Machine Learning Engineer Nanodegree

Capstone Proposal

Mohammed AlAyyaf

December 27th, 2018

Proposal

Domain Background

Fraud detection is a challenging problem. The fact is that fraudulent transactions are rare; they represent a very small fraction of activity within an organization. The challenge is that a small percentage of activity can quickly turn into big dollar losses without the right tools and systems in place. Criminals are crafty. As traditional fraud schemes fail to pay off, fraudsters have learned to change their tactics. The good news is that with advances in machine learning, systems can learn, adapt and uncover emerging patterns for preventing fraud.

Problem Statement

The Credit Card Fraud Detection Problem includes modeling past credit card transactions with the knowledge of the ones that turned out to be fraud. This model is then used to identify whether a new transaction is fraudulent or not. Our aim here is to detect as much of the fraudulent transactions as possible, while minimizing the incorrect fraud classifications.

Datasets and Inputs

The datasets contain transactions made by credit cards in September 2013 by European cardholders. This dataset presents transactions that occurred in two days, where we

have 492 frauds out of 284,807 transactions. The dataset is highly unbalanced, the positive class (frauds) account for 0.172% of all transactions.

It contains only numerical input variables which are the result of a PCA transformation. Features V1, V2, ... V28 are the principal components obtained with PCA, the only features which have not been transformed with PCA are 'Time' and 'Amount'. Feature 'Time' contains the seconds elapsed between each transaction and the first transaction in the dataset. The feature 'Amount' is the transaction Amount, this feature can be used for example-dependent cost-sensitive learning. Feature 'Class' is the response variable and it takes value 1 in case of fraud and 0 otherwise.

Data Source: https://www.kaggle.com/mlg-ulb/creditcardfraud

Solution Statement

Given the unbalanced nature of this dataset, we most likely need to find a way to rescale/resample the dataset to be more balanced and in a way that allows us to approach the problem as a normal balanced classification problem, I will need to look into resampling methods/techniques that would best suit this problem, most probably it would be under-sampling, which deletes instances from the over-represented class in order almost 50/50 representation for both classes. After that, I will test some supervised learning algorithms to find the best possible algorithm.

Benchmark Model

Looking at the other scores achieved from other Kaggle contributors, I would say that recall, f1-score and the accuracy should be no less than 80%, especially as this is a case where it is considered sensitive to miss any fraudulent transactions.

Evaluation Metrics

Since our target is to catch the fraudulent transactions, and hence we would rather wrongly catch suspicious transactions than to mistake fraudulent transactions as normal

ones, then Recall (the ratio of correctly predicted positive observations to all the observations in actual class.) should be our main metric. Yet, we will still aim to have a somewhat accurate model as much as possible.

Project Design

First of all, I will need to gather the data from its source (Kaggle), then I would need to take a deeper look into its nature and what preprocessing is required, which then I can use some EDA to explore the dataset, from looking at the class distribution of the data it is obvious that it will require some handling with class imbalance issue, I assume I will need to look more for resampling methods that would help me convert the dataset into a more balanced one, as it stands, my main candidates are SMOTE (synthetic minority oversampling technique), under-sampling or over-sampling, after that I will most likely start experimenting with the with the models and their hyper-parameters, until an acceptable score is reached.