AUTOENCODER FOR FRAUDULENT TRANSACTIONS DATA FEATURE ENGINEERING

Dalia Breskuvienė; Gintautas Dzemyda

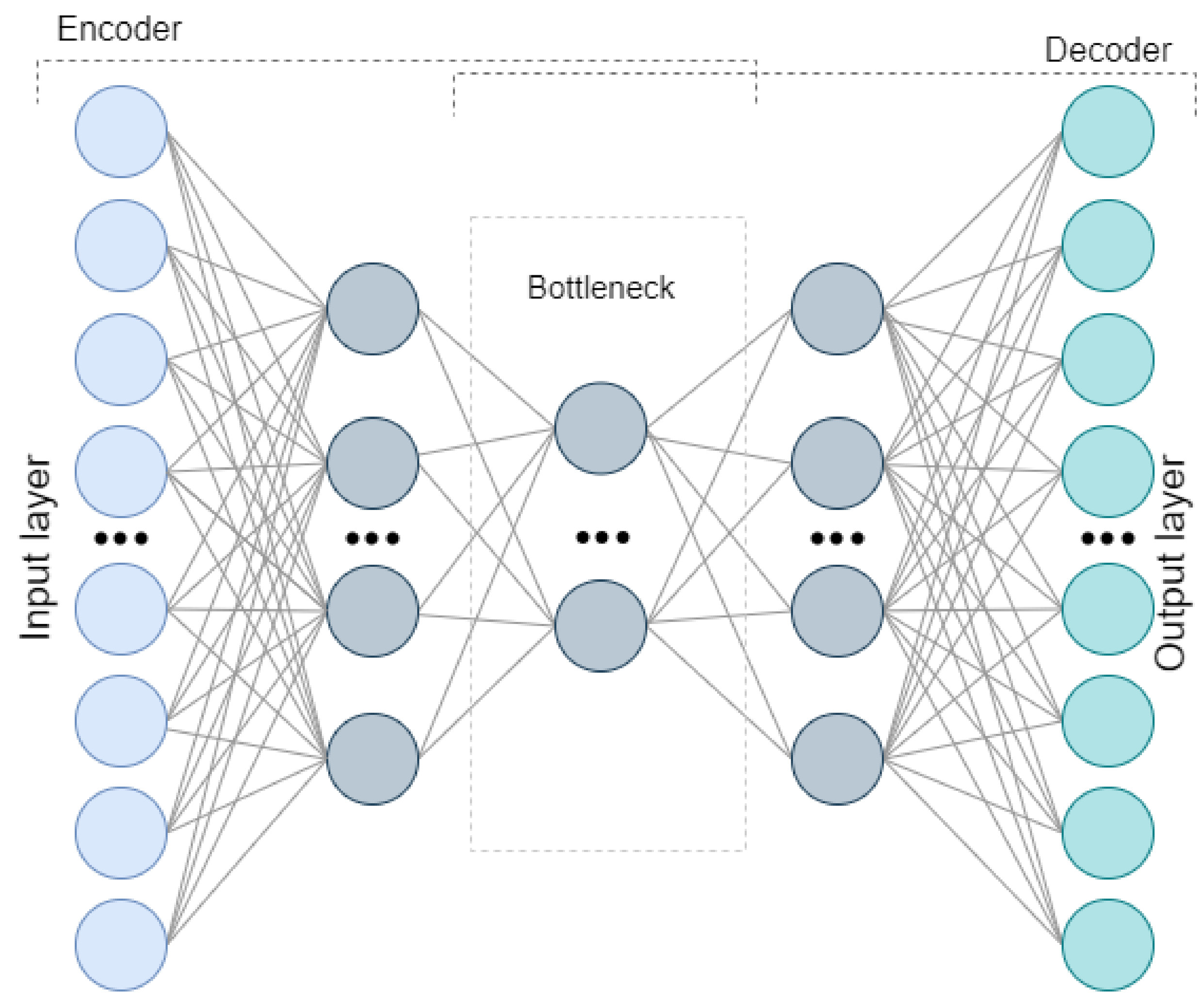
Data Science and Digital Technologies Institute, Vilnius University

Introduction

Fraud is an illegal action by someone who wants to gain financial benefit from another person or institution. It has evident economic consequences on private enterprises, public services, and individuals' financial situation. Usually, fraudulent activity is not a one-time act and is constantly evolving - it has no persistent patterns. Machine learning is one of the approaches to solve fraud detection problems. However, using machine learning algorithms for these issues requires careful data preparation, feature engineering, and feature encoding.

