Findings and research based off Homework Assignment 6.1

I think that there are lots of things deep autoencoders are good at, and one of them is credit card fraud. When there is a massive amount of data that needs to be compressed, I think that they make a great fit to solve the problem at hand. When thinking about the way that autoencoders are cramming tons of data to make predictions, I Think that this model type works the best in this situation. When thinking about fraud detection, there are lots of things one has to take into consideration, and behind the scenes’ is where lots of the data input has to be configured. In the realm of fraud detection, I think that deep autoencoders are the best model to use.

Some limitations of the Autoencoder are that autoencoders are extremely complex when taking all of the data. That being said, in the case of credit card fraud, the data does have to be precise and accurate. Considering that the data needs to be compressed for Credit card fraud. On top of that, credit fraud data sets are usually imbalanced. The number of transactions that happen during a regular business day is ton, and the ability to properly identify fraud keeps getting harder and harder. Auto-encoders need really good data sets to make these relationships and be able to figure out which transaction is fraud. The last drawdown of autoencoders is that they are operating in this "black box". The models’ actions are very hard to interpret, hence the need to decode the model after the predictions are being made. This process also makes it harder for the fraud detection to be as real time as possible, as the data does have to be de-coded for people to be able to read.

Some issues that present themselves when working with autoencoders and credit card fraud is the bias towards the non-fraudulent data. When working with this type of data, it is important to make sure that the model is not biased towards the non-fraudulent data. This is important because the model will not be able to detect the fraudulent data if it is biased towards the non-fraudulent data. A thought I had when building my model was that I should make the model really good at tracking non-fraudulent data, so the model was able to detect non-fraudulent data extremely well. Then when the model is introduced to fraudulent data, the model will be able to detect the fraudulent data with ease. But considering the nature of the compression of the data, the Deep autoencoder was able to have 95% accuracy when predicting the two classes, however it becomes tricky when looking at the F1 score, precision, and recall. This is where the model was not able to perform as well as I would have liked.

During the evaluation stage, looking at the models report, the model had a precision of 3.31%. With that low of a score the model is not able to detect the fraudulent transactions, 3.11% of the transactions flagged as fraudulent by the model are actually fraudulent. So the model is over flagging. My model’s recall was at 89.80% which is a okay score, and the model does do somewhat of a good job at detecting the non-fraudulent data. The F1 score was at 6.00% which is not a good score. F1 score is a really good indicator of how well a model is performing because the model even though having a high accuracy and a high recall, the model is not able to detect the fraudulent data accurately, and the model is just over flagging the data as being fraudulent.