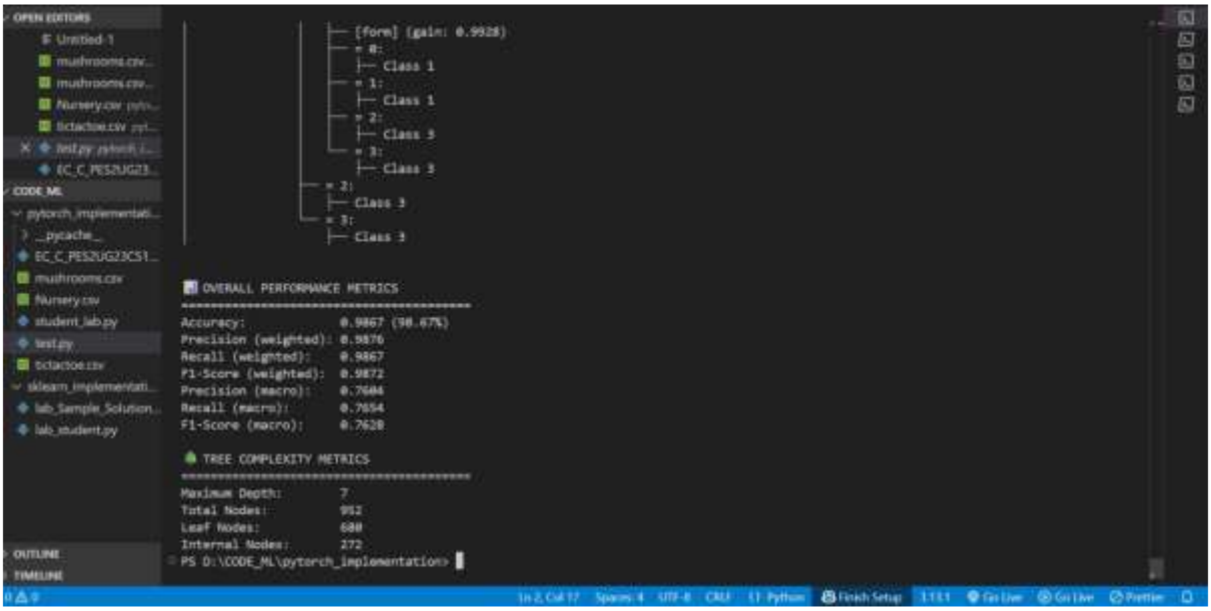
```
│   │       │   │   ├── = 1:
│   │       │   │   │   ├── [housing] (gain: 0.7219)
│   │       │   │   │   ├── = 0:
│   │       │   │   │   │   ├── Class 4
│   │       │   │   │   ├── = 1:
│   │       │   │   │   │   ├── Class 1
│   │       │   │   │   └── = 2:
│   │       │   │   │       ├── Class 4
│   │       │   │   ├── = 2:
│   │       │   │   │   ├── [housing] (gain: 0.3219)
│   │       │   │   │   ├── = 0:
│   │       │   │   │   │   ├── Class 1
│   │       │   │   │   ├── = 1:
│   │       │   │   │   │   ├── Class 1
│   │       │   │   │   └── = 2:
│   │       │   │   │       ├── Class 1
│   │       │   │   └── = 3:
│   │       │   │       ├── Class 4
│   │       │   ├── = 1:
│   │       │   │   ├── [form] (gain: 0.3601)
│   │       │   │   ├── = 0:
│   │       │   │   │   ├── [housing] (gain: 1.0000)
│   │       │   │   │   ├── = 1:
│   │       │   │   │   │   ├── Class 1
│   │       │   │   │   └── = 2:
│   │       │   │   │       ├── Class 4
│   │       │   │   ├── = 1:
│   │       │   │   │   ├── [housing] (gain: 0.9183)
│   │       │   │   │   ├── = 0:
│   │       │   │   │   │   ├── Class 4
│   │       │   │   │   ├── = 1:
│   │       │   │   │   │   ├── Class 1
│   │       │   │   │   └── = 2:
```



```
│   │       │   │   │   ├── Class 1
│   │       │   │   └── = 3:
│   │       │   │       ├── Class 1
│   │       │   ├── = 2:
│   │       │   │   ├── [housing] (gain: 0.3215)
│   │       │   │   ├── = 0:
│   │       │   │   │   ├── [finance] (gain: 1.0000)
│   │       │   │   │   ├── = 0:
│   │       │   │   │   │   ├── Class 4
│   │       │   │   │   └── = 1:
│   │       │   │   │       ├── Class 1
│   │       │   │   ├── = 1:
│   │       │   │   │   ├── Class 1
│   │       │   │   └── = 2:
│   │       │   │       ├── Class 1
│   │       │   └── = 3:
│   │       │       ├── [housing] (gain: 0.2669)
│   │       │       ├── = 0:
│   │       │       │   ├── [finance] (gain: 0.9852)
│   │       │       │   ├── = 0:
│   │       │       │   │   ├── Class 4
│   │       │       │   └── = 1:
│   │       │       │       ├── Class 1
│   │       │       ├── = 1:
│   │       │       │   ├── Class 1
│   │       │       └── = 2:
│   │       │           ├── Class 1
│   │       ├── = 1:
│   │       │   ├── Class 1
│   │       └── = 2:
│   │           ├── [housing] (gain: 0.2669)
│   │           ├── = 0:
│   │           │   ├── [finance] (gain: 0.4233)
```