Methodology

AutoEncoder is an unsupervised Artificial Neural Network that is made of two parts. The first one is an encoder that aims to encode the data by compressing it into the lower dimensions. The second one is the decoder which reconstructs the original data input. We have to create a bottleneck layer that holds the compressed representation of the input data. The bottleneck consists of as many neurons as we would like to have reduced features.



Experiments

We use a Synthetic credit card transaction database with the simulated United States customers' transaction information for the experimental evaluation. Many features of this data set are categorical (nominal), with no intrinsic ordering to the categories. We are especially interested in coding features with many distinct categories such as "State" with 51 and "MCC" (Merchant Category code) with 109 unique values.

Building AutoEncoder:

Multilayer Perceptron (MLP) AutoEncoder was used for the experiments.

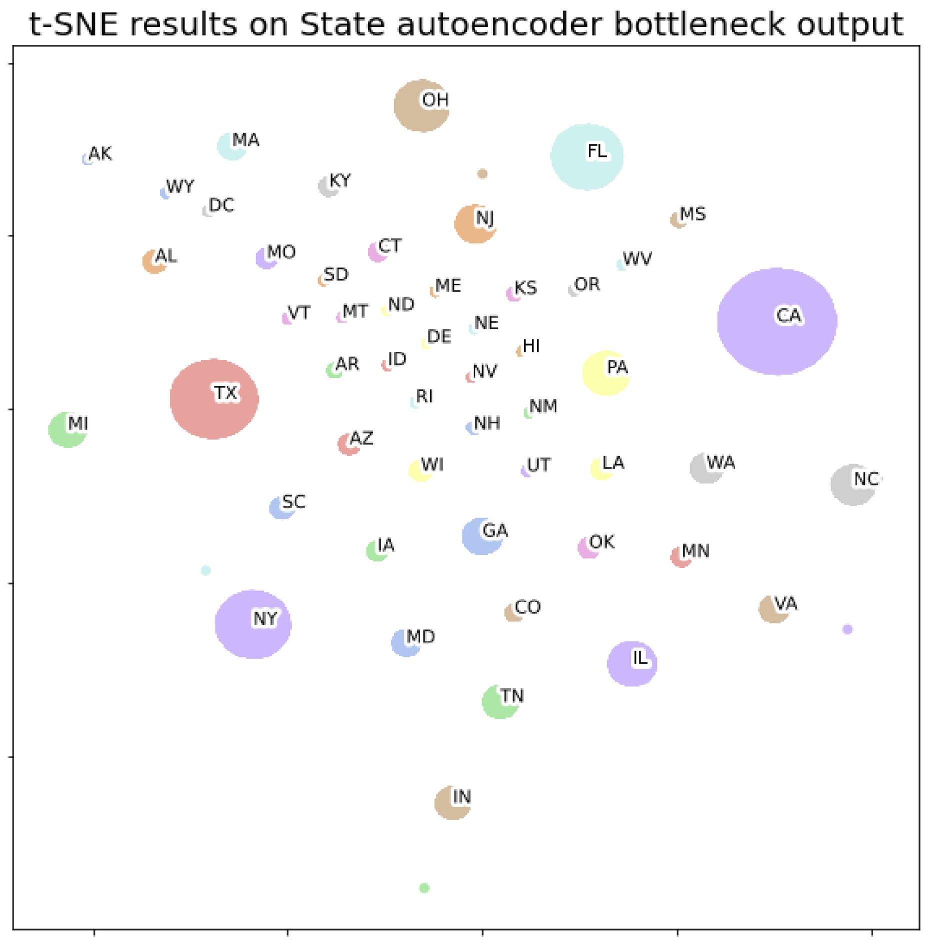
Activation functions used for the experiment:
- ReLu
- Tanh

