Credit Card Fraud Detection Capstone Project

Problem Statement: Fraudulent activities have surged dramatically, with approximately 52,304 cases of credit/debit card fraud reported in FY'19 alone. Due to this significant rise in banking frauds, it has become essential to timely detect these fraudulent transactions to protect both consumers and banks, which are losing their creditworthiness daily. Each fraudulent credit card transaction represents a direct financial loss to the bank and negatively impacts overall customer satisfaction.

Aim: This project's aim is to identify and predict fraudulent credit card transactions using machine learning models.

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

Approaches: The problem-solving approach can be divided into the following parts:

1. **Data Understanding, Preparation, and EDA:** Initial examination of the Kaggle dataset reveals a highly imbalanced nature. The positive class (frauds) constitutes only 0.172% of all transactions:
    - **Class 0:** 284,315 transactions
    - **Class 1:** 492 transactions

Class is the target variable to be predicted, where 0 signifies a normal transaction and 1 signifies a fraudulent transaction. Features V1, V2, … V28 are principal components obtained through PCA, except for 'Time' and 'Amount', which remain untransformed. 'Time' represents the seconds elapsed between each transaction and the first transaction in the dataset, while 'Amount' is the transaction amount, suitable for cost-sensitive learning. Since PCA-transformed variables are Gaussian, normalization isn't required. Basic EDA techniques like correlation and boxplots for outliers can be employed. Transformations to address data skewness (e.g., Box-Cox, Log transformation, Yeo-Johnson) can also be used.

2. **Class Imbalances:** Conventional oversampling methods won't be employed as they don't add new information, and under-sampling will be avoided due to information loss. Instead, two class imbalance handling techniques will be used:
    - **SMOTE:** Generates new data points vectorically between two minority class data points.
    - **ADASYN:** Similar to SMOTE, but generates synthetic data for harder-to-learn minority examples, rather than easier ones.
3. **Model Selection and Building:** Models will be built using a train-test split (with at least 100 class 1 rows in the test split) and stratified split (80-20 ratio). Various ML models will be tested for performance on imbalanced data:
    - **Logistic Regression:** Best for linearly separable data.
    - **KNN:** Simple and interpretable but computationally intensive for large data.
    - **Decision Tree:** Intuitive output but prone to overfitting if unchecked.
    - **Gradient Boosting Machines/Trees:** Reduce errors of earlier models.

- **XGBoost:** Extended version of gradient boosting with additional features like regularization and parallel tree learning for optimal split.

Hyper-parameter tuning will start with logistic regression, then move to decision trees, and so on. Ensemble models and techniques like voting classifiers, bagging, and boosting will also be employed.

4. **Hyper-parameter Tuning:** For imbalanced or small datasets, K-Fold Cross Validation will be used for performance evaluation. Stratified K-Fold Cross Validation ensures each fold represents all data strata. Grid search will be used for small datasets, while randomized search will be employed for large datasets.
5. **Model Evaluation:** The sklearn.metrics.roc_auc_score will be used, as it is suitable for highly imbalanced datasets. ROC curve, which plots TPR and FPR, will help determine the best cut-off. Unlike confusion matrix, ROC curve considers all threshold values.
6. **Cost-Benefit Analysis:** Depending on the use case, high precision or high recall is prioritized. Banks with smaller transaction values need high precision to label relevant transactions as fraudulent, whereas banks with larger transaction values need high recall to detect fraudulent transactions. The ultimate goal is to determine the financial value saved with the best-performing model.