# Title: Fraud Detection Algorithm Using Decision Trees Technique

Introduction

Fraud detection is an essential aspect of maintaining a secure financial system. With the rise in digital transactions, fraud cases are increasingly becoming more sophisticated, making it crucial for financial institutions to develop advanced fraud detection algorithms. In this report, we discuss the implementation of a fraud detection algorithm using the decision tree technique. The dataset used for this project was obtained from Kaggle.com, and the algorithm was trained and tested using a partitioning system for the data.

Dataset and Data Partitioning

The dataset obtained from Kaggle.com comprises various features of financial transactions, including income and credit attributes. These features are used as the basis for training the decision tree model to detect fraud patterns. To prepare the dataset for training and testing, it was partitioned into two sets: 2/3 of the data was used for training, while the remaining 1/3 was used for testing.

To further enhance the model's performance, classes were created for certain intervals of income and credit attributes. This allows the model to recognize patterns within specific income and credit ranges, thus improving its ability to identify fraud cases.

Training the Decision Tree Model

The decision tree algorithm was chosen for its simplicity, interpretability, and effectiveness in handling both categorical and numerical data. It works by recursively splitting the data into subsets based on the feature that provides the maximum information gain, ultimately creating a tree-like structure. The leaf nodes of the tree represent the class labels, while the internal nodes represent the decision points.

The decision tree model was trained using the 2/3 training dataset partition. Once the model was trained, it was saved to a file for future use. This allows the model to be quickly loaded and applied to new data without the need to retrain it.

Testing and Evaluation

To evaluate the performance of the fraud detection algorithm, the decision tree model saved in the file was loaded and applied to the 1/3 testing dataset partition. The confusion matrix, a widely-used evaluation metric, was utilized to calculate the model's accuracy. The confusion matrix compares the predicted class labels with the actual class labels, providing a detailed breakdown of the model's performance in terms of true positive, true negative, false positive, and false negative rates.

Accuracy, which is the proportion of true positives and true negatives to the total number of instances, was chosen as the primary evaluation metric for this project. It provides an overall assessment of the model's performance in correctly identifying both fraudulent and non-fraudulent transactions.

Conclusion

In summary, a fraud detection algorithm using the decision tree technique was successfully implemented and evaluated. The model leverages a dataset obtained from Kaggle.com and employs a partitioning system to divide the data into training and testing sets. By creating classes for certain intervals of income and credit attributes, the model is better equipped to recognize patterns indicative of fraud.

The decision tree model was trained on the 2/3 training dataset partition and saved to a file for easy deployment. The model's performance was tested using the 1/3 testing dataset partition, and its accuracy was calculated using the confusion matrix which turned out to be 0.92. This project showcases the potential of decision trees as an effective tool for fraud detection, providing a strong foundation for future improvements and optimizations.

Team members:

Abdullah Maqsood-24448

Aqib Mahmood-25235