FINAL PROJECT FOR DEEP LEARNING AND REINFORCEMENT LEARNING COURSE

CREDIT CARD FRAUD DETECTION

Bárbara Azevedo de Sá

SUMMARY

This report presents the analysis of a dataset for credit card fraud detection using Deep Learning techniques. The main goal is to define the best Deep Learning model that can predict if a credit card transaction is fraudulent or not, using some inputs such as location, amount and time of a specific transaction. This type of application of neural network is significant for several reasons, such as financial security, timely detection, reduced of false positive, adaptability, cost savings (early detection), customer trust, legal compliance and industry reputation.

DATA DESCRIPTION

The data set contains credit card transactions made by European cardholders in 2023, it was acquired in the website kaggle.com. All data points have been filtered to not disclosure any sensitive and personal information of any of the cardholders. The dataset has over 550 thousand records, in Table 1 there are a few samples of the dataset.

- **Id [int]**: Unique identifier for each transaction.
- **V1-V28 [float]**: Anonymized features representing various transaction attributes (for example, time, location and others).
- **Amount [float]**: The transaction amount.
- **Class [int]**: Binary label indicating whether the transaction is fraudulent (1) or not (0).

id	V1	V2	V3	...	V26	V27	V28	Amount	Class
107104	0,06029	-0,58180	1994880	...	-1692988	-0,40662	-0,31672	6532	0
442195	-1152400	0,55429	-1158501	...	1999446	1544257	-0,78903	23576	1
552215	-1040710	-0,88917	-0,71781	...	-0,57896	0,19587	-1711099	12679	1
521977	1388157	0,08528	-0,51706	...	-1501573	-0,29322	-0,07165	1247	1
192177	1746629	-0,44559	-0,39182	...	-1146310	-0,25778	-0,25930	5773	0
129580	-0,28387	-0,54658	1648059	...	0,11000	-0,25358	0,28966	23153	0

Table 1 – Sample from original data set

The distribution of the values in the data set is 50-50%, according to plot of Figure 1, each result (fraud/not fraud) has 284315 values. The data set distribution is important to define the accuracy target, as it is a 50-50% distribution, it means that if the model defines 1 for every single prediction, the model would already reach 50% accuracy. Then, that's why the accuracy target is at least higher than 50%.

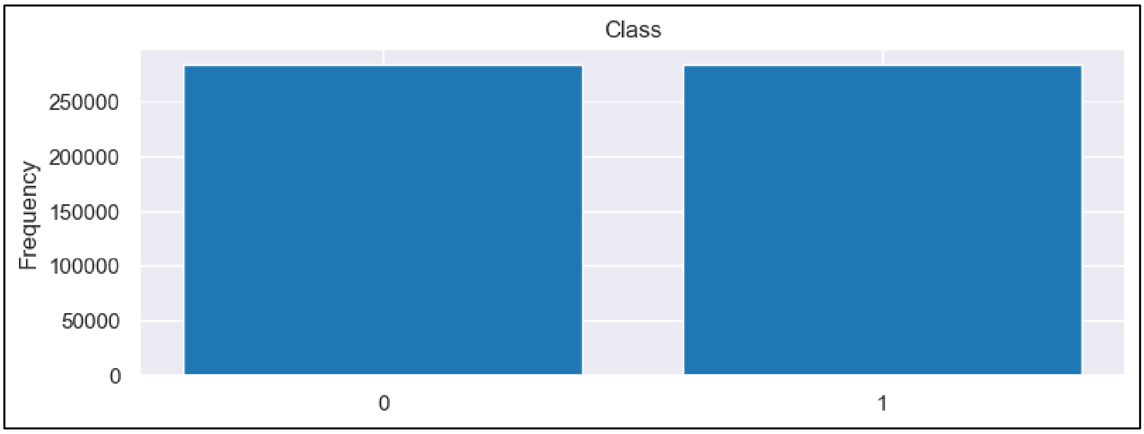

Figure 1 – Data set distribution

Another point to define the accuracy target is the correlation value between the output and all of the features. If the dataset has low correlation, it's more difficult to reach a good accuracy level, however, if the correlation is high, it's easier to reach higher accuracy with the model. In Figure 2 it's possible to check the correlation heatmap. As most of the features have a good correlation, it's important to define a neural network that used all of the features to calculate the predictions. The only feature that was removed from the dataset for lack of correlation was the "id" feature.

