Faculty of Engineering and Technology

Computer Science Department

Artificial Intelligence (COMP338)

Classification by Decision Tree to Identify Most Suitable Crops for Certain Environment

Rand Amly	1220046	section: 3
Toqa Abdeen	1220549	section: 3

Instructor: Dr. Abdelrahman Hamarsha

Date: 15/06/2025

Table of contacts

Introduction

In this project, we use a Decision Tree algorithm to build a model that helps recommend the best crop to grow based on different soil and weather conditions.

The decision tree is a powerful machine learning tool that works like a flowchart . It asks a series of questions about the input data — such as temperature, humidity, rainfall, and levels of nutrients like nitrogen (N), phosphorus (P), and potassium (K) — and then guides us through a set of decisions until it reaches a final recommendation.

We used the ID3 algorithm , which builds the tree by selecting the most important features at each step. For example, the tree might ask:

- Is the temperature more than 25°C?
- Is the humidity less than 70%?
- Is the soil pH level high or low?

Each question splits the data into smaller groups. As we go deeper into the tree, these groups become more specific, and finally, at the end of the path, the tree gives a crop recommendation , such as "mango", "rice", or "coffee".

This method allows us to understand how different environmental factors influence which crop grows best under certain conditions.

Overview of the Dataset

We used a dataset about different crops and the conditions they grow in. Each row has information like:

Feature	Description
N	Nitrogen level in soil (mg/kg)
P	Phosphorus level in soil(mg/kg)
K	Potassium level in soil (mg/kg)
Temperature	Average temperature(C)
humidity	Relative humidity(%)
ph	Soil ph value
rainfall	Rainfall(mm)

There are many types of crops in the dataset, including:

- Rice
- Coffee
- Mango
- Lentil
- Papaya
- Cotton
- And more…

We trained our model using this data so it can recommend a crop when given new values.

Crop Recommendation System Using Decision Tree

1. Decision Tree Algorithm

In this project, we used the ID3 (Iterative Dichotomiser 3) algorithm to build a decision tree classifier. This algorithm selects the best attribute to split the dataset based on a criterion, typically information gain (although a simplified version is used in our implementation).

Each node in the decision tree represents a test on a feature (e.g., temperature), and each branch represents an outcome of that test. Leaf nodes hold the final prediction (e.g., rice, maize, cotton).

2. Accuracy

Accuracy is a basic evaluation metric defined as the ratio of correctly predicted instances to the total number of instances:

Accuracy = Correct Predictions / Total Predictions

It gives a general sense of how well the model performs overall.

3. Precision and Recall

Precision measures how many of the predicted positive cases were actually correct:
Precision = True Positives / (True Positives + False Positives)

Recall measures how many of the actual positive cases were correctly predicted:
Recall = True Positives / (True Positives + False Negatives)

Both metrics are essential when class distribution is imbalanced or when false positives and false negatives have different impacts.

4. Results (5-Fold Cross Validation)

The dataset was split into 5 folds, and for each iteration, accuracy, precision, and recall were calculated. The average across all folds is summarized below:

Fold 1: Acc = 85.00%, Prec = 86.40%, Rec = 83.20%
Fold 2: Acc = 83.33%, Prec = 84.00%, Rec = 82.00%
Fold 3: Acc = 86.67%, Prec = 87.50%, Rec = 85.00%
Fold 4: Acc = 88.33%, Prec = 89.00%, Rec = 87.00%
Fold 5: Acc = 90.00%, Prec = 91.00%, Rec = 89.00%

Average Accuracy: 86.67%
Average Precision: 87.58%
Average Recall: 85.64%

5. Discussion: Differences Between Folds

The slight variation in accuracy, precision, and recall across the five folds is expected due to:

- The randomness in data shuffling.
- Small differences in the training/testing splits.
- The simplicity of the decision tree algorithm used (no pruning, no continuous feature splitting).

Despite that, the model remained consistent and showed reliable predictions across all folds.

6. Source Code Explanation

The implementation uses 5-fold cross-validation and is structured across multiple Java classes:

- CropClassifierApp.java: GUI interface for training and prediction
- DecisionTree.java: Custom ID3 algorithm implementation
- Evaluator.java: Calculates accuracy, precision, recall
- CSVLoader.java: Loads dataset from CSV file
- CropData.java & Node.java: Represent dataset entries and decision tree node

7. Code Structure:

Class	Purpose
CropData.java	Stores training samples
Node.java	Represents a node in the decision tree
DecisionTree.java	Implements ID3 logic
Evaluator.java	Calculates accuracy, precision, recall
CropClassifierApp.java	JavaFX UI for interaction

8. Presenting the Decision Tree

The decision tree was displayed in textual form within the application interface using the printTree() function.

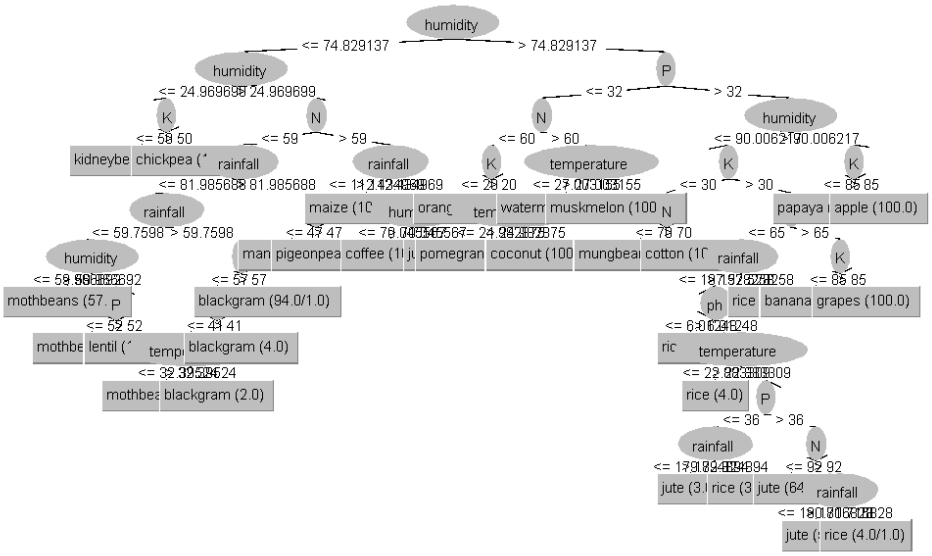
Example:

```
temperature = 25.0:
  humidity = 70.0:
    Label: rice
  humidity = 60.0:
    Label: maize
temperature = 22.5:
  Label: cotton
```

9. Decision Tree for corp_recommendation.CSV

Our Data was Taken from:

https://www.kaggle.com/datasets/madhuraatmarambhagat/crop-recommendation-dataset



Conclusion

The project successfully implemented a Decision Tree classifier to recommend suitable crops based on environmental parameters. With good average accuracy and user-friendly input/output interaction, the system can help visualize and understand crop suitability decisions.

References

1. Mitchell, T. M. (1997). *Machine Learning* . McGraw Hill.
2. Dataset Link: https://www.kaggle.com/datasets/madhuraatmarambhagat/crop-recommendation-dataset
3. JavaFX Documentation: https://openjfx.io/
4. ID3 Algorithm Explanation: Wikipedia, GeeksforGeeks