Credit Card Approval Prediction

Renjini Balachandran

Advisor Name:
Dr. Edward Ofori

Grand Canyon University

Introduction

In the world of lending, efficient and accurate credit card approval decisions are vital. **Traditional manual assessment processes are time-consuming, prone to errors, and lack consistency**. This often results in delays and inconsistencies in application decisions. Automating this process through predictive modeling can improve efficiency, reduce human bias, and enhance customer experience.

Here I am proposing, developing a **machine-learning model that predicts credit card approval** outcomes based on historical application data. By training on a diverse dataset of past approvals and denials, the model will learn patterns and relationships in the data, enabling it to make predictions on new applications. This will streamline the approval process, **providing faster responses to applicants and improving the overall quality of decisions.**

The Resources I am using include High-quality historical **credit card application data**, Access to the necessary **programming language, Python**, Computing resources for model training and evaluation, and Collaboration tools for efficient team communication and progress tracking.

Data Sources Detail

When designing **a model pipeline for predicting credit card approval**, it is essential to consider **relevant data sources** that capture information about credit applicants and their associated outcomes.

Some of the data sources to consider include,
- Historical credit card application data,
- Credit Bureau Data,
- income and employment verification data,
- fraud, and risk management systems.

The data set has been taken from the **kaggle.com data repository (Song, 2019).** This data set is the publicly available data set. Hence information security is not a concern here.

There are two data sets tables that could be merged by ID.

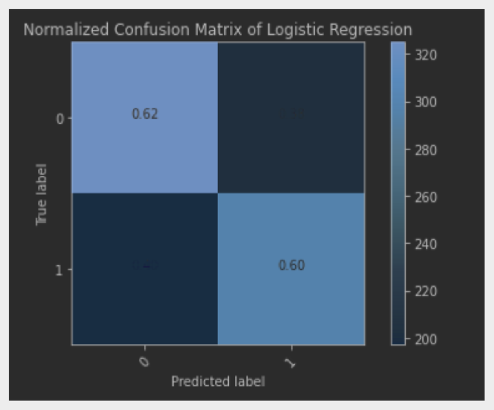
. • application_record.csv for applicant information

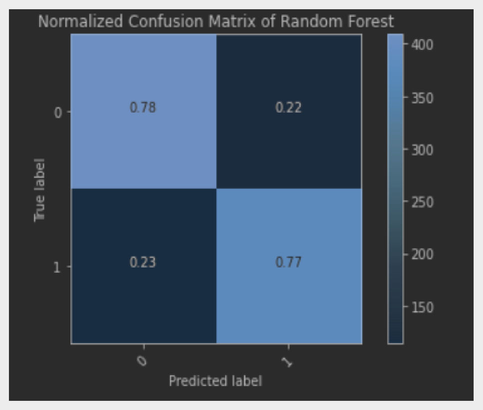
• credit_record.csv for credit record information

Methodology

After acquiring the relevant data set and **data preparation with feature selection** was done and then finalized the data set. Then divide the dataset as a **training and test into a ratio of 80:20**. The training data set is used to train the model by applying
- **Logistic Regression and**
- **Random Forest**.

Python programming and its libraries have been used to develop the models. Finally, **evaluate** the predicted results of these two models and compare the accuracy of the two models by using **Confusion Matrix** to choose the most accurate model. The **test data set is used to test the model and evaluate the outcome.**

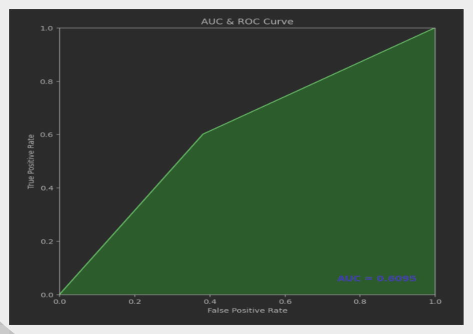




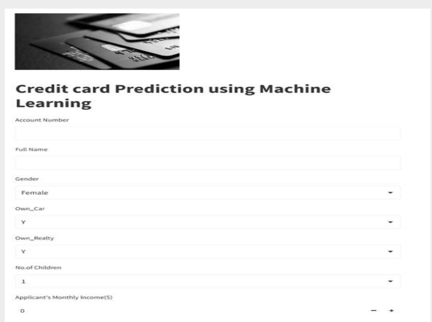
Model Evaluation & Deployment

Employ metrics such as **accuracy, precision, recall, F1-score, and ROC-AUC** to measure the models' effectiveness. Use techniques like **feature importance plots** to understand the contribution of individual features to predictions. **Grid Search/Random Search** is used to fine-tune hyperparameters to optimize model performance and enhance prediction accuracy.

After model evaluation, the **Random Forest performs better than other models**. Integrate the finalized model into a production environment. **Streamlit** is used for creating the data product. Using **Streamlit Cloud to deploy the product and GitHub as the code repository for the code versioning.**



Application Form in Streamlit



Conclusions

Collected data set and **explanatory analysis** were carried out to understand the data set. Then several actions related to data preparations such as **data preprocessing, feature selections, and feature scaling.** Divided the data set into two parts as a **training and test data set** and the aimed purpose is to validate the accuracy of the model.

Two predictive models were implemented,
- Logistic Regression and Random Forest

Performance measures were tested by using,
- Accuracy, Precision, Recall, and AUC

For the **Logistic Regression model**, Accuracy is 0.61, Precision is 0.61, Recall is 0.61 and AUC is 0.609.
For the **Random Forest model**, Accuracy is 0.89, Precision is 0.89, Recall is 0.89 and AUC is 0.889.

Accuracy, Precision, and Recall values are higher in Random Forest than in Logistic Regression and so the Random Forest model is used for deployment.

Resources

1. Fernando, J., 2021. Delinquent Account Credit Card [WWW Document]. Loan Basis. URL https://www.investopedia.com/terms/d/delinquent-account-credit-card.asp (accessed 4.28.21)
2. Hamid, A. J., & Ahmed, T. M. (2016). Developing prediction model of loan risk in banks using data mining. *Machine Learning and Applications: An International Journal*, *3*(1), 1-9. https://doi.org/10.5121/mlaij.2016.3101
3. Leo, M., Sharma, S., Maddulety, K., 2019. Machine Learning in Banking Risk Management: A Literature Review. Risks 7, 29. https://doi.org/10.3390/risks7010029
4. Nayak, L., Sangal, A., (2022, January 11.) Predicting Credit Card Approvals Using Machine Learning Predicting Credit Card Approvals using Machine Learning | by Aman Sangal | Medium
5. Peela,H. V., Gupta, T., Rathod, N., Bose, T., Sharma, N., (2022,January) International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307 (Online), Volume-11 Issue-2 Prediction of Credit Card Approval (ijsce.org)
6. Song, X., 2019. A Credit Card Dataset for Machine Learning. Credit Card Approval Prediction. URL https://www.kaggle.com/rikdifos/credit-card-approval-prediction (accessed 9.10.20).

I would like to give my gratitude to my family, friends, and professors for their continuous support and understanding throughout the project. Above all, I am grateful to Lord the Almighty for all the blessings you have bestowed on my life.