Predicting Credit Default on German Loan Data

Irene Le & Ran Liu

2025-06-03

Table of contents

1 Executive Summary

This project aims to predict credit default based on German loan data from the UCI Machine Learning Repository. The response variable was a binary value (Default = 0, Default = 1). The goal was to identify the most effective classification model for predicting credit default using 20 different credit applicant details as predictors. 9 Models were trained and evaluated (Linear Discriminant Analysis (LDA), Quadratic Discriminant Analysis (QDA), Logisitic Regression, K-Nearest-Neighbors (KNN), Naive Bayes, Support Vector Machines (Linear, Polynomial, and Radial Basis Function (RBF)). Based on a comparative analysis of the models, the Naive Bayes classification model yielded the best overall performance in separating non-default and default credit applicants. It achieved the highest accuracy and lowest error rate, while maintaining strong sensitivity, specificity, and AUC. Given its balance of accuracy and simplicity, Naive Bayes is recommended as the most optimal model for credit risk assessment in this analysis.

2 Introduction and Background

Credit risk assessment is critical for financial institutions to evaluate an applicant's creditworthiness. The process of assessing an applicant's credit risk involves analyzing a series of factors including credit history, age, employment, and other demographic and financial data.

Credit default occurs when a borrower fails to repay a loan, and signals that a borrower is unable to meet financial obligations.

One of the challenges in this analysis was addressing the class imbalance in the Default variable, which presented significantly more non-default cases than default cases. This imbalance introduce a bias in the models, where classifiers can favor the majority class. To address this, we undersampled the majority (non-default) class. After doing so, our models yielded much better performance.

3 Data

Data Description

The data used for this project comes from the German Credit dataset obtained from the UCI Machine Learning Repository. It contains information on 1,000 credit applicants. The dataset includes 20 predictor variables, consisting of categorical (e.g. credit history, personal status, employment status) and numerical (e.g loan duration, credit amount, age) variables. The response variable, Default, is binary:

- 0 = Good credit risk
- 1 = Bad credit risk

The dataset exhibited a class imbalance, with a higher number of good credit cases than bad. An undersampling method was performed to address the imbalance and improve the performance of the models.

Data Overview

Variable Name	Description	Type
Default	Credit default (0 = No, 1 = Yes)	Binary
checkingstatus1	Status of existing checking account	Categorical
duration	Duration in months	Numerical
history	Credit history	Categorical
purpose	Purpose of the loan	Categorical
amount	Credit amount	Numerical
savings	Savings account/bonds	Categorical
employ	Years at current employment	Categorical
installment	Installment rate (% of disposable income)	Numerical
status	Personal status and sex	Categorical
others	Other debtors / guarantors	Categorical
residence	Years at present residence	Numerical
property	Type of property	Categorical
age	Age in years	Numerical
otherplans	Other installment plans	Categorical
housing	Housing situation	Categorical
cards	Number of existing credits at this bank	Numerical
job	Job type	Categorical
liable	Number of people liable for maintenance	Numerical
tele	Telephone availability	Categorical
foreign	Foreign worker status	Categorical

Exploratory Data Analysis

1.1 Relationships of Age, Duration, and Amount by Default

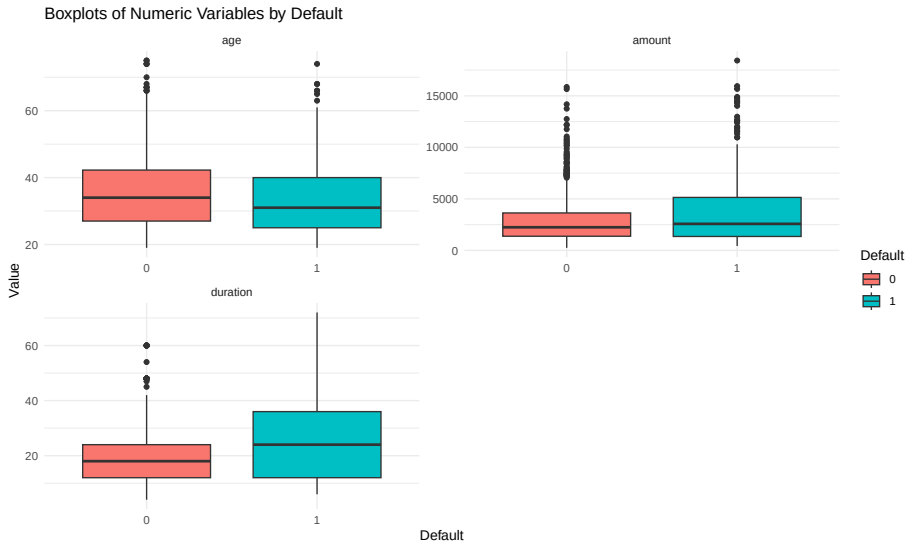

Figure 1: Boxplots of Numeric Variables by Default

Key numerical predictors were examined to identify trends related to default. Several trends were found:

- Credit Amount: Defaulters generally had higher credit amounts.
- Loan Duration: Higher default rates were observed among applicants with longer loan duration.
- Age: Defaulters tended to be slightly younger on average.