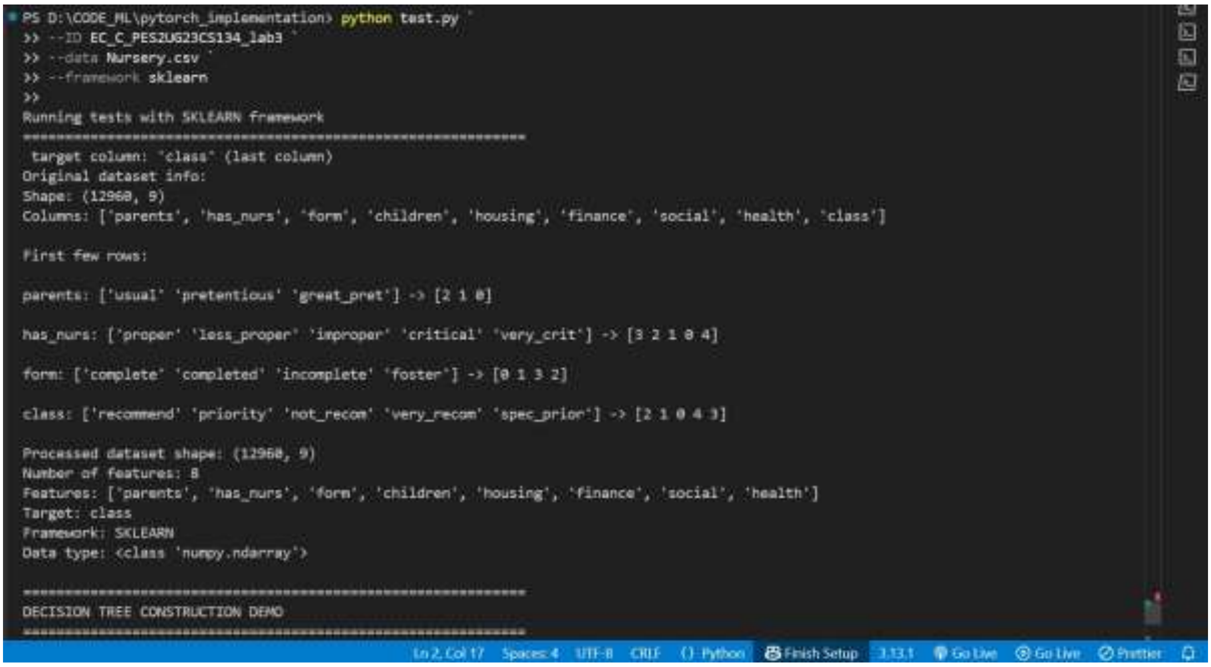
```
│   │           │   │   ├── [form] (gain: 0.3912)
│   │           │   │   ├── = 0:
│   │           │   │   │   ├── Class 2
│   │           │   │   ├── = 1:
│   │           │   │   │   ├── Class 4
│   │           │   │   ├── = 2:
│   │           │   │   │   ├── Class 4
│   │           │   │   └── = 3:
│   │           │   │       ├── Class 4
│   │           │   └── = 1:
│   │           │       ├── [form] (gain: 0.5710)
│   │           │       ├── = 0:
│   │           │       │   ├── Class 4
│   │           │       ├── = 1:
│   │           │       │   ├── Class 1
│   │           │       ├── = 2:
│   │           │       │   ├── Class 1
│   │           │       └── = 3:
│   │           │           ├── Class 1
│   │           ├── = 1:
│   │           │   ├── [children] (gain: 0.1769)
│   │           │   ├── = 0:
│   │           │   │   ├── [form] (gain: 0.9183)
│   │           │   │   ├── = 0:
│   │           │   │   │   ├── Class 4
│   │           │   │   ├── = 1:
│   │           │   │   │   ├── Class 1
│   │           │   │   ├── = 2:
│   │           │   │   │   ├── Class 1
│   │           │   │   └── = 3:
│   │           │   │       ├── Class 1
│   │           │   ├── = 1:
│   │           │   │   ├── Class 1
```



