Credit Card Fraud Detection w/ Imbalance Dataset

Tony Ly

AutoEncoder: Anomaly Detection

Contents

1. Problem Statement/ Research Objective
2. Related Research
3. Understanding The Data
4. Methods
    a. Data Acquisition and Preprocessing
    b. Model Design
    c. Metrics
5. Results/Discussion
6. Conclusion/Next Steps

Problem Statement/Research Objective

Credit card transactions have been on the rise. Total global credit card transactions were an estimated **678 billion** in 2022 for an average of 1.86 billion per day, 77.4 million per hour, 1.29 million per minute, 21,510 per second. [1] With the rise of credit card usage, there is also a rise with credit card fraud. **$8.8 billion** was lost to fraud in 2022 and **441,822 cases** were reported to the Federal Trade Commission (FTC). [2] There is a need to develop a model that is able to detect fraud transactions.

- Develop an AutoEncoder type neural network architecture that can detect fraud transactions while effectively handle imbalanced datasets.
- Minimize false negatives and false positives
- Compare the performance of the proposed model to existing fraud detection models on the same dataset.

Related Research

Existing Fraud Detection Methods:

- Random Forest Algorithm [3]

  Proposes an ensemble learning algorithm for classification and performance is measured on confusion matrix, reported an accuracy of 90%

- Artificial Neural Network [4]

  Proposes an ANN model for fraud detection and performance is measured on Accuracy: 99.92%, Precision: 81.15%, and Recall: 79.19%

Related Research

- Distributed Deep Neural Network (DDNN) [5]

  Described that a DDNN model can avoid privacy leakage and data handling costs, accelerates convergence of the model, and detects fraud better than multiple types of centralized models

- Decision Tree Classification [6]

  Proposes a simple ML classification algorithm and performance is measured by Precision: 89%, Recall: 88%, F1-score: 89%

Method: Data Acquisition and Preprocessing

Data is taken from Kaggle; Credit Card Fraud Detection: Anonymized credit card transactions labeled as fraudulent or genuine [7]

Details:

- Predetermined classes: Normal (0) or Fraud (1)
- Total of 30 features
- 28 of the features are anonymized due to confidentiality issues
- Does not contain any missing or null values

Features:

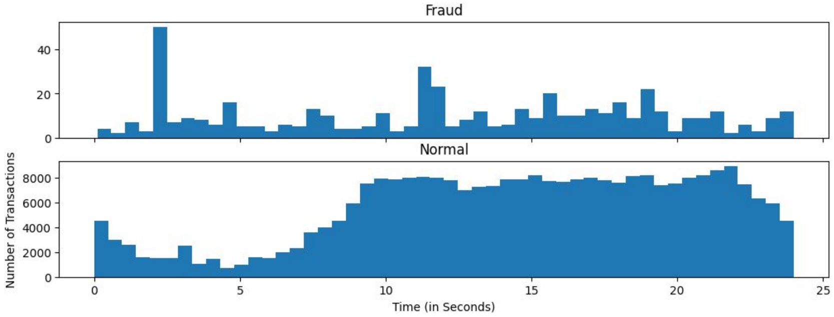
```
Index(['Time', 'V1', 'V2', 'V3', 'V4', 'V5',
'V6', 'V7', 'V8', 'V9', 'V10', 'V11', 'V12',
'V13', 'V14', 'V15', 'V16', 'V17', 'V18',
'V19', 'V20', 'V21', 'V22', 'V23', 'V24',
'V25', 'V26', 'V27', 'V28', 'Amount', 'Class'],
dtype='object')
```

	Class	**Count**	**Percent**
0	0	284315	99.83
1	1	492	0.17

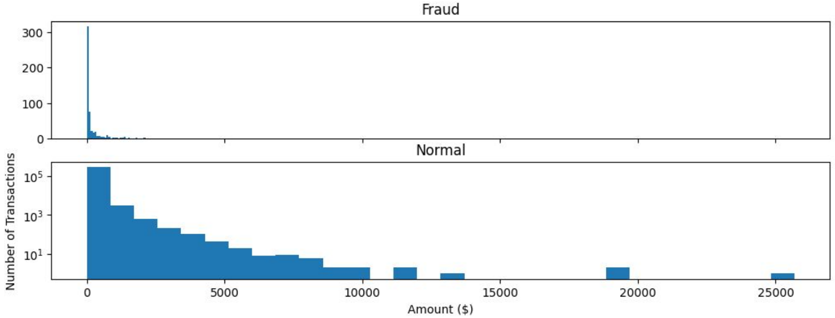
The dataset is heavily imbalanced.