1.2 Correlation of All Numerical Variables

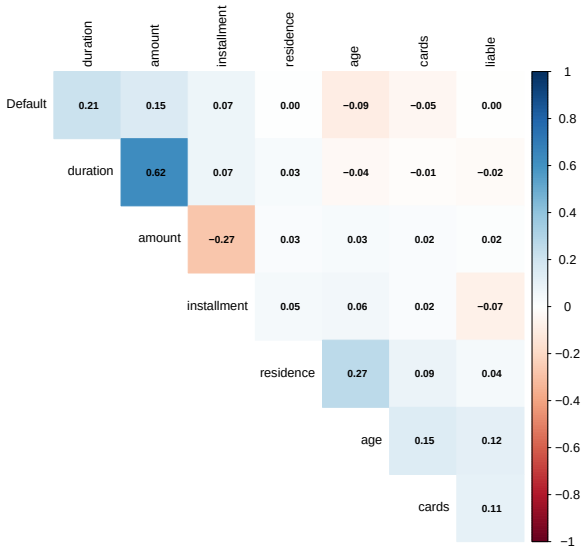

Figure 4: Correlation Matrix of Numerical Variables and Default

The above correlation matrix displays the relationships between numeric variables in our dataset.

- Duration and Amount show a moderate positive correlation (r = 0.62), indicating that longer loan durations may be associated with higher loan amounts.
- Default shows a weak but positive correlation with duration (r = 0.21) and amount (r = 0.15), suggesting that longer loans and higher loan amounts are slightly associated with a higher risk of default.
- Most other variables show weak correlations with Default, such as installment, residence, age, and cards.

1.3 Relationships of Numerical Variables

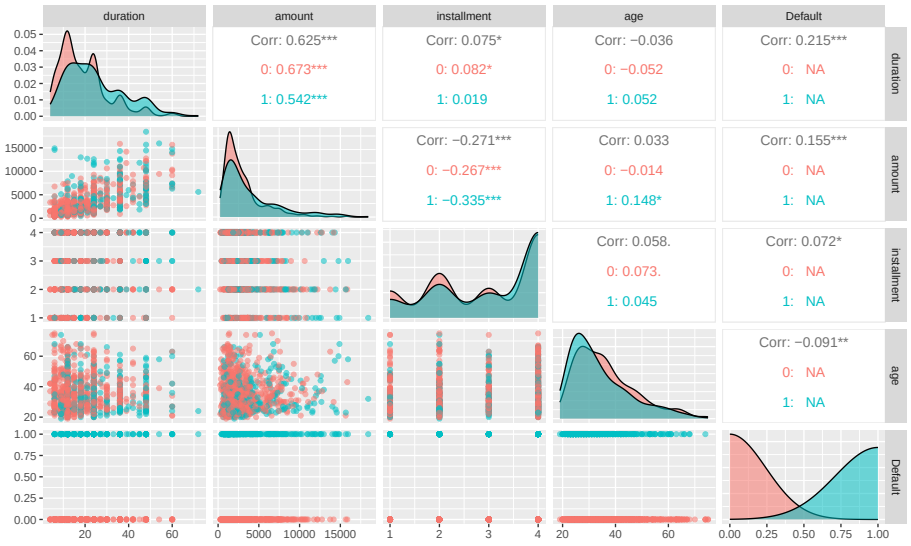

Figure 5: Pairplots, Density Plots, and Correlations of Numerical Variables

The figure above shows a pairwise correlation matrix and density plots for key numeric variables, including duration, amount, installment, age, and Default.

From the density plots, we observe that:

- Defaulters tend to have shorter loan durations, while non-defaulters show a more right-skewed distribution with longer durations.
- Loan amount distributions overlap, but defaulters show a higher concentration of lower loan amounts.
- The distribution of installment rates is uniform across default status.
- Colored by default status, the scatterplots do not reveal strong visual separability. None of the numerical variables individually offer clear linear separation between defaulters and non-defaulters.