Loss functions used for the experiment:
- Mean squared error (MSE)
- Categorical CrossEntropy

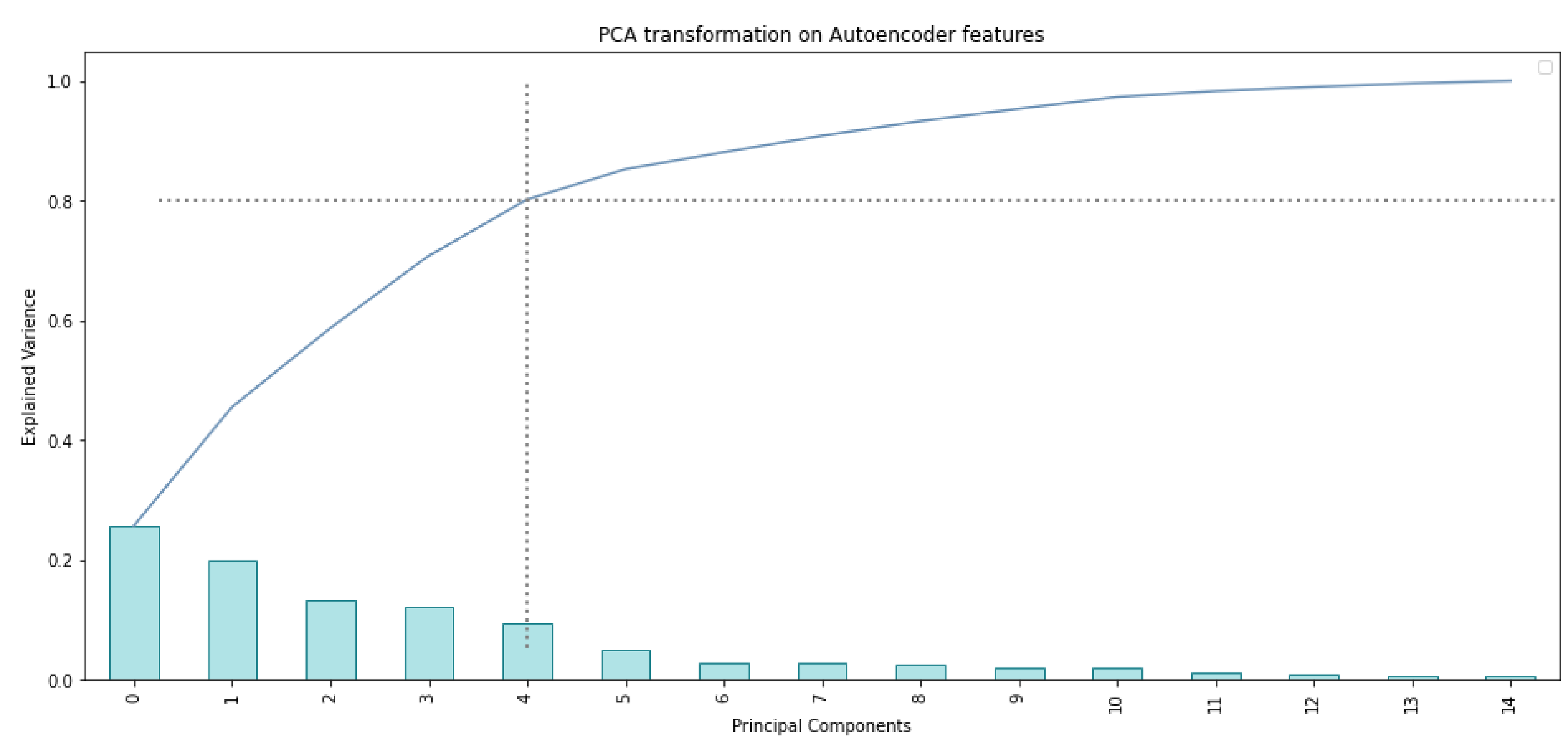
The best results were achieved with the CrossEntropy loss function and ReLu activation function. Feature "State" and "MCC" were transformed into 5 and 10 encoded features, respectively.

