**Credit Score Category Calculator Documentation**

Overview

The Credit Score Category Calculator is a Python application designed to predict credit score categories based on various input parameters provided by users. It leverages machine learning techniques, specifically a Random Forest Classifier model, trained on historical credit data to make predictions. This documentation provides a detailed overview of the application, including its functionalities, implementation details, deployment process, testing, and validation procedures.

Table of Contents

1. Introduction
2. Installation and Setup
3. Data Preprocessing
4. Model Training
5. Model Evaluation
6. Graphical User Interface (GUI)
7. Deployment
8. Testing and Validation
9. Documentation

1. Introduction

The Credit Score Category Calculator serves as a tool for individuals and financial institutions to assess creditworthiness. By inputting various financial and personal parameters, users can obtain a predicted credit score category, aiding in decision-making processes related to lending, financial planning, and risk assessment.

2. Installation and Setup

Prerequisites

Python (version 3.6 or higher)

Required Python libraries: tkinter, pandas, scikit-learn, numpy

Installation Steps

Clone the repository or download the source code files.

Install the required Python libraries using pip:

Copy code

pip install tkinter pandas scikit-learn numpy

Ensure that the dataset (credit.csv) is available in the same directory as the Python script.

3. Data Preprocessing

In the preprocessing phase, the dataset undergoes several transformations to prepare it for model training. This includes:

Loading the dataset from the CSV file.

Dropping unwanted features such as IDs, names, and irrelevant columns.

Normalizing numerical features using StandardScaler to ensure uniform scaling.

Capping outlier values to mitigate their impact on model training.

One-hot encoding categorical features like 'Credit\_Mix'.

Label encoding the target variable 'Credit\_Score' to convert categorical labels into numeric format.

No null values were found

4. Model Training

The preprocessed dataset is split into training and testing sets. A Random Forest Classifier model is trained on the training data with the following hyperparameters:

n\_estimators: 100

min\_samples\_split: 2

min\_samples\_leaf: 2

max\_depth: None

bootstrap: False

random\_state: 42

5. Model Evaluation

The trained model is evaluated on the test set to assess its performance. The accuracy score is computed and printed to the console, providing insights into the model's predictive capability.

6. Graphical User Interface (GUI)

The application features a user-friendly graphical interface built using the Tkinter library. Users can input various financial parameters via text entry fields and select credit mix options from a dropdown menu. Upon clicking the "Predict" button, the predicted credit score category is displayed.

7. Deployment

The application can be deployed locally on any machine with Python installed. Users can execute the Python script to launch the GUI and interact with the application seamlessly.

8. Testing and Validation

Thorough testing and validation procedures are essential to ensure the reliability and accuracy of the deployed model. Various test cases should be executed to validate the application's functionality across different scenarios and input combinations.

9. Documentation

This comprehensive documentation serves as a guide for users and developers, providing detailed insights into the Credit Score Category Calculator application. It covers installation instructions, preprocessing steps, model training details, GUI functionalities, deployment guidelines, testing procedures, and more.

---------------------------------------