plagirsm FindDefault: Predicting Credit Card Fraud Documentation

**Introduction**

FindDefault is a machine learning solution designed to assist credit card companies in detecting and preventing fraudulent transactions. By analyzing historical credit card transaction data, FindDefault aims to predict whether a transaction is fraudulent or legitimate, thereby reducing financial losses for both cardholders and financial institutions.

**Problem Statement**

Credit card fraud poses a significant threat to the financial security of cardholders and can result in substantial losses for credit card companies. FindDefault addresses this problem by leveraging machine learning techniques to identify fraudulent transactions in real-time.

**Approach**

*Exploratory Data Analysis (EDA****)***

* Libraries Used: Pandas, Matplotlib, Seaborn
* During EDA, we thoroughly examine the credit card transaction dataset to gain insights into its structure and characteristics.
* We use descriptive statistics, visualizations, and data quality checks to identify patterns, relationships, and trends within the data.

**Data Cleaning**

* Libraries Used: Pandas
* Data cleaning involves standardizing the format, handling missing values, and addressing outliers to ensure the dataset is accurate and consistent.
* By cleaning the data, we prepare it for further analysis and model training.

**Dealing with Imbalanced Data**

* Libraries Used: Scikit-learn
* The credit card transaction dataset is highly imbalanced, with fraudulent transactions being a minority class.
* We employ techniques such as oversampling, undersampling, or synthetic data generation to balance the dataset and prevent model bias towards the majority class.

**Feature Engineering**

* Libraries Used: Scikit-learn
* Feature engineering involves creating new features or transforming existing ones to improve the predictive performance of machine learning models.
* By selecting relevant features and engineering them appropriately, we aim to capture meaningful patterns in the data.

**Model Selection**

* Libraries Used: Scikit-learn
* We evaluate various machine learning algorithms, such as Support Vector Machine (SVM), K-Nearest Neighbors (KNN), Decision Tree, Logistic Regression, and Random Forest.
* *Hyperparameters* of these models are tuned using techniques like GridSearchCV to optimize performance

**Model Training**

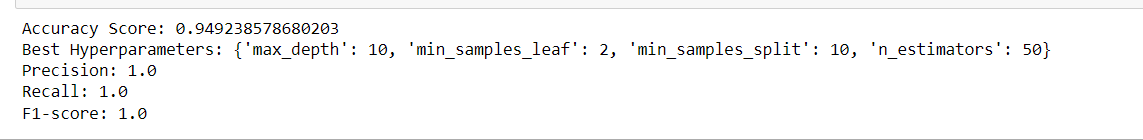
* Libraries Used: Scikit-learn
* The dataset is divided into training and testing sets using train\_test\_split() function.
* We train the selected model using the training data to learn patterns and relationships within the data.

**Model Evaluation**

* Libraries Used: Scikit-learn
* We evaluate the trained model's performance using metrics such as accuracy, precision, recall, F1-score, and confusion matrix.
* These metrics help us assess the model's ability to correctly classify fraudulent and non-fraudulent transactions.

***Success Metrics:***

* The accuracy of the model on the test data set should be &gt; 75% (Subjective in nature)



* Add methods for Hyperparameter tuning.

*Hyperparameter Tuning:*

* Methods Used: GridSearchCV
* Hyperparameters of machine learning models are tuned using techniques like GridSearchCV.
* GridSearchCV exhaustively searches through a specified hyperparameter space to find the optimal combination of hyperparameters that maximizes model performance.

**Pipeline:**

* This Python script demonstrates the utilization of a machine learning pipeline for credit card fraud detection.
* Leveraging the scikit-learn library, the pipeline integrates preprocessing steps, such as feature scaling.
* The pipeline also includes a trained Random Forest Classifier model.
* The purpose of the pipeline is to predict fraudulent transactions.

**Purpose:**

* The purpose of this script is to automate the process of loading data, preprocessing features, and making predictions using a pre-trained machine learning model.
* By encapsulating these steps within a pipeline, it ensures consistency and reproducibility.
* Additionally, it simplifies the deployment of the fraud detection model.

*Implementation:*

* *Data Loading:* The script loads data from an Excel file ('predictions.xlsx') containing features and labels (fraudulent or non-fraudulent transactions).
* *Pipeline Definition*: A scikit-learn pipeline is created, consisting of two main steps: feature scaling using StandardScaler and model prediction using a pre-trained Random Forest Classifier loaded from a .pkl file.
* Model Evaluation: The pipeline is fitted to the data, and predictions are made on the same dataset to evaluate model performance. The accuracy of the model is calculated using scikit-learn's accuracy\_score function.
* *Output****:*** The script prints the accuracy of the model on the provided dataset.

**Conclusion:**

* This script demonstrates a streamlined approach to deploy a pre-trained machine learning model for credit card fraud detection. By encapsulating preprocessing and prediction steps within a pipeline, it ensures efficient and consistent model deployment, facilitating real-time fraud detection in financial transactions
* Perform model validation

*Model Validation:*

* Model validation is performed to ensure the model's ability to generalize to new, unseen data.
* Techniques such as cross-validation are utilized to assess the robustness and reliability of the trained model.
* By evaluating the model on different subsets of the data, we can identify any potential issues such as overfitting or underfitting.
* Validation metrics such as accuracy, precision, recall, F1-score, and ROC-AUC are used to measure the performance of the model on unseen data.