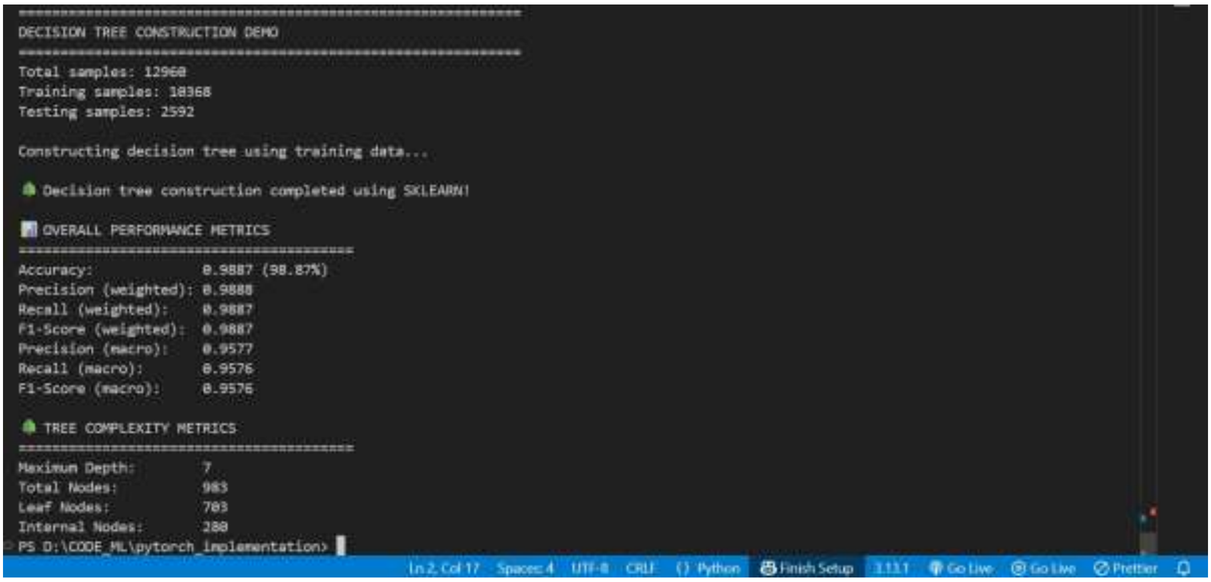
```
                    ├── [form] (gain: 0.9928)
                    ├── = 0:
                    │   ├── Class 1
                    ├── = 1:
                    │   ├── Class 1
                    ├── = 2:
                    │   ├── Class 3
                    └── = 3:
                        ├── Class 3
               ├── = 2:
               │   ├── Class 3
               └── = 3:
                   ├── Class 3

📊 OVERALL PERFORMANCE METRICS
========================================
Accuracy:             0.9867 (98.67%)
Precision (weighted): 0.9876
Recall (weighted):    0.9867
F1-Score (weighted):  0.9872
Precision (macro):    0.7604
Recall (macro):       0.7654
F1-Score (macro):     0.7628

🌳 TREE COMPLEXITY METRICS
========================================
Maximum Depth:        7
Total Nodes:          952
Leaf Nodes:           680
Internal Nodes:       272
PS D:\CODE_ML\pytorch_implementation> 
```

- **Class Imbalance:** Multi-class imbalance affected macro scores despite high weighted scores.
- **Feature Types:** Multi-valued features like form and housing provided better performance.
- **Interpretability:** Lower due to deep tree and many classes, but manageable with visualization.

