Credit Score Model Documentation

Introduction

The Credit Score Model predicts the likelihood of a credit cardholder defaulting on payments based on demographic, financial, and transactional attributes. This model aims to support data-driven decisions for credit risk management and customer segmentation in the financial sector. It employs advanced machine learning techniques to provide accurate predictions and actionable insights.

Features

The model utilizes the following key features extracted from the dataset:

- **Demographic Data**:
  - `SEX`: Gender of the cardholder.
  - `EDUCATION`: Educational background.
  - `MARRIAGE`: Marital status.
  - `AGE`: Age of the cardholder.
- **Financial Attributes**:
  - `LIMIT_BAL`: Credit limit assigned to the cardholder.
  - `BILL_AMT1-6`: Monthly billed amounts over six months.
  - `PAY_AMT1-6`: Monthly payments made over six months.
  - `default payment next month`: Target variable indicating default status.
- **Behavioral Data**:
  - Payment history across the last six months (`PAY_0`, `PAY_2`, etc.).

Model Process

The model follows a structured process for data preparation, training, and evaluation:

Data Preprocessing

1. **Data Cleaning**:
   a. Missing values imputed using statistical methods.
   b. Categorical variables encoded.
2. **Feature Engineering**:

    a. Created new features based on payment behavior and utilization rate.
    b. Normalized numerical columns for consistent scaling.
3. **Splitting Data**:
    a. The dataset is split into training (70%) and testing (30%) sets.

Training Methodology

- **Algorithm**: XGBoostClassifier was selected for its efficiency and ability to handle large datasets with complex patterns.
- **Hyperparameter Tuning**:
    - Performed grid search for parameters like `learning_rate`, `max_depth`, and `n_estimators`.

Model Algorithm

The Credit Score Model employs a **XGBoost Classifier**. This gradient boosting algorithm optimizes model performance by combining the predictions of weak learners to form a strong learner.

Reasons for Choosing XGBoost

- Handles missing values and outliers effectively.
- Provides built-in regularization to prevent overfitting.
- Supports parallel processing for faster training.

Model Output

The model produces the following outputs:

1. **Default Probability**:
    a. Probability scores for each user indicating their likelihood of defaulting.
2. **Predicted Default Status**:
    a. Binary classification (0: No Default, 1: Default).
3. **Feature Importance**:
    a. Ranked list of features contributing to the prediction.

Model Metrics

The model's performance is evaluated using the following metrics:

- **Accuracy**:
    - Measures overall correctness of predictions.
- **Precision**:
    - Indicates the proportion of true positive predictions.
- **Recall (Sensitivity)**:
    - Reflects the ability to detect all defaulters.
- **F1 Score**:
    - Harmonic mean of precision and recall.
- **ROC-AUC**:
    - Measures model discrimination capability across threshold levels.

Dependencies

- **Dataset:** The credit card dataset used for training and testing.
- **Libraries**:
    - XGBoost for model development.
    - Scikit-learn for evaluation.
    - Pandas and NumPy for data manipulation.
    - Matplotlib and Seaborn for visualizations.
- **Azure Machine Learning Services**:
    - Used for deployment and monitoring.

Deployment and Monitoring

The trained model is deployed on Azure Machine Learning platform with the following considerations:

- **Endpoint Setup**:
    - Real-time endpoint configured for prediction requests.
- **Performance Monitoring**:
    - Metrics like latency, throughput, and model drift monitored using Azure Monitor.
- **Versioning**:
    - MLflow used to track experiments and manage model versions.

User Guide

Steps to Interact with the Deployed Model

1. **Access the Endpoint**:
    a. Use the provided REST API endpoint for making predictions.
2. **Input Data Requirements**:
    a. Ensure the input data matches the format and schema of the training data.
3. **Retrieve Predictions**:
    a. Send a POST request with input data to the endpoint to receive predictions.
4. **Monitor Results**:
    a. Track model performance and outputs in real time.

Conclusion

The Credit Score Model enables effective credit risk management by predicting potential defaulters. The integration of demographic, financial, and behavioral data ensures a comprehensive understanding of user risk profiles, enhancing the bank's decision-making capabilities. By leveraging XGBoost and Azure Machine Learning, the model offers high performance, scalability, and reliability for real-world applications.