Method: Data Acquisition and Preprocessing

- The 'Time' feature looks pretty similar across both types of transactions.
- Could argue that fraudulent transactions are more uniformly distributed, while normal transactions have a cyclical distribution.
- This could make it easier to detect a fraudulent transaction during at an 'off-peak' time.

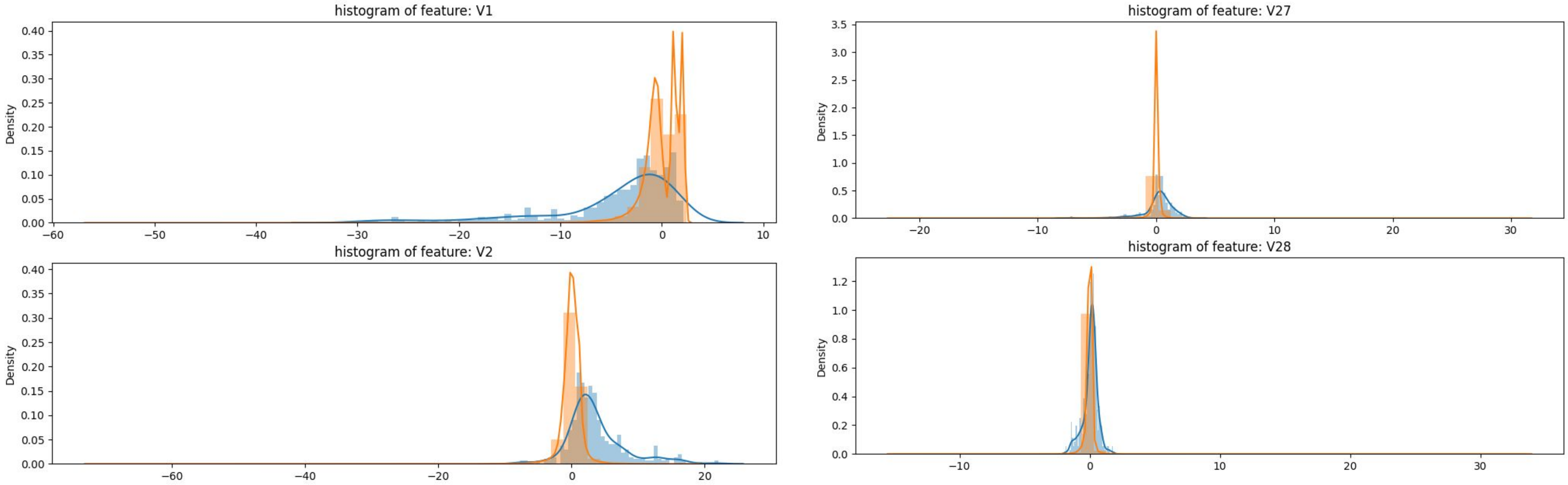


- Most transactions are small amounts, less than 100. Fraudulent transactions have a maximum value far less than normal transactions, $2,125.87 vs $25,691.16.



