Credit Approval Decision Tree Project Report

1. Introduction

This report describes the implementation and results of the decision tree algorithms **C4.5** and **CART** applied to the **Credit Approval** dataset. The goal is to predict whether a credit application is approved (+) or denied (-) based on 15 features provided in the dataset.

Objectives:

- Implement two decision tree algorithms: **C4.5** and **CART**.
- Perform **10-fold cross-validation** on the training dataset to evaluate model performance.
- Evaluate the best model using a separate **test dataset**.
- Compare the performance of both models based on **accuracy** and **F1 score**.

---

2. Dataset Overview

The **Credit Approval** dataset consists of records detailing personal information and credit history, with 15 features and a target label indicating whether the credit was approved or denied. The dataset is divided into two parts:

- **Training Data**: Contains 553 records.
- **Test Data**: Contains 137 records.

Features:

- The dataset contains both categorical and continuous features, such as A1 to A15, representing attributes like age, credit history, income, etc.
- The **target variable** (A16) contains the decision, where + represents an approved credit application, and - represents a denied application.
- Missing values (?) are handled via median/middle-value imputation.

---

3. Methodology

The following two decision tree algorithms were implemented and evaluated on the dataset:

1. **C4.5 Algorithm**: A decision tree algorithm that uses **Gain Ratio** to select attributes for splitting.
2. **CART Algorithm**: A decision tree algorithm that uses the **Gini Index** to split attributes.

Cross-Validation Approach:

- **10-fold cross-validation** was used to train and evaluate the models. This method involves splitting the training dataset into 10 subsets, training the model on 9 subsets, and testing it on the remaining 1 subset. This process is repeated 10 times, and the average performance is recorded.

Evaluation Metrics:

- **Accuracy**: The percentage of correct predictions made by the model.
- **F1 Score**: A balanced measure of precision and recall, useful in evaluating performance when the dataset has class imbalance.

---

4. Implementation Details

The program was implemented in Python and used the following libraries:

- **numpy**: For numerical computations.
- **pandas**: For data manipulation and handling missing values.
- **math**: For logarithmic calculations required in the C4.5 algorithm.

Steps:

1. **Data Preprocessing**: The dataset was loaded, and missing values were imputed using the median for continuous features and the middle value for categorical features.
2. **Model Training**: Both the C4.5 and CART algorithms were implemented and trained on the training data using 10-fold cross-validation.
3. **Model Evaluation**: The performance of the models was evaluated using accuracy and F1 score on both the training and test datasets.

---

5. Results and Discussion

C4.5 Algorithm Results:

- **Cross-Validation Accuracy**: 81.45%
- **Cross-Validation F1 Score**: 77.91%
- **Test Set Accuracy**: 81.43%
- **Test Set F1 Score**: 79.69%

The **C4.5 algorithm** performed well on the training dataset, achieving an accuracy of approximately 81.45% during 10-fold cross-validation. It also performed similarly on the test dataset, with an accuracy of 81.43% and an F1 score of 79.69%.

CART Algorithm Results:

- **Cross-Validation Accuracy**: 76.36%
- **Cross-Validation F1 Score**: 72.98%

- **Test Set Accuracy**: 76.42%
- **Test Set F1 Score**: 74.13%

The **CART algorithm** showed lower performance compared to C4.5, with a cross-validation accuracy of 76.36% and an F1 score of 72.98%. The test set results were similar, with an accuracy of 76.42% and an F1 score of 74.13%.

Comparison of C4.5 and CART:

Model	Cross-Validation Accuracy	Cross-Validation F1 Score	Test Set Accuracy	Test Set F1 Score
C4.5	81.45%	77.91%	81.43%	79.69%
CART	76.36%	72.98%	76.42%	74.13%

From the comparison, **C4.5** outperformed **CART** in both accuracy and F1 score on both the training and test datasets.

---

6. Conclusion

In this project, both **C4.5** and **CART** decision tree algorithms were implemented and evaluated on the **Credit Approval** dataset. The **C4.5 algorithm** consistently outperformed **CART** in terms of accuracy and F1 score, both during 10-fold cross-validation and when tested on unseen data.

The results suggest that **C4.5** is the better model for credit approval prediction, as it achieves higher accuracy and F1 scores. However, further analysis could be conducted by tuning hyperparameters or applying other classification techniques to improve model performance.

---

7. Future Work

Future improvements could include:

- Hyperparameter tuning (e.g., adjusting tree depth or minimum gain threshold).
- Exploring alternative machine learning models such as **Random Forest** or **SVM** for comparison.
- Using **feature engineering** to improve the predictive power of the models.

---

8. References

- C4.5 Algorithm: Information on the C4.5 decision tree algorithm.

- CART Algorithm: Details of the CART decision tree algorithm.
- **Dataset Source**: UCI Machine Learning Repository or any relevant source for the Credit Approval dataset.