Project proposal

-Submitted By Ritu Ramakrishnan

Tittle: Credit Card Fraud detection using machine learning algorithms

Project Summary: A credit card is a credit facility offered by banks that lets people borrow funds up to a pre-determined credit limit. It allows customers to make purchases of goods and services. Because of the lack of funds, a credit card is one of the most frequently utilized financial products to make purchases online and pay-outs such as fuel, grocery shopping, TVs, travel, shopping bills, and so on. Credit cards are the most valuable because they provide numerous advantages in the way of points when used for various transactions. Today, several types of credit card fraud are expected, including losing cards, card abuse, and merchant abuse. In these cases, the Merchant loses two times the product or service. So, there is a need to protect merchants from credit card fraud. In this project, I will outline the different methods available to detect fraud and the best way to reduce the merchant impact.

Project Background: The following are the situations where the customers can file chargebacks for fraudulent or unauthorized charges on the account. Orders that were never delivered, damaged or defective products. Incorrect orders, incorrect charges. Business is grown through a process called a chargeback. It could happen due to unsatisfied customers or fraudulent scenarios. In this project, we will cover the reduction of chargebacks through the early detection of fraudulent transactions. Chargebacks have long-term repercussions for merchants. Each time a chargeback is filled, the card processor levies a fee for administrative causes for the merchant losses the revenue and future profit. Exceeding the chargeback threshold set by credit-by-credit card processors can result in massive fines. Exceeding the second level of chargeback threshold may result in merchant account termination.

Potential Challenges: The fraud protections are not straight forward because of the following challenges. They are.

- Challenging fraud patterns over the time
- Class imbalance
- Model Interpretations
- Feature generations can be time consuming.
- Collecting large amount of the data.

These challenges make the fraud detection in the transaction as a challenging to derive a model to predict all kinds of fraud transaction.

Past efforts for credit card fraud detection:

These are the some of the existing algorithms and their challenges.

- **Decision Tree**

  The decision tree method employs a decision tree logic-generated similarity tree. A similarity tree is defined by nodes and leaves that contain attributes and factors. They are significantly more unstable than other decision predictors.

- **Genetic Algorithms and A Range of Additional Algorithms**

  Algorithms can detect fraud by employing predictive methods. The algorithms work by establishing a set of logic-based rules. This allows the data to be classified as either non-suspicious or suspicious. They can be difficult to debug and computationally costly.

- **Neural Networks**

  Credit card fraud can also be effectively combated using neural networks. The drawback of this approach is that it relies on clustering techniques, which can only be aggregated by account type. The disadvantages are black box, duration of development is more.

- **Logistic Regression**

  Logistic regression is a data analysis technique that uses mathematics to find the relationships between two data factors.If we use the wrong variables as input, the outcome will suffer. The model of logistic regression estimates only categorical data, which may limit our ability to obtain a precise result from our data.

- **Random Forest Classifier**

  The random forest classifier is useful for solving regression and classification problems. Random forest's main limitation is that a large number of branches can end up making the technique too slow and unproductive for real-time predictions. In general, these methods are quick to train but slow to generate predictions once trained.

- **K-Nearest Neighbor Algorithms**

  The K-Nearest Neighbor Algorithm, or KNN, uses existing instances to classify new cases based on their similarity. The disadvantages are requires high memory, accuracy depends on the quality of the data and for large data the prediction might be slow.

- **Support Vector Machines (SVMs)**

The Support Vector Machine is a useful statistical learning method for detecting credit card fraud. It performs poorly when the data set contains more noise, i.e. target classes overlap.

Proposed Solution: The merchants are responsible for the reduction of the risks due to chargebacks. Having a continuous monitoring system to detect potential chargebacks and a streamlined chargeback strategy is crucial. This project will outline the available strategies (Decision trees, K nearest neighbors, logistic regression, support vector machines, random forest classifier, XG Boost) to effectively determine fraudulent transactions and compare their performances.

Characteristics Of the Data Source:

The simulated credit payment data - set having legitimate and fraudulent transactions from January 1st, 2019, to December 31st, 2020. It protects the credit of 1000 customers who transact with such a pool of 800 vendors. This dataset contains 492 frauds out of 284,807 transactions that occurred over the course of two days. The dataset is highly unbalanced, with positive transactions accounting for 0.172% of all transactions. The data set collected has the 556k columns with the merchant, category, customer details like name, address, gender and more columns which are used for the credit card transactions for the purchase of the items. The below mentioned data set will be initially used for the comparison of the credit card protection model for analysing the performances.

Requirements:

We use the following libraries and frameworks in credit card fraud detection project.

- Python – 3.x
- NumPy – 1.19.2
- Pandas
- Seaborn
- Scikit-learn – 0.24.1
- XG Boost
- Matplotlib – 3.3.4

Timeline:

- Gathering requirements 1 week is required.
- Designing the fraud protection system 2 weeks is required.
- Testing and debugging 1 week is required.
- Implementation and analysis 2 weeks is required.