Method: Data Acquisition and Preprocessing

Normal vs Fraud on Anonymized Features



Orange - Normal cases (0) Blue - Fraud cases (1)

Method: Data Acquisition and Preprocessing

11 of the 28 anonymized features have similar distributions between the two types of transactions:

`'V28','V27','V26','V25','V24','V23','V22','V20','V15','V13','V8'`

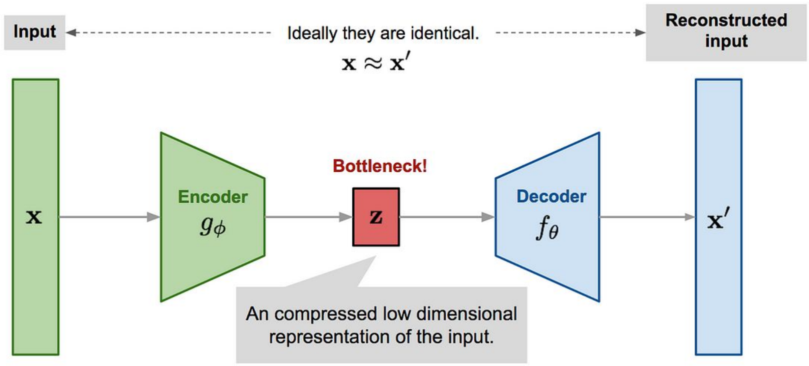
The PCA have already normalized the anonymized features V1 - 28, but not for features: Time and Amount.

Used Scikit's StandardScaler to normalize the features

Split data into train, validate and test sets

- Train - Test: 80/20
- Train - Val: 80/20
- Ensure train and validate sets only contain normal transactions (class = 0)
- Ensure test set contains fraud transactions (class = 1)

Method: Model Design



- Encoder compresses the normal transactions into a lower-dimensional space representation
- When presented with new transaction the decoder tries to reconstruct the transaction from the output of the encoder
- If transaction is normal, reconstruction error will be low, else if the transaction is fraudulent, the error will be high

Method: Model Design

Model: "sequential_3"

Layer (type)	Output Shape	Param #
dense_40 (Dense)	(None, 25)	775
dense_41 (Dense)	(None, 19)	494
dense_42 (Dense)	(None, 13)	260
dense_43 (Dense)	(None, 6)	84
dense_44 (Dense)	(None, 6)	42
dense_45 (Dense)	(None, 13)	91
dense_46 (Dense)	(None, 19)	266
dense_47 (Dense)	(None, 30)	600

Total params: 2612 (10.20 KB)
Trainable params: 2612 (10.20 KB)
Non-trainable params: 0 (0.00 Byte)

Based on the Normal vs Fraud Features Plots, we can use the information to determine the bottleneck of the model.

- Optimizer = Adam
- Metrics = Acc
- Loss = MSE
- Epoch = 50
- Batch size = 512
- Learning rate = 1e-7

Method: Metrics

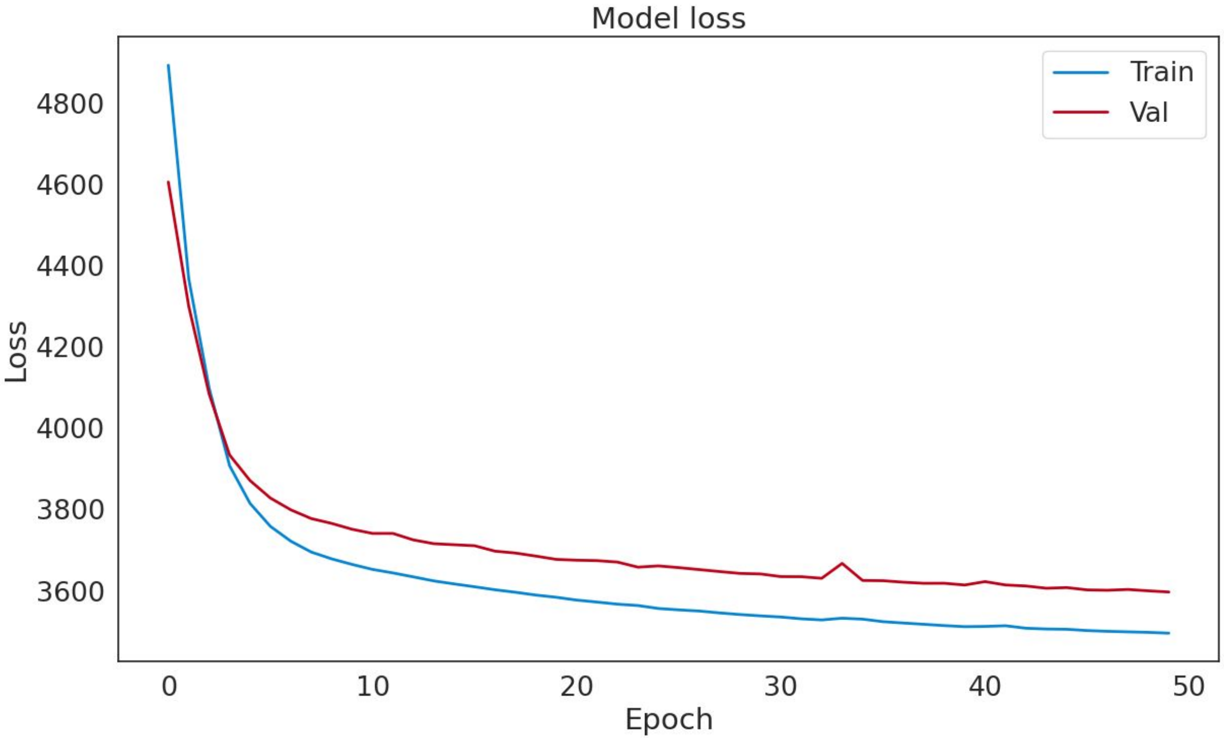
$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

