Credit Risk Prediction Model

This project aims to build a predictive model for assessing the risk of loan defaults using machine learning techniques. By analyzing historical loan and customer data, we will develop a logistic regression model to determine the likelihood of default based on key financial indicators.

The analysis involves preprocessing the data to handle missing values, performing exploratory data analysis (EDA) to understand relationships between features, and training the model on relevant attributes such as income, loan amount, and credit score. We will evaluate the model's performance using metrics like precision, recall, and F1-score to ensure its effectiveness in risk assessment.

Complete code below with piecewise implementation toward the end

In [16]:
```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import classification_report, confusion_matrix

class CreditRiskModel:
    def __init__(self, data_file):
        """Initialize by reading the dataset."""
        self.data = pd.read_csv(data_file)
        self.model = LogisticRegression()

    def preprocess_data(self):
        """Prepare data for model training."""
        self.data = self.data.dropna()  # Handle missing values
        X = self.data[['income', 'loan_amount', 'credit_score']]
        y = self.data['default']  # 1 if default, 0 otherwise
        self.X_train, self.X_test, self.y_train, self.y_test = train_test_sp

    def train_model(self):
        """Train the logistic regression model."""
        self.model.fit(self.X_train, self.y_train)

    def evaluate_model(self):
        """Evaluate the model's performance."""
        predictions = self.model.predict(self.X_test)
        print(confusion_matrix(self.y_test, predictions))
        print(classification_report(self.y_test, predictions))
```

```python
# Usage
if __name__ == '__main__':

    crm = CreditRiskModel('~/python_projects/financial-data-analysis/loan_da
    crm.preprocess_data()
    crm.train_model()
    crm.evaluate_model()
```

```
[[2 0]
 [0 1]]
              precision    recall  f1-score   support

           0       1.00      1.00      1.00         2
           1       1.00      1.00      1.00         1

    accuracy                           1.00         3
   macro avg       1.00      1.00      1.00         3
weighted avg       1.00      1.00      1.00         3
```

Test data Used in the .csv below, above results

income,loan_amount,credit_score,default 55000,15000,700,0 60000,20000,720,0 45000,10000,680,1 75000,30000,740,0 50000,25000,650,1 80000,35000,780,0 30000,5000,600,1 90000,20000,800,0 65000,15000,710,0 40000,3000,620,1

Conclusion and Analysis of Results

In this project, we successfully developed a credit risk prediction model using logistic regression. The model was trained on historical loan and customer data, focusing on key features such as income, loan amount, and credit score. After preprocessing the data and handling any missing values, we achieved a high accuracy in predicting loan defaults.

The evaluation metrics, including precision, recall, and F1-score, demonstrated that our model effectively distinguishes between default and non-default cases. Although the results indicate strong performance, it is essential to recognize potential limitations such as overfitting, especially if the dataset is small or imbalanced.

Future work could involve enhancing the model by incorporating additional features, such as employment history and economic indicators, as well as exploring more complex algorithms like decision trees or ensemble

methods. Continuous monitoring of model performance in real-world scenarios will also be crucial to ensure its reliability over time.

Definitions of Precision, Recall, and F1-Score

1. Precision

Definition: Precision is the ratio of true positive predictions to the total predicted positives. It answers the question: *Of all the instances predicted as positive, how many were actually positive?*

Formula:
Precision = TP / (TP + FP)
where:
TP = True Positives (correctly predicted positive cases)
FP = False Positives (incorrectly predicted as positive)

2. Recall (Sensitivity)

Definition: Recall, also known as sensitivity or true positive rate, is the ratio of true positive predictions to the total actual positives. It answers the question: *Of all the actual positive instances, how many did we correctly predict?*

Formula:
Recall = TP / (TP + FN)
where:
FN = False Negatives (actual positive cases incorrectly predicted as negative)

3. F1-Score

Definition: The F1-score is the harmonic mean of precision and recall. It balances the two metrics, making it a good overall measure of a model's performance, especially when you need a balance between precision and recall. It is particularly useful in situations with class imbalance.

Formula:
F1-Score = 2 * (Precision * Recall) / (Precision + Recall)

Key Points

- **Precision** focuses on the accuracy of positive predictions, important when the cost of false positives is high (e.g., fraud detection).
- **Recall** emphasizes the model's ability to find all relevant cases, crucial when the cost of false negatives is high (e.g., disease detection).
- **F1-Score** provides a single metric that combines both precision and recall, making it easier to compare different models.

Example

If we have a confusion matrix like the following:

	Predicted Positive	**Predicted Negative**
Actual Positive	TP (10)	FN (2)
Actual Negative	FP (3)	TN (15)

You can calculate:

- **Precision:** 10 / (10 + 3) ≈ 0.769 (76.9%)
- **Recall:** 10 / (10 + 2) ≈ 0.833 (83.3%)
- **F1-Score:** 2 * (0.769 * 0.833) / (0.769 + 0.833) ≈ 0.800 (80.0%)

These metrics give us a comprehensive view of our model's performance beyond just accuracy.