# Credit Card Approval Prediction

**Author:** ISTIAK ALAM
**Affiliation:** DataCamp Real World Project

## Executive Summary

This project builds an **automatic credit card approval predictor** using machine learning, replicating real-world bank processes for evaluating applications.
The goal is to streamline and standardize credit risk screening, saving time and resources compared to manual review.

Working with a subset of the **UCI Credit Card Approval Dataset**, the workflow covers:

- **Data loading & exploration**
- **Cleaning & preprocessing** (handling missing values, scaling, encoding)
- **Model training & hyperparameter tuning**
- **Performance evaluation** using confusion matrices

The final tuned **Logistic Regression model** delivers robust prediction capability, positioning this pipeline as a reproducible framework ready for expansion with more data or features.

## Table of Contents

## Introduction

## Background

Commercial banks process thousands of credit card applications daily.
Rejection or acceptance typically depends on factors such as:

- Credit history and outstanding loans
- Annual income
- Residential status
- Employment stability

Manual vetting is slow, subjective, and prone to errors. Automating it via **machine learning** reduces operational overhead and provides consistent, objective assessments.

## Dataset Description

- **Source:** UCI Machine Learning Repository — Credit Card Approval dataset (subset)
- **Shape:** ~690 rows × 14 columns
- **Target Column:** Column index `13` (`+` for approval, `-` for rejection)
- **Feature Types:**
    - Categorical: e.g., gender, resident status, employment type
    - Numerical: e.g., age, years employed, income
    - Mixed data types with missing values

**Sample Data (first 5 rows):**

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| b | 30.83 | 0.000 | u | g | w | v | 1.25 | t | t | 1 | g | 0 | + |
| a | 58.67 | 4.460 | u | g | q | h | 3.04 | t | t | 6 | g | 560 | + |
| a | 24.50 | 0.500 | u | g | q | h | 1.50 | t | f | 0 | g | 824 | + |

## Methodology

1. **Import & Exploration**
    - Libraries: `pandas`, `numpy`, `scikit-learn` (`LogisticRegression`, `GridSearchCV`), `matplotlib`, `seaborn`
    - Loaded dataset into DataFrame `cc_apps`
2. **Data Preprocessing**
    - Detected & handled missing values
    - Encoded categorical variables into numerical formats (label/one-hot encoding)
    - Scaled numerical features using `StandardScaler` for model stability
3. **Data Splitting**
    - 80/20 train-test split using `train_test_split`
4. **Model Training**
    - Baseline Logistic Regression model fit to training set
    - Hyperparameter tuning for regularization (`C` parameter) via `GridSearchCV`
5. **Evaluation**
    - Predicted on test set
    - Assessed performance using a **confusion matrix** and accuracy metrics

## Results

- **Best Hyperparameters:** Logistic Regression with tuned `C` found by GridSearchCV
- **Confusion Matrix:**
    - High true positives (approved correctly)
    - Low false positives (minimizing risky approvals)
- **Test Accuracy:** Good classification performance for a baseline model

**Interpretation:**

- Logistic Regression offers interpretable coefficients, identifying which features influence approval likelihood.
- Accuracy balanced between approval and rejection classes — important for real banking scenarios.

## Discussion

- **Strengths:**
  - Simple, interpretable model suitable for explaining to non-technical stakeholders
  - Effective at capturing most approval/rejection patterns with minimal tuning
  - Modular design allows easy swap to more complex classifiers
- **Limitations:**
  - Dataset size is small — scaling with more applications will improve robustness
  - Additional feature engineering (e.g., credit utilization ratios, debt-to-income) could improve accuracy
  - Class imbalance not deeply addressed — might need SMOTE or weighted loss if imbalance is significant

## Conclusion & Recommendations

**Conclusion:**

This notebook demonstrates a clean, reproducible **machine learning pipeline** for automating credit card approval predictions, effectively reducing manual overhead and improving decision accuracy.

**Recommendations:**

- Expand dataset with recent applications and more features
- Evaluate other classifiers (e.g., Random Forests, XGBoost)
- Implement explainability tools (SHAP, LIME) to support regulatory compliance
- Deploy model via an API for real-time in-bank use

## Appendix

**Core Deliverables:**

- Preprocessed Dataset (ready for model input)
- Tuned Logistic Regression model
- Confusion matrix & accuracy metrics
- Scalable workflow adaptable to other credit scoring tasks