	Predicted: 0	**Predicted: 1**
Actual: 0	True Negatives (**TN**)	False Positives (**FP**)
Actual: 1	False Negatives (**FN**)	True Positives (**TP**)

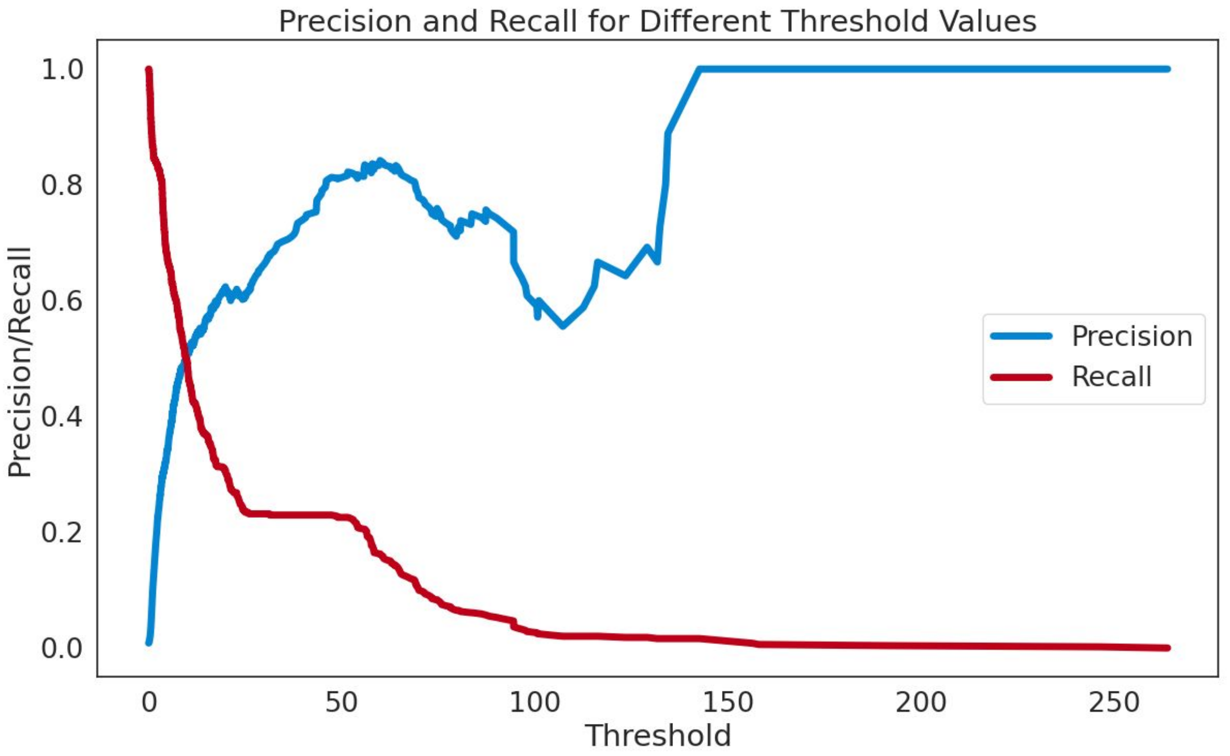
$$F_1 = \frac{2}{precision^{-1} + recall^{-1}}$$

