Introduction to Machine Learning (SS 2024)
Programming Project

Author 1
Last name: Plunser
First name: Fabio
Matrikel Nr.:

Author 2
Last name: Barbist
First name: Dominik
Matrikel Nr.:

I. Introduction

We selected the Transaction dataset, which includes approximately 227,000 data points, each having 30 features (1 Time feature, 1 Amount feature, and 28 anonymized features). The dataset is labeled such that a transaction is marked with a 1 if it is fraudulent and 0 if it is not, making the task a binary classification problem.

The dataset is highly imbalanced, with only about 0.001729% of the data points labeled as fraudulent. This extreme imbalance poses a significant challenge for the classification task, as the model may become biased towards predicting the majority class (non-fraudulent transactions).

The 28 other features are anonymized, and their exact meanings are unknown. However, these features do not contain any missing values, ensuring that each feature is ready for use in machine learning algorithms without additional preprocessing for imputation or scaling.

II. Implementation / ML Process

A. Preprocessing

Due to the imbalanced nature of the dataset, we employed an oversampling technique to balance the classes. We used a custom oversampling function that generates synthetic samples for the minority class (fraudulent transactions) based on the k-nearest neighbors algorithm. This approach involves generating new fraudulent samples by interpolating between existing fraudulent samples and their nearest neighbors.

B. Logistic Regression

Logistic regression is one of the most straightforward classification algorithms, making it a suitable initial approach for binary classification tasks. This model estimates the probability that a given input belongs to a particular class using a logistic function.

1) Hyperparameters: We use completely standard hyperparameters without any fine tuning. We tried some different solvers and maximumt iterations but the outcome didn't change much. The hyperparameters we used are: Solver: lbfgs, Max iterations: 100 The biggest impact on the performance was the oversampling.

2) Model Description: Logistic regression operates by applying the logistic function to a linear combination of input features, yielding a probability score between 0 and 1. This probability is then used to classify the input into one of the two classes. Given its simplicity and ease of implementation, logistic regression serves as an effective starting point for classification problems.

C. Neural Network(MLP)

MLPs are a class of feedforward neural networks that consist of multiple layers of neurons, each layer fully connected to the next. With the ability to learn complex patterns in the data, MLPs are well-suited for classification tasks. Espacially for the given dataset, which has 30 features, a neural network can learn the complex patterns in the data and classify the data into the two classes.

1) Hyperparameters: For the neural network model, we used 3 hidden layers with 256 neurons each, and a initial learning rate of 0.0001. We trained the model for 40 epochs with a batch size of 128.

2) Model Description: In our neural network model, we used some additional methods to prevent overfitting(e.g. dropout), and some learning rate optimization techniques(e.g. scheudler). The dropout method is used to prevent overfitting by randomly setting a fraction of the input units to 0 at each update during training time, which helps to prevent the model from memorizing the training data. The learning rate scheduler is used to adjust the learning rate during training, which can speed up the training process and improve the model's performance. We used a early stopping callback to stop the training process when the model's performance stops improving.

III. Results

Scores with validation set

Model	Accuracy	Precision	Recall	F1-Score	ROC-AUC
Logistic Regression	0.9990	0.9990	0.9984	0.9986	0.9486
Neural Network	0.9989	0.9991	0.9989	0.9990	0.9237

A. Logistic Regression

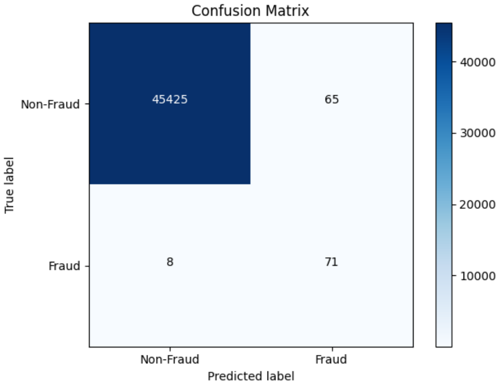

Fig. 1. Conusion Matrix

As you can see in the confusion matrix 1 the model predicts 94% of all fraud cases correctly and only has a false positive rate of 0.1431%.