1.4 Relationships of Categorical Variables

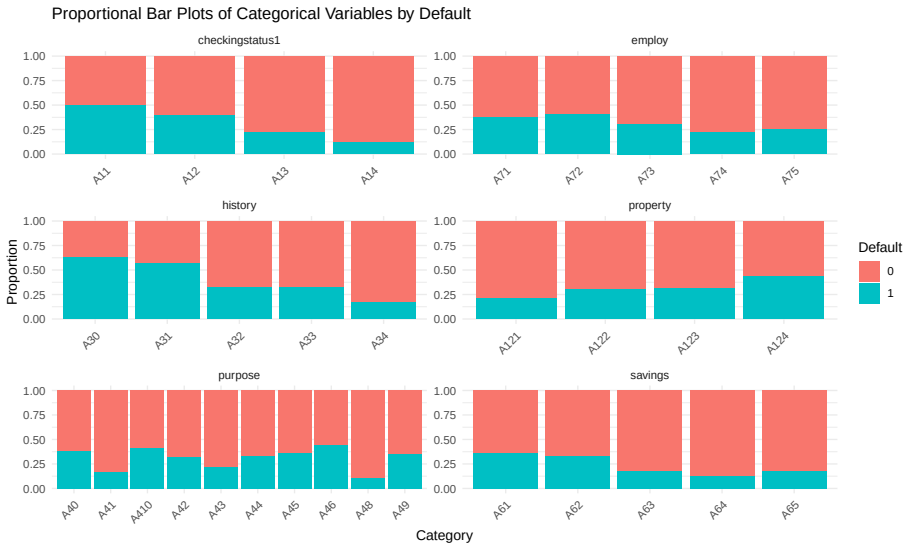

Figure 2: Proportion Bar Plots of Categorical Variables by Default

Proportional barplots were used to visualize the how different factor levels correspond to default outcomes. Several trends were found:

- Checking Account Status: Applicants with negative balances (A11) had the highest default rate.
- Employment Status: The unemployed group (A71) showed a higher proportion of defaults.
- Credit History: Applicants who had no credits taken / all credits paid back (A71) had the highest proportion of defaults.
- Property Ownership: Applicants with no property (A124) were more likely to default.
- Purpose of Loan: Used car purchases (A41) was associated with increased defaults.
- Savings: Applicants with low or no savings (A61) were more likely to default.

1.5 Class Imbalance

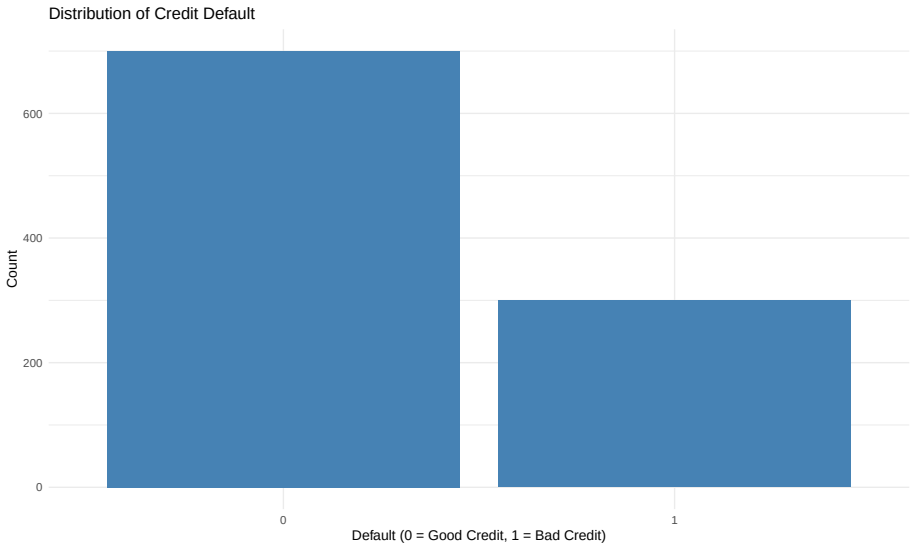

Figure 3: Distribution of Credit Default

A significant class imbalance was present in the data, with the number of non-default cases exceeding that of default cases. This imbalance can bias classification models in predicting the majority class (non-default) if not addressed properly. We applied undersampling to the non-default group, which improved model sensitivity and specificity.