Results/Discussion



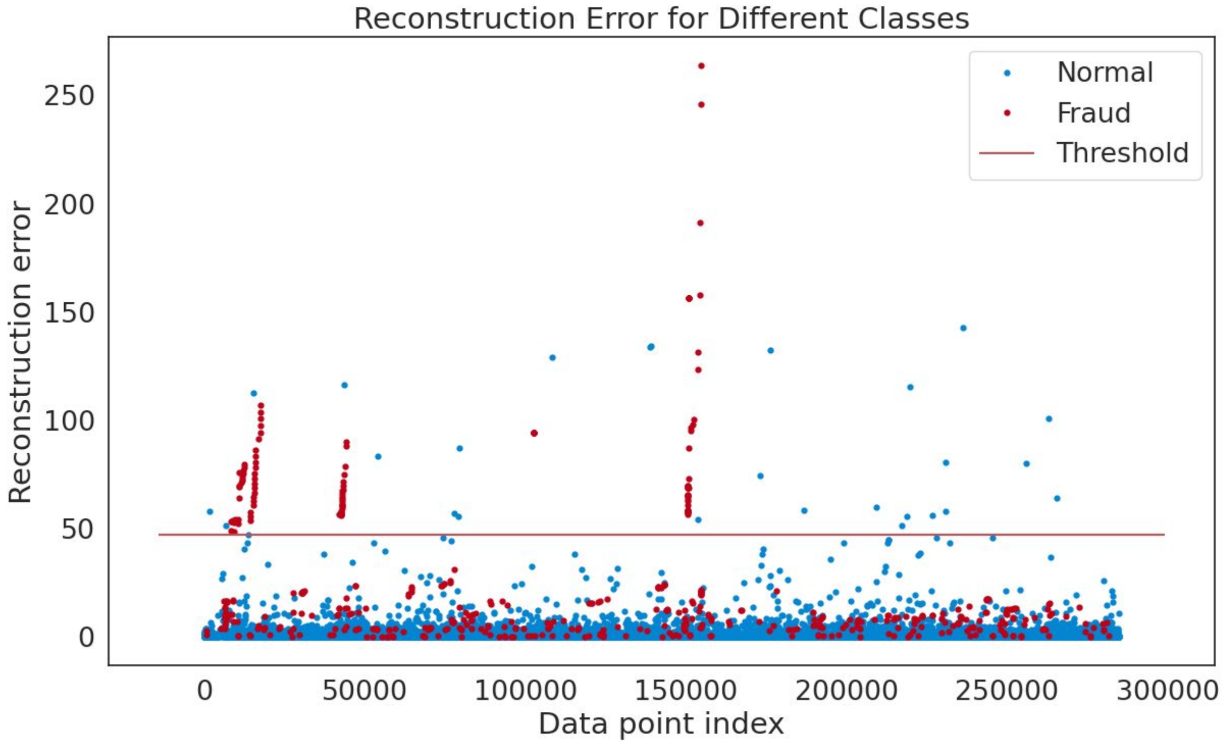
The model is **OVERFIT**

Results/Discussion



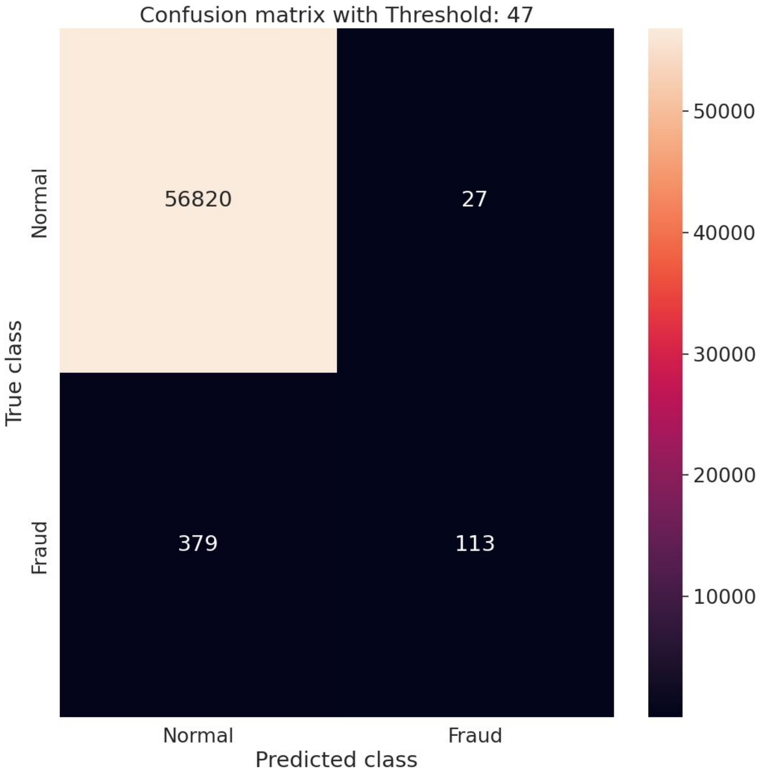
Threshold ≈ 50

Results/Discussion



Threshold = 47

Results/Discussion



Precision: 80.71%

Recall: 22.97%

F1-score: 35.76%

Conclusion/Next Steps

Proposed an AutoEncoder neural network to detect fraud in credit card transactions. Unfortunately, the performance was not what we expected/hoped for. The existing fraud detection models yield better results in all metric categories: precision, recall, and f1-score in comparison to our proposed model.

- Data Expansion
  - The dataset used to train the model is rather small (284,808 data entries); acquire additional data
- Model Improvement
  - Additional fine-tuning of AutoEncoder model
    - Hyperparameters
    - AutoEncoder architecture
  - Explore alternative deep learning architectures

References

[1] “Number of credit card transactions per second & Year: 2023 data,” Capital One Shopping, https://capitaloneshopping.com/research/number-of-credit-card-transactions/.

