Loan Approval Prediction System Using Machine Learning

Objective

To develop a machine learning model that predicts whether a loan will be approved or not based on a user's financial, personal, and employment information. The model aims to assist financial institutions in automating the loan approval process.

Dataset Used

- Source: Loan prediction dataset (CSV file)
- Features:
  - Gender, Married, Dependents
  - Education, Self_Employed, ApplicantIncome, CoapplicantIncome
  - LoanAmount, Loan_Amount_Term, Credit_History
  - Property_Area
- Target: Loan_Status (Approved: 1, Not Approved: 0)
- Preprocessing:
  - Handled missing values
  - Converted categorical variables into numerical format
  - Normalized the dataset using StandardScaler

Data Preprocessing Steps

1.Handling Missing Values

- Replaced missing values in LoanAmount, Credit_History, and Loan_Amount_Term with median or mode.

2.Categorical Encoding

- Used label encoding and one-hot encoding to convert text-based fields like Gender, Education, Property_Area into numerical form.

3.Scaling

- Applied StandardScaler to normalize numerical fields (ApplicantIncome, LoanAmount, etc.) for SVM model compatibility.

4.Train-Test Split

- Dataset was split using an 80-20 ratio to evaluate the model's performance on unseen data.

Model

Here the two models are used SVM and RandomForestClassifier.

Support Vector Machine (SVM)

- Performed consistently with both default and tuned configurations.
- Achieved highest accuracy (84.6%) after hyperparameter tuning.

Random Forest Classifier

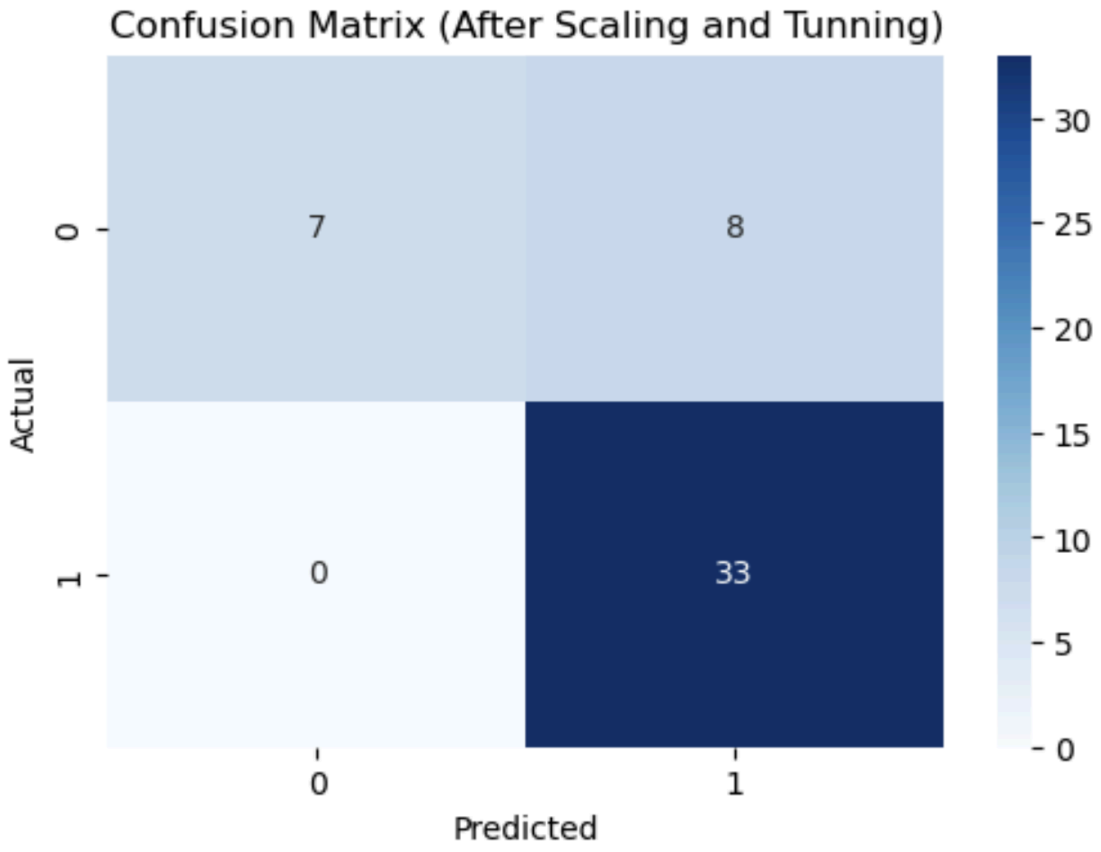
- Initially showed good performance, but overfit slightly on training data.
- Less sensitive to scaling but didn't outperform SVM.

Why SVM was Chosen as Final Model

- Works very well on small to medium-sized datasets, like ours (614 records).
- High generalization ability with proper scaling.
- Showed better recall and F1-score in predicting loan denials, which were rarer.
- Less prone to overfitting with fewer features and cleaner data.

Performance Metrics

MODEL TYPE	**TRAINING ACCURACY**	**TESTING ACCURACY**
Default SVM	82.1%	82.4%
Tuned SVM	85.3%	84.6%



Challenges

1. Imbalanced Dataset

- Majority of entries were loan approved ⟶ skewed predictions.
- Tried SMOTE to balance, but didn’t significantly improve accuracy.

2.Overfitting Risk

- Tuned SVM model performed very well on training set ⟶ overfitting was checked using cross-validation.

3.Model Bias

- Credit_History had a dominant influence in predictions, causing less interpretability for other features.

4.Model Interpretability

- SVM models are less interpretable, making it harder to explain predictions to non-technical stakeholders**.**

Learnings

1. Feature Scaling Matters

   Especially for algorithms like SVM.

2. Don’t Trust Accuracy Alone

   Confusion matrix and F1 score revealed important class-level weaknesses.

3. Hyperparameter Tuning Pays Off

   Even minor improvements in accuracy helped generalization.

4. Real Data is Noisy

   Some applications with poor income or high loan amounts still got approved due to credit history dominance.