T-SNE visualization on the right proves that using only 5 AutoEncoder features we still hold relevant information of 51 category of feature "State".

t-SNE results on State autoencoder bottleneck output



Having 51 States and 109 Merchant Category Codes AutoEncoder might increase dimensionality to the level where machine learning algorithm performance starts to deteriorate. Principal component analysis (PCA) transformation solves this problem.



We faced dimensionality problem when adding both encoded features to the model. The performance deteriorated if compared to adding these features separately. Five principal components of PCA transformation were used for the classification model as they carry 80% of variability.

Results

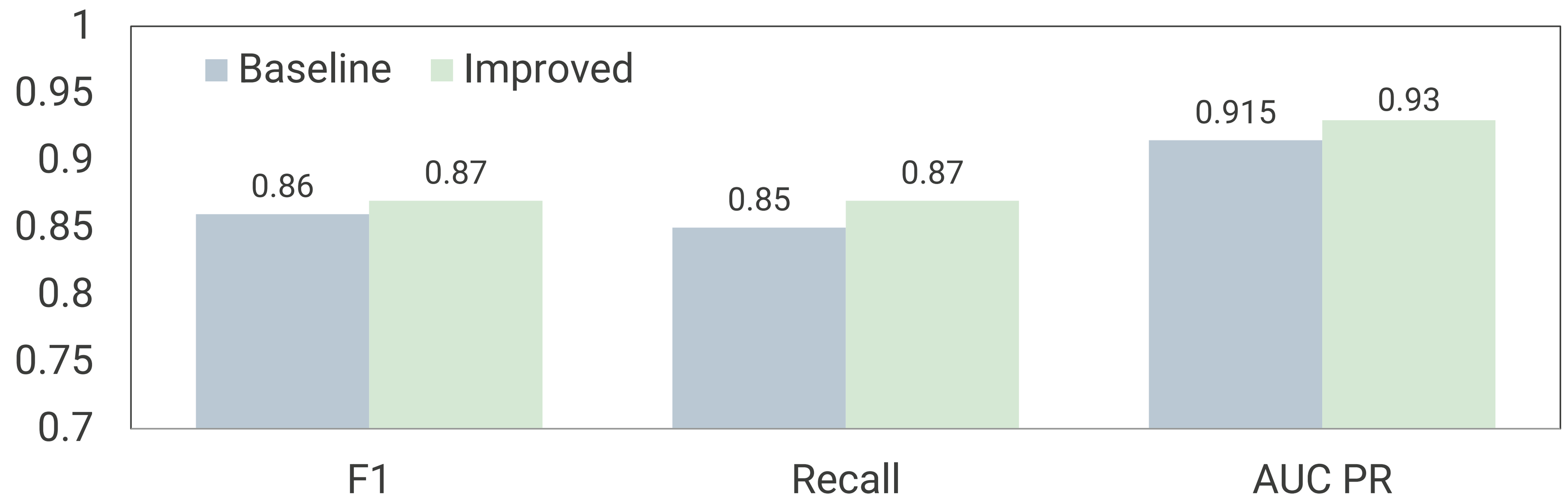
Using above mentioned features in the ML model improved the performance by identifying the real fraudulent cases better and at the same time reducing the number of faulty labeled fraud cases.

Improvement

Actual	Predicted: Not Fraud	Predicted: Fraud
Not Fraud	2 058 120	231
Fraud	323	1 764