[2] A. Miller, “Credit Card Fraud & ID theft - facts & statistics [2023 data study],” UpgradedPoints.com, https://upgradedpoints.com/credit-cards/credit-card-fraud-and-id-theft-statistics/.

[3] M. S. Kumar, V. Soundarya, S. Kavitha, E. S. Keerthika and E. Aswini, "Credit Card Fraud Detection Using Random Forest Algorithm," 2019 3rd International Conference on Computing and Communications Technologies (ICCCT), Chennai, India, 2019, pp. 149-153, doi: 10.1109/ICCCT2.2019.8824930.

[4] Asha RB and Suresh Kumar KR, “Credit Card Fraud Detection Using Artificial Neural Network,” 2021 Global Transitions Proceedings , 2021, pp. 35-41, doi: 10.1016/j.gltp.2021.01.006.

References

[5] Yu-Tian Lei, Chao-Qun Ma, Yi-Shuai Ren, Xun-Qi Chen, Seema Narayan, Anh Ngoc Quang Huynh, "A Distributed Deep Neural Network Model for Credit Card Fraud Detection," 2023 Finance Research Letters, 2023, doi: 10.1016/j.frl.2023.104547.

[6] "Credit card fraud detection with classification algorithms in Python," Dataaspirant, https://dataaspirant.com/credit-card-fraud-detection-classification-algorithms-python/#t-1600793624243.

[7] M. L. G.- ULB, "Credit Card Fraud Detection," Kaggle, https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud/data.