**Ethereum Fraud Detection**

Ethereum is crypto currency and it usage is vastly increased. So, the chances of fraud transaction might spread on. To reduce the risk factor current model helps to identify transaction type as fraud or valid and help the provider to avoid loss.

Data:

We have found dataset from Kaggle. This dataset contains rows of known fraud and valid transactions made over Ethereum. Dataset has been described with address, various transaction time and amount details with aggregate functions(min,max,avg).

Data Cleaning Report:

By reviewing dataset, initially some unwanted rows have been removed.

* Later we have observed few columns has only values as ‘0’, so these details will hamper model and not required, so those rows have been removed.
* It was checked for missing values. we have found 829 rows has missing values and need to be treated. Further observation showed provides that rows with missed data are belong to Fraud transaction rows.
  + 1. It can be replaced with zeroes, as it belongs to fraud but a chance that model will train focusing on current values. So, this method has been not chosen.
    2. By analysing, we had filled the NaN values with Mean value during Data pre-processing process.

Exploratory Data Analysis:

To check correlated functions between columns we used Heatmap.

* 1. Initial we checked Heatmap with column including zeroes, so it shows zero correlation and being empty can make affect the model.
  2. So next columns have been removed and performed heat map again, now we found exact relation between columns.

Since dataset has provided with aggregate values like min, max, avg for Time transactions and Amount transactions, it already helped to find correlation without adding new features.

Data Pre-Processing:

By checking class balance on dataset, we have found Dataset is imbalanced. So, to fix the class we have applied scaler to dataset.

Using Pipeline module, we have filled Nan values with Mean for missing values and with scaler function class imbalance has been treated.

Once dataset has been fixed, we have added required features to ‘X’ and flag value to ‘y’ by converting flag to categorical column.

Now the pre-processed has been prepared and We have used train test split method and divided data into 70/30 for training and testing the data.

Modeling:

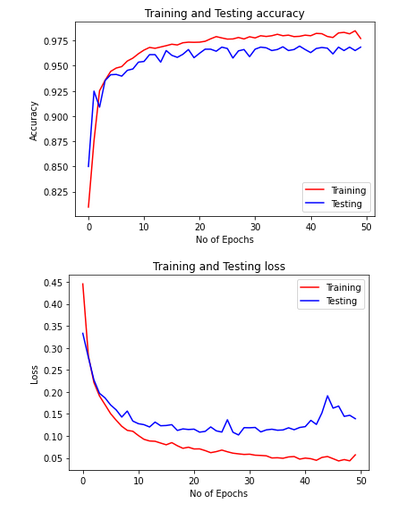
Now Data has been ready and we have applied Deep learning Keras model on the dataset and re-checked performance by comparing with Random forest classifier.

* We used sequential Keras model by giving input as 38 representing 38 columns and added Flatten layer on it.
* Further 3 hidden Layers has been added by using ‘Relu’ activation method.
* Final output layer has been taken with 1 output, so model gives only two outputs (either fraud or not)
* In Final layer we have used sigmoid function as it shows between 0 or 1.

To compile we have used Optimizer and for calculating Loss we have used Binary cross entropy as it output results in two variants.

We processed model in batch to deal with any limitations if exist and have applied 50 epochs to train the model better.

Be using plot we have seen the variation in epoch. Attached final Result.



The Keras Model has given accuracy around 97-98%.

To check the model, we have used RandamforestClassifier model on the pre-processed dataset and it also gives 98% accuracy.