## 3.SKLEARN IMPLEMENTATIONS



```
PS D:\CODE_ML\pytorch_implementation> python test.py `
>> --ID EC_C_PES2UG23CS134_lab3 `
>> --data Nursery.csv `
>> --framework sklearn
>>
Running tests with SKLEARN framework
============================================================
 target column: 'class' (last column)
Original dataset info:
Shape: (12960, 9)
Columns: ['parents', 'has_nurs', 'form', 'children', 'housing', 'finance', 'social', 'health', 'class']

First few rows:

parents: ['usual' 'pretentious' 'great_pret'] -> [2 1 0]

has_nurs: ['proper' 'less_proper' 'improper' 'critical' 'very_crit'] -> [3 2 1 0 4]

form: ['complete' 'completed' 'incomplete' 'foster'] -> [0 1 3 2]

class: ['recommend' 'priority' 'not_recom' 'very_recom' 'spec_prior'] -> [2 1 0 4 3]

Processed dataset shape: (12960, 9)
Number of features: 8
Features: ['parents', 'has_nurs', 'form', 'children', 'housing', 'finance', 'social', 'health']
Target: class
Framework: SKLEARN
Data type: <class 'numpy.ndarray'>

============================================================
DECISION TREE CONSTRUCTION DEMO
============================================================
```



```
============================================================
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
PS D:\CODE_ML\pytorch_implementation> 
```

- **Real-World Applications:** Education systems, admission automation, and social services.
- **Performance Improvements:** Class balancing and feature selection could reduce tree depth and improve macro scores.

ANALYSIS OF NURSERY DATASET:

**Accuracy**

- Achieved **98.67% accuracy** with PyTorch.
- Multi-class classification with deeper tree (**Depth: 7**, **Nodes: 952**).

**Dataset Size & Features**

- **12,960 samples**, **8 multi-valued features**.
- Larger dataset helped generalization but increased tree complexity.

**Class Balance**

- Imbalanced classes (e.g., fewer "very_recom" cases).
- High weighted scores but lower macro scores due to imbalance.

**Feature Types**

- Rich multi-valued features (e.g., housing, finance, health) enabled detailed splits.

**Practical Applications**

- Applicable in **education systems**, **automated admissions**, and **social policy modeling**.

**Improvements**

- Use **class balancing** and **feature selection** to reduce depth and improve macro scores.

OVERALL ANALYSIS

**a) Algorithm Performance**

**• Which dataset achieved the highest accuracy and why?**

- **Mushroom Classification** achieved **100% accuracy**.
- Reason: It had **binary classification**, **balanced classes**, and highly **informative features** like odor, which provided strong information gain and low entropy, making it ideal for decision tree learning.

**• How does dataset size affect performance?**

- **Nursery Dataset** (12,960 samples): High accuracy but required **deeper trees** and more computation.
- **Tic-Tac-Toe Dataset** (958 samples): Lower accuracy and higher tree complexity due to limited data.
- Larger datasets tend to **generalize better**, but may also introduce **noise** and require **more complex trees**.

• **What role does the number of features play?**

- **Mushroom**: 22 features → High accuracy with shallow tree.
- **Tic-Tac-Toe**: 9 features → Lower accuracy despite deeper tree.
- **Nursery**: 8 features → High accuracy with deeper tree.
- Conclusion: **Feature quality** matters more than quantity. Informative features lead to better splits and performance.

**b) Data Characteristics Impact**

• **How does class imbalance affect tree construction?**

- **Mushroom and Tic-Tac-Toe**: Balanced classes → Stable tree construction.
- **Nursery**: Multi-class imbalance → Biased splits and lower macro scores.
- Class imbalance can lead to **overfitting** on dominant classes and **reduced generalization**.

• **Which types of features (binary vs multi-valued) work better?**

- **Multi-valued features** (e.g., odor, form, housing) provide **richer splits** and **higher information gain**.
- **Binary features** (e.g., Tic-Tac-Toe squares) are simpler but less expressive.
- Conclusion: **Multi-valued features** generally perform better in decision trees.

**c) Practical Applications**

• **For which real-world scenarios is each dataset type most relevant?**

- **Mushroom**: Food safety, toxicology, agriculture.
- **Tic-Tac-Toe**: Game AI, pattern recognition, strategy modeling.
- **Nursery**: Education systems, admission automation, social services.

• **What are the interpretability advantages for each domain?**

- **Mushroom**: Highly interpretable due to shallow tree and binary splits.
- **Tic-Tac-Toe**: Moderate interpretability; deeper tree increases complexity.
- **Nursery**: Lower interpretability due to multi-class and deep tree, but manageable with visualization.

• **How would you improve performance for each dataset?**

- **Mushroom**: Already optimal; apply **pruning** for efficiency.
- **Tic-Tac-Toe**: Use **feature engineering** or **ensemble methods** (e.g., Random Forest).
- **Nursery**: Apply **class balancing** and **feature selection** to reduce tree depth and improve macro scores.