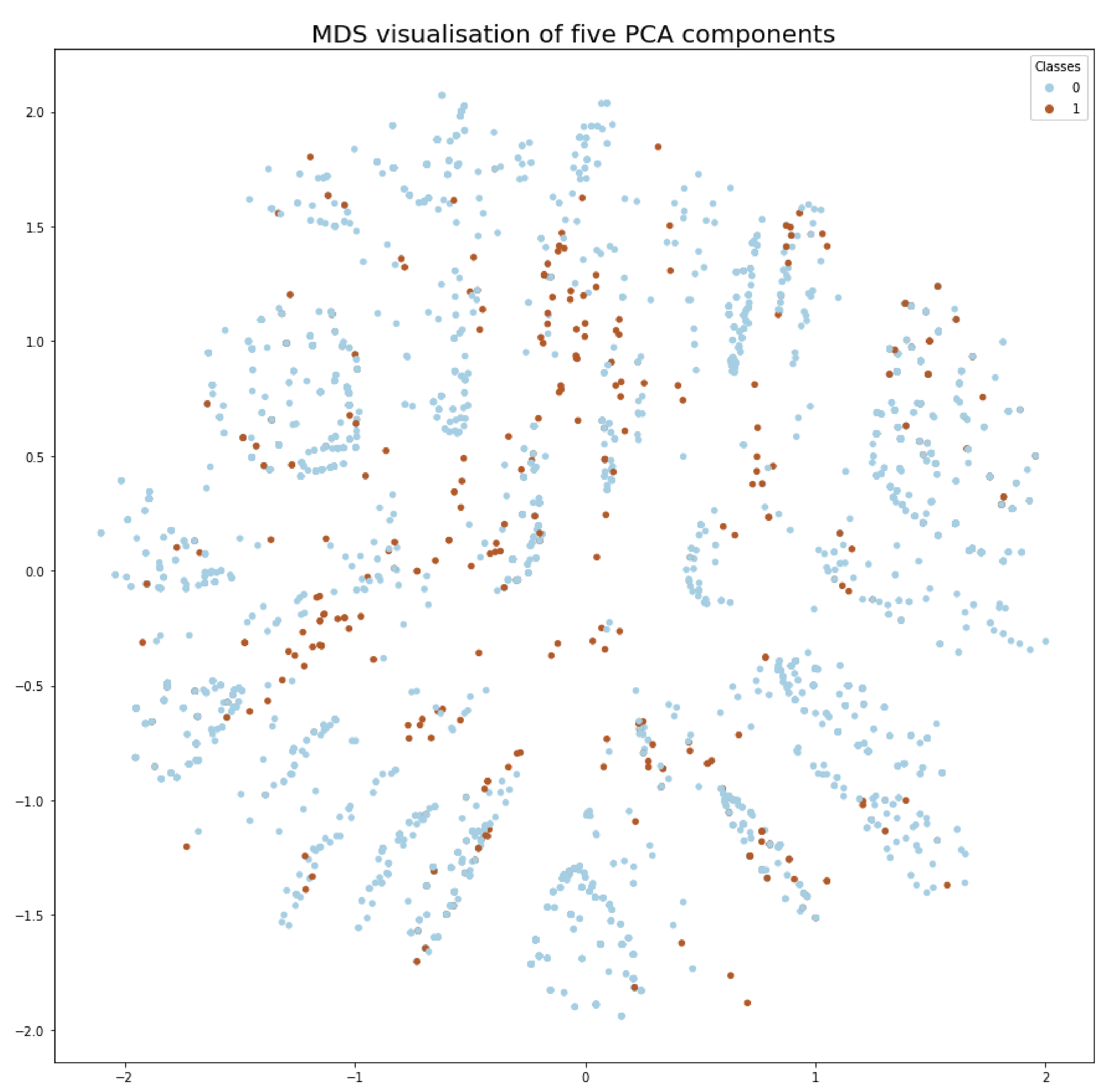
Actual	Predicted: Not Fraud	Predicted: Fraud
Not Fraud	2 058 094	257
Fraud	278	1 809

All three main metrics – F1, Recall and AUC Precision-Recall - have improved only by using AutoEncoder for feature engineering.



Conclusion

Multidimensional scaling gives an interesting pattern of fraudulent cases, which, after PCA transformation on AutoEncoder output, is grouped into separate clouds that are based on 'State' and 'MCC'. This outcome could be used for improving data clustering in the approach proposed in article [1]. At the same time, we can notice that some groups have very low probability of the event of fraud, which lets us put a different weights in the ML classification.



Contact

Dalia.Breskuviene@mif.stud.vu.lt

References

1. "D. Breskuvienė, G. Dzemyda (2023). Imbalanced data classification approach based on clustered training set. In: Dzemyda G., Bernatavičienė J., Kacprzyk J. (Eds.), Data Science in Applications. Studies in Computational Intelligence. Springer (accepted)"