Loan Default Prediction Project

Aryaman Gulati (Github: agulati18)

Tl;dr

The goal of this project was to develop a predictive model to determine whether a loan applicant would default or not. The project began with exploratory data analysis (EDA) to gain insights into the dataset and identify potential relationships between the variables.

Initially, logistic regression was used to build the predictive model. However, the model had limitations in accurately predicting the default status of loan applicants.

To improve the model, a random forest algorithm was used, which resulted in a more accurate model with an accuracy of around 94%. Feature engineering, hyperparameter tuning, and cross-validation were used to optimize the random forest model.

Finally, the model was deployed on new, unseen data to test its performance.

In reflection, there were some areas for improvement in the project, such as exploring additional feature engineering techniques and experimenting with other machine learning algorithms. Additionally, collecting more data and performing more thorough EDA could potentially improve the model's accuracy.

Overall, this project provided valuable insights into the loan default prediction problem and demonstrated the importance of feature engineering, hyperparameter tuning, and cross-validation in building accurate predictive models.

Introduction

In this project, I built a machine learning model to predict whether a loan applicant will default on their loan or not. The dataset I used for this project was sourced from Kaggle and contained information on loan applicants' demographics, financial history, and loan characteristics. My goal was to build a predictive model that could accurately predict whether a loan applicant would default on their loan based on these features.

Methodology

Data Preprocessing

I started by performing data preprocessing to clean and prepare the dataset for analysis. This included checking for missing values, handling outliers, one-hot encoding categorical variables, and scaling numerical features.

Exploratory Data Analysis

I also performed some exploratory data analysis (EDA) to gain insights into the relationships between the features and the target variable. I started by looking at the descriptive statistics of the data, such as the mean, median, standard deviation, minimum and maximum values, to get a general idea of the range and distribution of the variables.

Next, I created several visualizations to gain further insights into the relationships between the features and the target variable. For example, for the non-binary features, I created histograms and boxplots to examine the distributions of the continuous variables and to identify any potential outliers. I also created scatterplots to explore the relationships between pairs of continuous variables, and a correlation matrix to identify any highly correlated variables.

For the binary variables, I created bar charts to examine the frequency distributions of the categories and their relationship with the target variable. I also calculated the proportion of observations with positive target values for each category to assess their influence on the target variable.

Through EDA, I gained important insights into the data and the relationships between the variables, which helped me make informed decisions about feature selection, engineering, and model building.

Model Selection and Baseline Model
I started by building a baseline model using logistic regression. I used the processed dataset from the previous step and split it into training and testing sets with a ratio of 80:20. I then fit the logistic regression model on the training set and evaluated its performance on the test set using accuracy, precision, recall, and F1-score metrics. The baseline logistic regression model achieved an accuracy of 0.865 and F1-score of 0.645.

Feature Engineering
To improve the performance of the model, I performed feature engineering by creating new features from the existing ones. I also removed features that were not relevant to the target variable. I used domain knowledge and correlation analysis to create new features that could capture the relationships between the existing features and the target variable.

Model Improvement
I then tried different machine learning algorithms to improve the performance of the model. I started with logistic regression and then tried decision trees, random forests, and gradient boosting algorithms. I used hyperparameter tuning and cross-validation techniques to optimize the parameters of the models and evaluated their performance using the same metrics as before.

I found that the random forest algorithm performed the best, achieving an accuracy of 0.933 and F1-score of 0.829. The best hyperparameters for the random forest algorithm were {'max_depth': None, 'min_samples_split': 2, 'n_estimators': 100}.

Model Deployment
Once I had the final model, I deployed it to predict the loan status of new loan applicants. I used the same preprocessing steps as before to clean and preprocess the new data. I then used the final random forest model to predict the loan status of the new loan applicants.

Tools and Skills Used

- Python programming language
- Pandas library for data preprocessing and manipulation
- Scikit-learn library for machine learning algorithms and evaluation metrics
- Seaborn library for data visualisation
- Jupyter Notebook for coding and documentation
- Exploratory data analysis
- Feature engineering
- Model selection and optimization
- Linear algebra

- Model deployment

Conclusion

In conclusion, I successfully built a machine learning model to predict whether a loan applicant would default on their loan or not. I started with a baseline logistic regression model and then performed feature engineering and tried different machine learning algorithms to improve its performance. I finally selected the random forest algorithm, which achieved an accuracy of 0.933 and F1-score of 0.829. The project helped me develop skills in data preprocessing, exploratory data analysis, feature engineering, model selection, hyperparameter tuning, and model deployment, which are valuable skills for a data scientist.

Next Steps and Areas of Improvement

Next Steps

In the future, there are several possible next steps to consider. One area for improvement is the data itself; I could consider adding more features to improve the model's performance or collecting more data to increase the dataset's size. I could also explore other machine learning algorithms such as gradient boosting or deep learning to see if they can further improve model performance. Additionally, I could use techniques such as ensemble modeling to combine multiple models and improve the predictive power of my overall model.

Areas of Improvement:

One area where I could improve is in the feature engineering step. Although I explored various techniques such as one-hot encoding and scaling, there may be additional methods I could have used to further improve my model's performance. Additionally, I could have done a better job of handling missing data; I simply dropped rows with missing values, but there may have been more sophisticated methods I could have used to impute missing data.

Data Source: https://www.kaggle.com/datasets/laotse/credit-risk-dataset