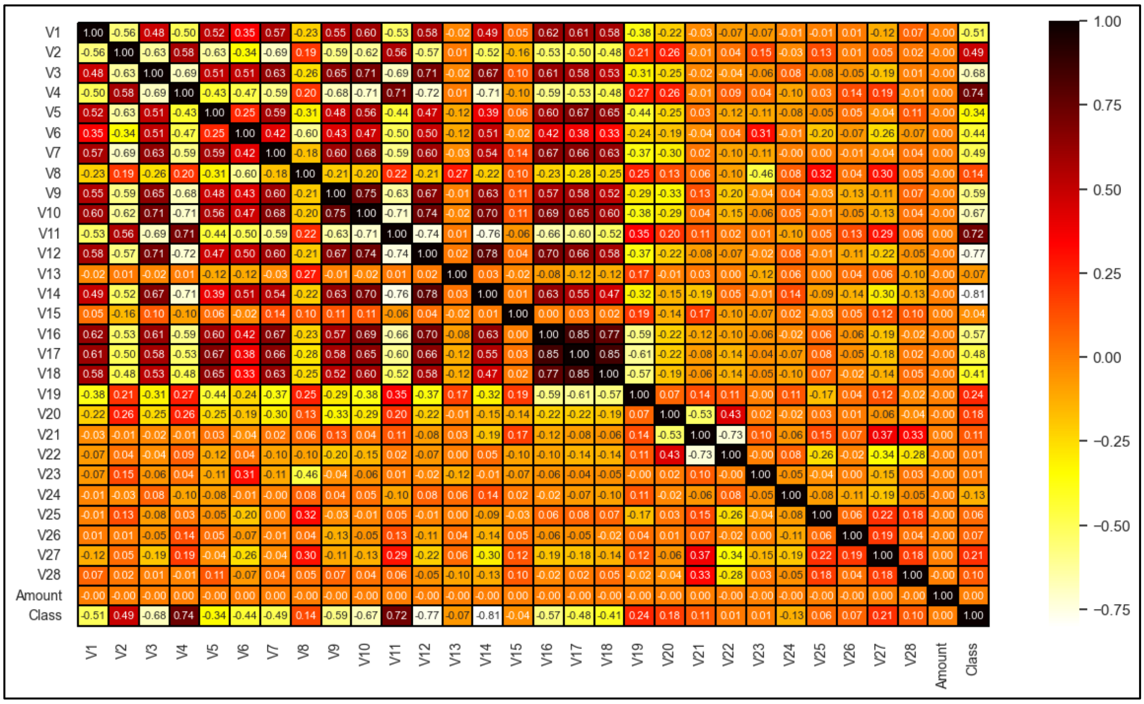

Figure 2 – Features correlation heatmap

FINAL PROJECT FOR DEEP LEARNING AND REINFORCEMENT LEARNING COURSE

VARIATIONS OF DEEP LEARNING MODELS

The dataset was divided in training and testing data, considering 20% of the dataset for testing. Three neural network model variations were tested to check which one had the best performance for this case. All of the models had same training and testing set.

MODEL 1

The first neural network model has:

- **Input layer**: with 29 inputs (all of the features).
- **Hidden layer (1)**: with 64 neurons and ReLu activation.
- **Output layer**: with 1 output and sigmoid activation.
- **Optimizer**: adam.
- **Training EPOCHS**: 10.

With this model it was possible to reach 96,05% accuracy in the test set.

MODEL 2

The second neural network model was based on model 1 with addition of mini-batches and data shuffle for training.

- **Input layer**: with 29 inputs (all of the features).
- **Hidden layer (1)**: with 64 neurons and ReLu activation.
- **Output layer**: with 1 output and sigmoid activation.
- **Optimizer**: adam.
- **Mini-batches for training.**
- **Data shuffle for training**.
- **Training EPOCHS**: 10.

With this model it was possible to reach 96,10 % accuracy in the test set.

MODEL 3

The third neural network model was based on model 2 with addition of more two hidden layers.

- **Input layer**: with 29 inputs (all of the features).
- **Hidden layer (1)**: with 64 neurons and ReLu activation.
- **Hidden layer (2)**: with 128 neurons and ReLu activation.

- **Hidden layer (3)**: with 64 neurons and ReLu activation.
- **Hidden layer (4)**: with 32 output and ReLu activation**.**
- **Output layer:** with 1 output and sigmoid activation.
- **Optimizer**: adam.
- **Mini-batches for training.**
- **Data shuffle for training**.
- **Training EPOCHS**: 30.

With this model, it was possible to reach 96,21% accuracy in the test set.

Check Table 2 for comparison between the three models.

	Model 1	**Model 2**	**Model 3**
Number of hidden layers	1	1	4
Activation function of hidden layers	ReLu	ReLu	ReLu (for all 4)
Data shuffle?	No	Yes	Yes
Mini-batch training?	No	Yes	Yes
Epochs	10	10	30
Time to train the model	96,05%	96,10%	96,21%
Test accuracy	4m 11.7s	4m 45.8s	12m 32.1s

Table 2 – Models comparison

KEY FINDINGS

With model number 3 the best accuracy was reached. With this model is possible to predict if the transaction is fraud or not with 96,21% accuracy. Comparing the performance of each model it’s possible to check that for this problem, a few points are very critical for the model performance, such as:

- The use of mini-batches and data shuffling, didn’t have a big impact in the accuracy or in the time to train; In some cases, the mini-batches approach offers several benefits, for example, faster convergence, reduced memory requirements, improved generalization, adaptative learning rate and others.

- It's possible to check how much the number of epochs can impact on the performance of the model.
- The third model had the higher accuracy value, 96,21%, to achieve that value the training considered 30 epochs. However, it was possible to check that in epoch number 15 the accuracy in the training set was already very close to the final value.

POSSIBLE FLAWS IN THE MODEL, ACTION PLAN TO REVISIT AND DIFFERENT MODELING TECHNIQUES

96,21% accuracy with this computing time and data size is acceptable, however, it's still possible to improve. To improve the model, a few points can be checked.

- Increase model complexity: adding more layers and/or more neurons in the existing layers.
- Check possibility of more data for model training.
- Regularization: add L2 regularization to reduce overfitting and improve generalization.
- Hyperparameter adjust: experiment with different learning rates, mini-batch size, epoch numbers and others.
- Increase dataset input: acquire more training data.
- Dropout layer: add dropout layer to reduce overfitting and improve time for training.

It's possible to change the model to use more complex tools as well.