4 Methodology

Data Preprocessing

The dataset was examined for data quality and consistency. The preprocessing steps included:

- **Data Type Conversion**: All character variables were converted into factor variables to ensure proper handling.
- **Missing Value Check**:
  A check for missing values confirmed that the dataset contained no missing entries.
- **Class Imbalance Handling**:
  The dataset originally exhibited class imbalance, with significantly more non-default (class 0) cases than default (class 1). To address this, we applied random undersampling to the majority class (non-default) to match the number of observations in the minority class. This resulted in a balanced dataset, helping prevent bias in model training.
- **Train-Test Split**:

The balanced dataset was randomly split into 80% training and 20% testing data. All models were trained on the training set and evaluated on the testing set.

- **Categorical Encoding for KNN**:
  Since K-Nearest Neighbors (KNN) requires numeric inputs, we applied **one-hot encoding** to convert categorical variables into dummy variables for KNN modeling.
- **Feature Scaling**:
  SVMs and KNN relied on normalized input due to the structure of the model matrix generated from `model.matrix()` during encoding.

Methods

A variety of statistical classification models were explored to determine the most effective model for predicting credit default. The following models were implemented:

- **Logistic Regression:** A linear classification model that estimates the probability of default using a logistic function.
- **Linear Discriminant Analysis (LDA):** A probabilistic classifier that finds a linear combination of features that best separates default vs. non-default cases.
- **Quadratic Discriminant Analysis (QDA):** Similar to LDA but allows each class to have its own covariance matrix, making it more flexible in capturing non-linear boundaries.
- **Naive Bayes:** A model based on Bayes' Theorem, assuming conditional independence between features.
- **K-Nearest Neighbors (KNN):** A model that classifies datapoints based on the majority class among the k closest observations in the training data.
- **Support Vector Machines (SVM):** A model that utilizes the optimal hyperplane separating the classes. We tested SVMs with:
  - **Linear kernel:** Assumes the classes are linearly separable.
  - **Polynomial kernel (degrees 2 and 3):** Captures non-linear relationships using polynomial transformations.
  - **Radial Basis Function (RBF) kernel:** Enables the SVM to separate data with curved or complex boundaries.

All models were based on an 80/20 train-test split.

Model Evaluation

Each model was evaluated on the 20% test set using the following five performance metrics:

- **Accuracy**: The overall proportion of correct predictions.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

- **Sensitivity (Recall or True Positive Rate)**: The ability of the model to correctly identify actual defaulters.

$$\text{Sensitivity} = \frac{TP}{TP + FN}$$

- **Specificity (True Negative Rate)**: The ability to correctly identify non-defaulters.

$$\text{Specificity} = \frac{TN}{TN + FP}$$

- **Error Rate**: The overall proportion of incorrect predictions.

$$\text{Error Rate} = \frac{FP + FN}{TP + TN + FP + FN}$$

- **AUC (Area Under the ROC Curve)**: A measure of the model's ability to distinguish between default and non-default classes across all thresholds.

Table 2: Confusion Matrix Terminology

Term	Description
TP (True Positive)	Model predicted "Default" and they actually did default
TN (True Negative)	Model predicted "No Default" and they did not default
FP (False Positive)	Model predicted "Default" but they did not default
FN (False Negative)	Model predicted "No Default" but they actually did default

Hyper-parameter Tuning

For models requiring tuning, we used 10-fold cross-validation on the training set to select optimal hyperparameters:

- **K-Nearest Neighbors (KNN):** The number of neighbors k was selected by minimizing classification error on the validation folds.
- **Support Vector Machines (SVM):** We tuned the cost parameter, which controls the trade-off between margin width and classification error, and the kernel parameter for the RBF kernel, which determines the influence of individual support vectors.

Grid search was performed over a range of values, and the parameter that achieved the highest average cross-validation accuracy was selected for final evaluation on the test set.

The table below summarizes the optimal hyperparameter values selected for each model.

Table 3: Hyperparameter Tuning Summary

Model	**Hyperparameter**	**Optimal Value**
KNN	K (number of neighbors)	K = 6
SVM (Linear)	Cost	Best cost = 0.01
SVM (Polynomial Degree 2)	Cost	Best cost = 10
SVM (Polynomial Degree 3)	Cost	Best cost = 100
SVM (RBF)	Cost, Gamma	Best cost = 10 Best gamma = 0.01

5 Results

This study aimed to classify defaulters and non-defaulters based on an applicants' financial and personal attributes. The results below summarize the predictive performance of each model, highlighting trade-offs between correctly identifying defaulters and minimizing misclassifications.

Model Performance Summary

Table 4: Model Performance Summary

Model	Accuracy	Error.Rate	Specificity	Sensitivity	AUC
LDA	0.8083	0.1917	0.7778	0.8421	0.8630
QDA	0.7667	0.2333	0.7460	0.7895	0.8131