Credit Card Fraud Detection Summary

Problem Statement

Predicting fraudulent credit card transactions with the help of machine learning models.

Business Problem

Banking fraud poses a significant threat to the bank's goal of retaining high profitable customers, since the trust and credibility of the bank and of the customer is lost. This leads to substantial financial losses.

Advantage of Machine Learning

In the banking industry, credit card fraud detection via ML becomes a necessity for the banks to put proactive monitoring and fraud prevention mechanisms in place. This helps the institutions to reduce time-consuming manual reviews, costly chargebacks and fees, and denials of legitimate transactions.

Dataset

The data set includes credit card transactions made by European cardholders over a period of two days in September 2013. Out of a total of **2,84,807 transactions, 492 were fraudulent.** This data set is highly imbalanced with the **positive class (frauds) accounting to 0.172%.** The data set is also been masked with PCA to maintain confidentiality. There is a total of **31** columns, out of which **28** variables are PCA transformed and the remaining three are **Amount**, **Time** and **Class**, where '**Amount**' is the transaction amount, '**Time**' contains the seconds elapsed between first transaction in the data set and the subsequent transactions, and the feature '**Class**' is the target feature which takes **1 as fraud cases** and **0 as non-fraud cases**.

Approach

The approach will be based on a pipeline consisting of the following six steps:

1. Data Understanding
2. Exploratory Data Analysis
3. Data Preparation
4. Model-Building
5. Model Evaluation
6. Cost Analysis

A brief summarization of the pipeline is as follows:

1. **Data Understanding:** Loading and understanding the features present in it such as, number of rows and columns, their data types, etc. Further, check for any duplicate data and missing values across rows and columns.

2. **Exploratory Data Analysis:** Check the imbalance percentage of the class labels and plot them for the number and the percentage of fraudulent vs non-fraudulent transactions. There will be no specific univariate or bivariate analysis on the predictors as majority of the columns are already PCA transformed, which takes care of the scaling and transforming the features into Gaussian distribution. Additionally, plotting a scatter chart between the raw features, such as Amount and Time with the class labels independently to understand the distribution of the classes with them. Lastly, drop any unnecessary column, if any.

3. **Data Preparation:** Split the data into train and test with stratified feature as there is a huge imbalance in the dataset. Note, dummy variables will not be created as there are no categorical features. Check for skewness for all the predictors by plotting histograms. If present, transform them using log or power transformation to make them gaussian in nature, followed by scaling them, if required. This will be used mostly for the raw features.

4. **Model-Building:** Build multiple models for each of the two cases, namely Imbalanced dataset and Balanced dataset. This will be done to understand the importance of balancing the dataset w.r.t to the classes. Start with the simplest model, i.e., logistic regression, because of the principle Occam's Razor.

    a. **Model building with imbalanced dataset:** Perform k-fold cross validation on the train data and then tune the hyperparameters using GridSearchCV or RandomizedSearchCV. Build the model with the optimal hyperparameters obtained and then predict on the test set. Evaluate the model on the relevant scores and then try different other models such as, decision trees, random forest, etc.

    b. **Model building with balanced dataset:** Perform **stratified k-fold cross validation** using an appropriate value of k so that the minority class is correctly represented in the test folds. Post this, apply few of the class balancing techniques such as, **Random oversampling, SMOTE, ADASYN**, etc. to handle class imbalance. Proceed with hyperparameter tuning as before and build model with optimal hyperparameters. Predict on test and then explore other models such as, **KNN, SVM, RandomForest, XGBoost**, etc.

5. **Model Evaluation:** Evaluate the best model on the metrics, such as Roc-Auc score, precision and recall, since capturing the fraudulent cases is more important than identifying non-fraudulent cases. Accuracy can be considered a good metric only if the data is balanced.

6. **Cost Analysis:** Since, this data is from a bank which is huge, evaluating the cost benefit will be based on the **recall / sensitivity**, since the amount pertaining to the fraudulent transactions can be huge thousands of dollars.