IT1244 Project Report: Fraud Detection Dataset

Team 8
Cheong Zhi Kai Lionel
Koh Liang Han
Wayne Foo Fong Way
Ishan Lokhandwala

Introduction

Fraud detection is crucial for companies and organisations. Fraudulent activities, can lead to financial losses, decreased productivity and even erosion of trust from clients and partners (Pimentel, B., 2025). In this project, machine learning techniques are used to detect fraudulent activity in an Electricity and Gas Consumption dataset. The techniques employed are Long-Short Term Memory (LSTM), Extreme Gradient Boosting (XGB) and Logistic Regression models.

Machine Learning is already being utilised in fraud detection by organisations in the current day. Some examples include:

1. Using Neural Networks and Logistic Regression to detect energy meter anomalies and customer fraudulent behaviour such as meter tampering. (B. Coma-Puig, J. Carmona, R. Gavaldà, S. Alcoverro and V. Martin, 2016; Petrlik, I., Lezama, P., Rodriguez, C., Inquilla, R., Reyna-González, J. E., & Esparza, R., 2022)
2. Use of XGB models to classify fraudulent transactions in a well-known case study regarding a Tunisian energy company which suffered major financial losses caused by electricity fraud. (Oprea, S.-V., & Bâra, A, 2021)

Still, these models still possess limitations:

1. They occasionally report a high rate of false positives, that is non-fraudulent transactions are marked as frauds.
2. The XGB model may face issues when dealing with highly imbalanced datasets, such as large datasets where instances of fraud only account for less than 1% of cases. (Manchev, N, 2023)

Dataset

The dataset consists of two files, client.csv (six features) and invoice.csv (12 features). The client.csv contains features regarding client information, and invoice.csv contains features regarding invoice information. The common feature between the two datasets is ‘id’, which identifies which invoices in invoices.csv belong to which client in client.csv. The client.csv also contains the target feature, which classifies the client as fraudulent or non-fraudulent.

For the purposes of compatibility with multiple model types, the dataset was processed in two different ways. Firstly, dates were standardized to the date-time format, and the client and invoice features were aggregated into a single file, aggregated_datasetV2 using invoice information to compute the following features:

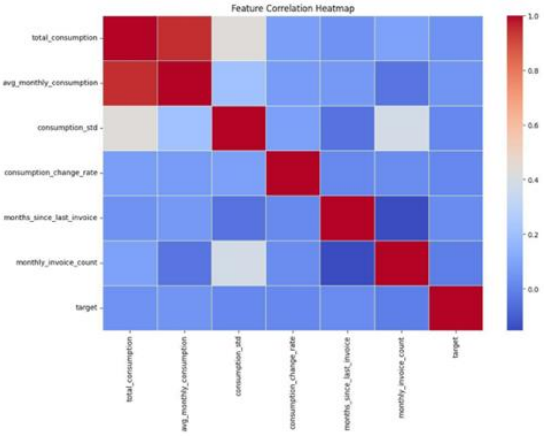
- total_consumption
- avg_monthly_consumption
- consumption_std
- consumption_change_rate
- months_since_last_invoice
- monthly_invoice_count

This was done to retain the sequential nature of the data, suitable for the LSTM. Using the ‘id’ feature, the dataset was also combined and processed to form aggregate_datasetV3.csv. Through feature

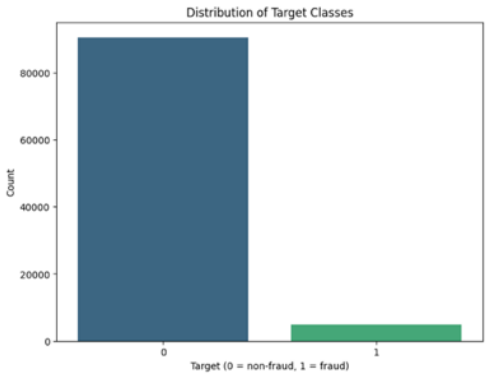
aggregation, the following variables are procured:

- Consommation (for levels 1 – 4: mean, max, std and sum)
- Consommation (for levels 1 – 4: sum by year)
- Months_number_mean
- Months_number_max
- Months_number_std

This was done because some features do not provide insightful information. As seen below on the correlation heatmap, the target class had very low to no correlation with almost every independent feature, even within the processed dataset aggregate_datasetV2.



