Loan Default Prediction Using Machine Learning

Project Overview

This project focuses on predicting loan default using machine learning algorithms. We utilize structured borrower data to train predictive models that assist financial institutions in making informed lending decisions. The implementation follows a comprehensive machine learning pipeline with emphasis on data preprocessing, handling class imbalance, and iterative model improvement.

Objective

The primary objective is to develop a predictive system that accurately identifies whether a loan applicant is likely to default. Given the inherent challenges of imbalanced financial datasets, our solution implements a complete ML pipeline including exploratory data analysis, data cleaning, feature engineering, class balancing, model training, evaluation, and deployment through an interactive Streamlit web application.

Complete Project Pipeline

1. Data Generation and Cleaning

- Generated synthetic loan data simulating real-world scenarios
- Created realistic features including age, income, credit score, employment details
- Ensured data quality by handling potential missing values
- Validated data distributions to maintain realistic patterns

2. Exploratory Data Analysis

- Analyzed feature distributions using histograms, boxplots, and density plots

- Examined correlation between numerical features using heatmaps
- Investigated class imbalance through target variable distribution
- Identified key relationships between borrower characteristics and default risk

3. Feature Engineering

- Created derived features including debt-to-income ratio and loan-to-income ratio
- Implemented proper encoding for categorical variables (One-Hot Encoding)
- Scaled numerical features to ensure model convergence
- Selected relevant features based on domain knowledge and correlation analysis

4. Handling Class Imbalance

- Addressed significant class imbalance using SMOTE technique
- Generated synthetic minority class samples to balance dataset
- Maintained data integrity while creating balanced training sets
- Ensured proper evaluation metrics to account for imbalance

Models and Iterative Improvement Process

Model 1 – Logistic Regression (Baseline)

```
text

LogisticRegression(
   class_weight='balanced',
   max_iter=1000,
   random_state=42
)
```

Training Time: ~15 seconds
Classification Report:

- Precision (Class 1): 0.21
- Recall (Class 1): 0.19

- F1-score (Class 1): 0.20
  Accuracy: 0.83
  ROC-AUC Score: 0.71

Interpretation:

- The model demonstrates basic predictive capability but struggles with minority class identification
- Linear assumptions limit performance on complex patterns
- Provides a solid baseline for comparison with more advanced models

Model 2 – Random Forest Classifier

- Implemented with 100 estimators and balanced class weights
- Captured non-linear relationships effectively
- Handled feature interactions more efficiently
- Demonstrated improved minority class detection

Performance:

- Accuracy: 0.85
- Precision (Class 1): 0.32
- Recall (Class 1): 0.30
- ROC-AUC: 0.82

Model 3 – Gradient Boosting Classifier

- Optimized with appropriate learning rate and depth parameters
- Leveraged sequential learning for enhanced performance
- Demonstrated superior handling of complex decision boundaries
- Achieved best overall performance metrics

Performance:

- Accuracy: 0.86
- Precision (Class 1): 0.35
- Recall (Class 1): 0.33
- ROC-AUC: 0.85

Model 4 – Support Vector Machine

- Implemented with RBF kernel for non-linear classification
- Utilized class weighting for imbalance handling
- Demonstrated strong performance on well-separated cases

Performance:

- Accuracy: 0.84
- Precision (Class 1): 0.28
- Recall (Class 1): 0.26
- ROC-AUC: 0.79

Comprehensive Evaluation Summary

Model	Accuracy	Precision (1)	Recall (1)	F1-Score (1)	ROC-AUC
Logistic Regression	0.83	0.21	0.19	0.20	0.71
Random Forest	0.85	0.32	0.30	0.31	0.82
Gradient Boosting	0.86	0.35	0.33	0.34	0.85
Support Vector Machine	0.84	0.28	0.26	0.27	0.79

Advanced Visualization and Interpretation

Model Performance Analysis

- **Confusion Matrices**: Comparative analysis across all models to visualize true/false positives and negatives
- **ROC Curves**: Detailed evaluation of classification threshold performance
- **Precision-Recall Curves**: Focused analysis on minority class performance
- **Feature Importance**: Comprehensive visualization of key predictive factors

Key Insights

- Credit score emerged as the most significant predictor of loan default
- Debt-to-income ratio and employment status showed strong correlation with default risk
- Loan amount and applicant income demonstrated moderate predictive power
- Categorical features like loan purpose and home ownership provided valuable contextual information

Streamlit Web Application

Application Features

- **Interactive User Interface**: Clean, intuitive design for easy navigation
- **Real-time Prediction**: Instant loan default risk assessment
- **Comprehensive Input Form**: Detailed borrower information collection
- **Visual Results Display**: Clear presentation of prediction probabilities and risk factors
- **Model Comparison**: Side-by-side performance metrics visualization

Application Sections

1. **Data Overview**: Complete dataset exploration and statistics
2. **EDA Analysis**: Interactive visualizations and correlation studies
3. **Model Training**: Flexible model selection and training interface
4. **Predictions**: Real-time default risk assessment
5. **Model Evaluation**: Comprehensive performance metrics and comparisons

Project Repository Structure

```
text

LOAN-DEFAULT-PREDICTION/
├── app/
│   ├── screenshots/
│   │   ├── data_overview.png
│   │   ├── model_training.png
│   │   └── predictions.png
│


├── loan_default_app.py
├── requirements.txt
└── README.md
```

GitHub Repository

Project URL: https://github.com/RameenKhan/loan-default-prediction

Key Findings and Conclusions

Technical Achievements

- Successfully implemented a complete machine learning pipeline for loan default prediction
- Demonstrated the effectiveness of ensemble methods over traditional classifiers
- Achieved significant improvement in minority class detection through proper imbalance handling
- Developed a production-ready web application for practical implementation

Business Impact

- Provides financial institutions with reliable default risk assessment tools
- Enables data-driven lending decisions reducing potential losses
- Offers interpretable results for regulatory compliance and customer communication
- Supports scalable implementation for various lending scenarios

Methodological Insights

- Emphasized the importance of proper evaluation metrics beyond accuracy
- Demonstrated the value of iterative model improvement and comparison
- Highlighted the critical role of feature engineering in financial prediction
- Established best practices for handling class imbalance in credit risk modeling

Future Enhancements

Technical Improvements

- Integration of deep learning models for enhanced pattern recognition
- Implementation of explainable AI techniques for model interpretability

- Real-time model updating with streaming data capabilities
- Advanced feature selection using automated ML techniques