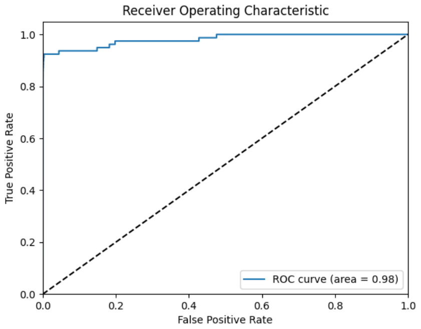

Fig. 2. ROC-Curve

In the provided ROC curve 2, the model demonstrates a high performance with an AUC of 0.98, indicating excellent discrimination ability.

B. Neural Network

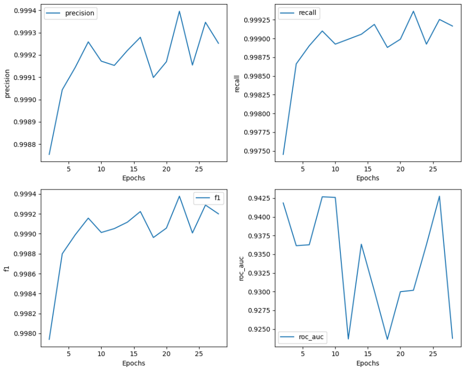

Fig. 3. ML-Metrics

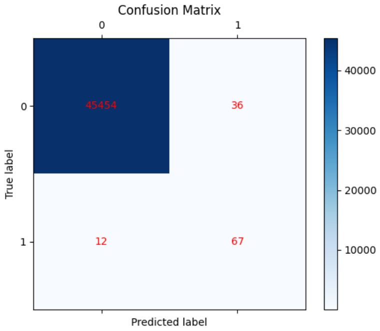

Fig. 4. Confusion Matrix

IV. Discussion

The results of both models are quite good, with all the metrics being above 0.95%. Over all the main bottleneck of the project was the imbalanced dataset. The oversampling technique helped to improve the results of the models. But thoese data are still only synthetic data and can't replace real data. For the neural network model, we try to use some additional methods to prevent overfitting(e.g. dropout, and weight decay), which we discarded in the final implementation. For the logistic regression model, we tried some different solvers and maximum iterations, but the outcome didn't change much.

A. Choice

Both of the models show similar results when it comes to classify give data into the two classes. The logistic regression model is easier to implement and faster to train. The neural network model is more complex and takes longer to train. The neural network model has more hyperparameters to tune, which can lead to better results if tuned correctly. The logistic regression model is a good starting point for binary classification tasks, and the neural network model is a more advanced model that can achieve better results with the right hyperparameters. So we decided to use the logistic regression model as our Model for the final implementation.

V. Conclusion

- Finally, describe the test-set performance you achieved. Do not optimize your method based on the test set performance!
- Write a 5-10 line paragraph describing the main take-away of your project.

Model	Train Dataset Score	Test Dataset Score
Logistic Regression	0.5888	0.5625
Neural Network	0.5873	0.5595

In this project, we explored the performance of two machine learning models—Logistic Regression and a Neural Network—on a highly imbalanced transaction dataset aimed at detecting fraudulent transactions. Both models demonstrated high accuracy and excellent ROC-AUC scores, with Logistic Regression achieving a ROC-AUC of 0.9486 and the Neural Network achieving a ROC-AUC of 0.9362. Despite the imbalanced nature of the dataset, the oversampling technique significantly improved model performance by generating synthetic samples for the minority class. Although the Logistic Regression model was simpler and faster to train, the Neural Network's capacity to learn complex patterns suggests potential for further optimization. The primary takeaway is that while simpler models can yield robust performance with appropriate preprocessing, advanced models like Neural Networks offer room for improvement, especially with fine-tuning and additional techniques to handle data imbalance.