However, the dataset has issues that need to be addressed. The data is significantly imbalanced as only 5% of the data are fraud cases, as seen in the figure below.



Adaptative Synthetic Sampling (ADASYN) was used to balance the dataset as it helps to improve the classification of imbalanced datasets generating synthetic samples of the minority class (fraud cases).

Methods

Three models were created to execute fraud detection on the dataset – LSTM, XGB and Logistic Regression. SKLEARN was used to split the data into an 80/20 split of training data and test data for all three models. Five-fold cross validation was also used for all models, and the best resulting model was used for evaluation on the test set.

LSTM

An LSTM is a special type of recurrent neural network (RNN), used for sequential data such as time-series data. This makes it appropriate for the dataset as the transaction data is sequential. The implemented model architecture consists of the following five layers:

1. A mask layer that filters out padded layers to not skew the dataset
2. An LSTM layer to process time-series data sequentially
3. A dense layer to enable the learning of non-linear combinations of features
4. A dropout layer to prevent overfitting
5. An output layer which squashes the output of the model between 0 and 1 (binary classification) using the Sigmoid function.

Binary cross-entropy was chosen as the loss function due to compatibility with sigmoid outputs compared to other loss functions. Paired with under and over sampling, it provides a solid loss criterion. Using cross-

fold validation, the model with the best results was chosen for evaluation.

XGB

Extreme Gradient Boosting, also known as XGBoost, is a scalable, distributed gradient boosted decision tree machine learning library for regression, classification and ranking problems (NVIDIA, 2025). XGB utilises decision trees to output predictions and is also able to perform feature selection by calculating and using feature importance scores during training. XGB is very effective in dealing with multivariate, imbalanced datasets, allowing it to be suitable for the current dataset. Hyperparameter tuning was also conducted to find the best parameters.

Logistic regression

Logistic Regression is a supervised machine learning algorithm, that classifies data points and uses a logistic function to return a probability value, which is used to produce a binary output. The logistic regression model is suitable as fraud classification is binary (i.e., fraud or not fraud). It may be a primitive method, but it has the benefit of being easy to implement.

Results & Discussion

For the evaluation of each model, a few key performance metrics were chosen. These were:

- Accuracy
- Precision
- Recall
- AUC Score
- Loss

After data processing we obtained the following results from our three models:

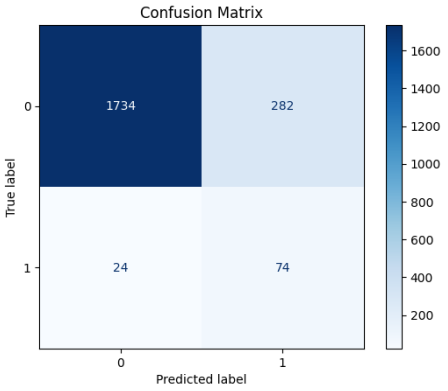
LSTM:

While architecture remained largely the same, multiple methods were used to train and tune the model. To combat the limitations faced in the studies cited above, namely the lack of fraudulent data, over sampling of the minority class and under sampling of the majority class was done. The LSTM was tested with and without these methods, and a combination of over and under sampling yielded best results.

This model showed the most promising results. It had the highest accuracy of the models, and most acceptable precision. However, due to class imbalance, a small volume of false positives skews the precision, as seen in the confusion matrix. Despite this, using sampling showed improvement over previous iterations and other models, which had a precision of <10%.

Metric	Value (%)
Test Loss	35
Test Accuracy	86
Test Precision	21
Test Recall	76
Test AUC	89

The confusion matrix reveals that the LSTM model was able to accurately predict majority of fraudulent and non-fraudulent cases, despite the imbalance in data skewing metrics like precision.



