**Business Problem**

Customers expect to have a secure banking experience, and providing a solid fraud-detecting system has become an integral part of the credit card industry. I would like to build an automated system that detects fraudulent patterns and flags and temporarily blocks such transactions for customers to review, approve, or deny.

The main aim of this project is to build a prediction system using different models to track various patterns, abort abnormal transactions, and notify customers about suspected fraudulent activity on their credit cards.

**Background/ History**

As the world progresses toward digitization, we rely on digital payments. While digital payments are on the rise, frauds in digital payments especially using a credit card, are increasing, and cybersecurity has become a crucial part of the credit card industry.

Credit Card Fraud Prediction System will bear considerable benefits to the customers and credit card firms so that we can reduce financial risks and losses due to these frauds.

As the system develops to predict fraudulent activities on credit card transactions, fraudsters will create new ways of doing things. So, we can not rely on standard rules for identifying fraudulent patterns.

We need to develop a system that gets trained with new patterns and identifies fraudulent activities as recent frauds develop.

**Datasets**

The datasets used in this project will be sourced from Kaggle, a public-domain dataset. I am planning to use data from two different sources and utilize the following data:

trans\_date\_trans\_time – Transaction date and time.

cc\_num – Credit Card Number.

merchant – Merchant name where the transaction has been done.

category – Category name of the transaction.

amt – Amount transacted.

first – First name of the credit card user.

last - Last name of the credit card user.

gender – Gender of the credit card user.

street – Street address of the credit card user.

**Data Preparation**

I have combined the fraud train and fraud test datasets as part of Data preparation; I have done this step to provide the visualizations with the complete dataset. In the second step, I have dropped unwanted columns, i.e., Unnamed: 0, as this is not useful for our prediction models.

**Visualizations**

**Column Distribution:**

Chart, waterfall chart, box and whisker chart

Description automatically generated

**Fraud vs. Gender:**

Chart, bar chart

Description automatically generated

**Fraud per Category:**

**Chart, bar chart

Description automatically generated**

**Methods**

As part of this project, I will be building the below models:

1. Logistic Regression: This is a Supervised Learning algorithm. It is used to predict default events. As we are interested in credit card fraud prediction, we are using Logistic Regression, which is best suited for prediction and classification problems.
2. Random Forest: This is a Supervised Learning algorithm. This is widely used for Classification and Regression problems. It builds multiple decision trees and takes a majority vote to classify.

Random Forest algorithm is used to classify fraud transactions in real-time or batch processes.

1. Decision Tree Approach: This is a Supervised Learning used to make predictions based on how the previous set of questions is answered.

Constructing a proper decision tree can be helpful in stock price prediction. We can see if the price of the stock will rise or fall.

1. XG Boost Algorithm: XGBoost is a decision-tree-based ensemble Machine Learning algorithm. It uses a [gradient-boosting](https://en.wikipedia.org/wiki/Gradient_boosting) framework.

This is a popular and open-sourced gradient-boosted trees algorithm. As this algorithm has gained popularity over time, I plan to implement it in this project and see if it performs better than the other two algorithms listed above.

In the above-proposed models, I will be implementing SMOTE technique to increase the number of records in the datasets in a balanced way to overcome issues with imbalanced datasets.

**Analysis**

As the dataset was imbalanced, which would impact the prediction outcomes, I used encoding of the categorical columns and then applied a scaling mechanism to get a balanced dataset.

I have implemented four prediction models, i.e., Logistic Regression, Random Forest, Decision Tree, and XG Boost models. Upon building and evaluating all the models, I used the accuracy score to see which model performed better.

The below tables provide the accuracy scores of all the implemented models:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Logistic Regression | Random Forest | Decision Tree Classifier | XGBoost |
| Accuracy Score | 94.89 | 95.96 | 96.71 | 99.56 |

With an accuracy score of 99.56%, the XG Boost model is the best-suited model for this dataset.

**Assumptions**

As we have taken a dataset for a month and simulated, we will have an imbalanced dataset; we need to perform SMOTE analysis or other similar ones to balance the dataset.

**Limitations**

As we are sourcing the data from public websites, which are not the latest and not real-world data, our analysis will not be able to provide exact results, but the model developed can be used with real-world data in the future.

**Ethical Considerations**

Though this project's credit card transaction data is simulated, it can match real-time PII data.

Credit card fraud detection might not work as expected with smaller datasets. As we are using a smaller dataset, I am not confident that the model accuracy is as expected.

**Challenges**

As we are using the dataset for 1 Jan 2019 - 31 Dec 2020, we might be missing the newer patterns that have come up post-Dec 2020.

As we are using public domain datasets, we cannot expect that this covers all the possible fraudulent transactions.

While working with the real-time datasets for credit card fraud detection, we will be working with massive datasets which might need higher CPU capacities and scaling of the algorithm implemented. We need to keep in mind the scaling of the model to support the substantial real-time datasets.

**Implementation Plan**

Below are the steps which I will be following as part of the implementation plan:

* Data Reader
* Data Cleansing and Transformation
* Data Visualizations
* Build and fit various Models.
* Calculate accuracy
* Compare the models based on accuracy.

**Future Uses/Additional Applications**

To make this application easily accessible to Bankers and Credit Card industry, we can implement the following:

1. Provide an API that will give data in JSON format with whether the transaction is fraudulent or not in the API response.

In addition, we need to find a way to source a user's transactions from various sources.

**Recommendations**

As we are working with a smaller dataset, I would like to run the same models with larger datasets to see how the models perform and fine-tune them if required.

**References**

* SHENOY, K. (n.d.). Credit Card Transactions Fraud Detection Dataset. Kaggle. <https://www.kaggle.com/datasets/kartik2112/fraud-detection?select=fraudTrain.csv>
* XIANG, N. (n.d.). Credit Card Fraud Analysis and Modeling. Kaggle. <https://www.kaggle.com/code/nathanxiang/credit-card-fraud-analysis-and-modeling>

**Appendix**

1. PII Data: It stands for Personal Identifiable Information. PII data is any data that could identify a particular individual.
2. SMOTE: It stands for Synthetic Minority Oversampling TEchnique. It is a statistical technique for increasing the number of cases in a dataset in a balanced way.
3. XG Boost Model: It stands for Extreme Gradient Boosting. It is used for regression predictive modeling with an efficient way of gradient boosting.

**Questions and Answers**

1. What is Credit card fraud?
2. What is Credit card simulated data?
3. Why do we need to simulate the credit card data?
4. What is XG Boost Model?
5. What is the difference between the Decision Tree model and XG Boost model?
6. Which model performs better between the Decision Tree model and the XG Boost model?
7. Will these models withstand the vast datasets?
8. Can we provide real-time fraud detection using these implemented models?
9. Will these implemented models work with data from various credit card firms?
10. Can these models be extended for fraud detection in any other sector?