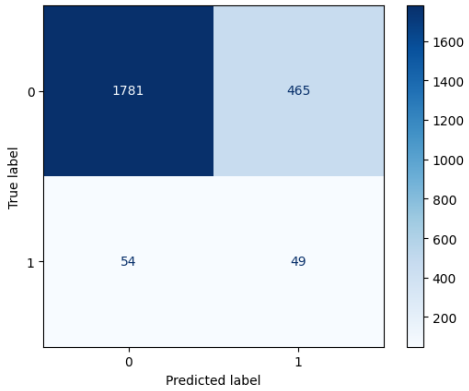
XGB:

The XGB model had the second-best results of the models trained, performing decently in all metrics. This was accomplished through the aggregated_datasetV3 as discussed prior. The result of this change is seen in improvements in all metrics.

Like LSTM, data was under-sampled and then balanced via ADASYN. However, the model performed worse compared to just using ADASYN for over sampling. Hyperparameter tuning was also conducted using RandomizedSearchCV, along with cross-validation. However, due to class imbalance, precision fell to 9.53%.

Metric	Value (%)
Test Loss	7.964
Test Accuracy	77.91
Test Precision	9.533
Test Recall	47.57
Test AUC	63.43

The confusion matrix reveals that while the XGB model predicted many true negatives, it struggled to classify true positives.

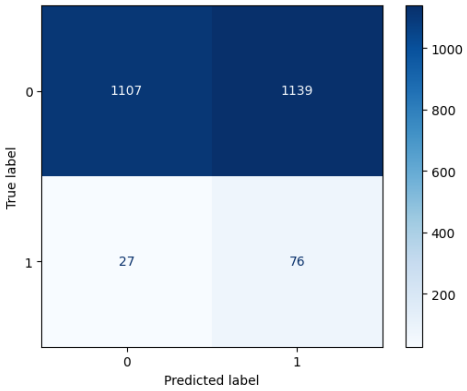


Logistic Regression:

The Logistic Regression model also uses the same dataset as XGB, showing improvements in all metrics. A combination of under-sampling and over sampling yielded worse performance compared to only over sampling. Hyperparameter tuning was also done using GridSearchCV. Despite adjustments, this had the worst performance of the models.

Metric	Value (%)
Test Loss	17.89
Test Accuracy	50.36
Test Precision	6.255
Test Recall	73.79
Test AUC	61.54

The confusion matrix reveals that the model struggled to even accurately classify non-fraudulent clients.



Overall, only over-sampling was beneficial for XGBoost and Logistic Regression. Since accurate classification is ultimately the goal, LSTM was the best, followed by XGB and Logistic Regression.

Appendix

References

B. Coma-Puig, J. Carmona, R. Gavaldà, S. Alcoverro and V. Martin, "Fraud Detection in Energy Consumption: A Supervised Approach," 2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA), Montreal, QC, Canada, 2016, pp. 120-129, doi: 10.1109/DSAA.2016.19.

Manchev, N. (2023, September 1). Credit card fraud detection using XGBoost, smote, and threshold moving. Domino Data Lab. https://domino.ai/blog/credit-card-fraud-detection-using-xgboost-smote-and-threshold-moving

Oprea, S.-V., & Bâra, A. (2021). Machine learning classification algorithms and anomaly detection in conventional meters and Tunisian electricity consumption large datasets. *Computers & Electrical Engineering*, *94*, 107329. https://doi.org/10.1016/j.compeleceng.2021.107329

Petrlik, I., Lezama, P., Rodriguez, C., Inquilla, R., Reyna-González, J. E., & Esparza, R. (2022). Electricity Theft Detection using Machine Learning. *International Journal of Advanced Computer Science and Applications*, *13*(12).

Pimentel, B. (2025, February 25). Fraud detection: An overview. Thomson Reuters Law Blog. https://legal.thomsonreuters.com/blog/what-is-fraud-detection

NVIDIA. (2019). What is XGBoost? NVIDIA Data Science Glossary. https://www.nvidia.com/en-sg/glossary/xgboost/