EECS 4404/5327 Project Part 5

Aum Patel, Hrushi Patel, Huy Vuong

(218153338), (218534206), (217746801)

December 5, 2023

Abstract

Our application is Loan Eligibility Prediction which determines if a borrower would be eligible for a loan amount based on their selected features. We have opted to use Logistic Regression as our base line model which takes in 614 samples with 13 features from the dataset and when trained the model had around 80% accuracy. After analysing different models, it was noticed that Logistic Regression performed better in terms of accuracy than decision tree (random forest algorithm) and SVM, as for time, all three algorithms had almost similar performance as the dataset was not significantly large. This analysis helped us determine the number of features needed to increase the accuracy and select appropriate model out of the three.

1 Introduction

1.1 What This Application Is

This is a Loan Eligibility Prediction Application which allows lenders to quickly assess the required criteria and evaluate if a borrower would be eligible for a loan amount, and as for a borrower they can have a preapproval before formally applying for a loan in a much faster way then before.

1.2 What Are the Assumptions & Scope of This Project

This application only focuses on the eligibility part of the loan application. Also, the application assumes that the data obtained from Kaggle or another source, is accurate and representative of real-world loan applications. The selected features, such as income, credit history, and property area, are relevant and significant in predicting loan approval. The scope includes the exploration and comparison of machine learning models, with a focus on Logistic Regression, Random Forest, and SVM. The project implies the use of the model as an aid in decision-making for loan approval but does not replace human judgment or regulatory compliance.

1.3 Justification For Why This Application is Important

The traditional process of applying for loan is time-consuming and is inconvenient for borrowers physically visit banks or lenders, especially for those with busy schedules or limited mobility. Also, opportunities that are limited due to time constraints are lost due to the delays injected by manual assessment. It would be very convenient for a borrower that are thinking of applying for a loan, or to get a rough estimate beforehand if they would be eligible for the loan. While for lenders, they will have less pressure on their work and speed up the process time, which will result in people getting their loan approval earlier.

1.4 Similar Applications

Mortgage Loan Calculators: They predict how much money a borrower will get based on the price of the properties, their incomes, and other documents. Student Loan Eligibility Platforms: Students will know if they are eligible for student loans or not. Credit Scoring System: They are mostly centered based on the credit score and relative financial status to determine the eligibility of the loan which are later sent to the lender and not to the borrower. Other application models used clustering as for techniques while our application takes advantage of the Classification model, this is because our application focuses on whether a burrower would be eligible for a certain loan amount which is a classification problem. While existing applications uses decision tree but according to our analyses our dataset was more efficient with our technique.

1.5 Adjustments to Part One Proposal

In part 1 we were also aiming to provide an output of how much loan amount a burrower is eligible to burrow, this is now being held due to dataset constraints as the data does not contain any information about how much each of the burrower was approved for, and also it would require to change the model based on the eligibility which would make the model a bit more complex and might result in reduction of accuracy.

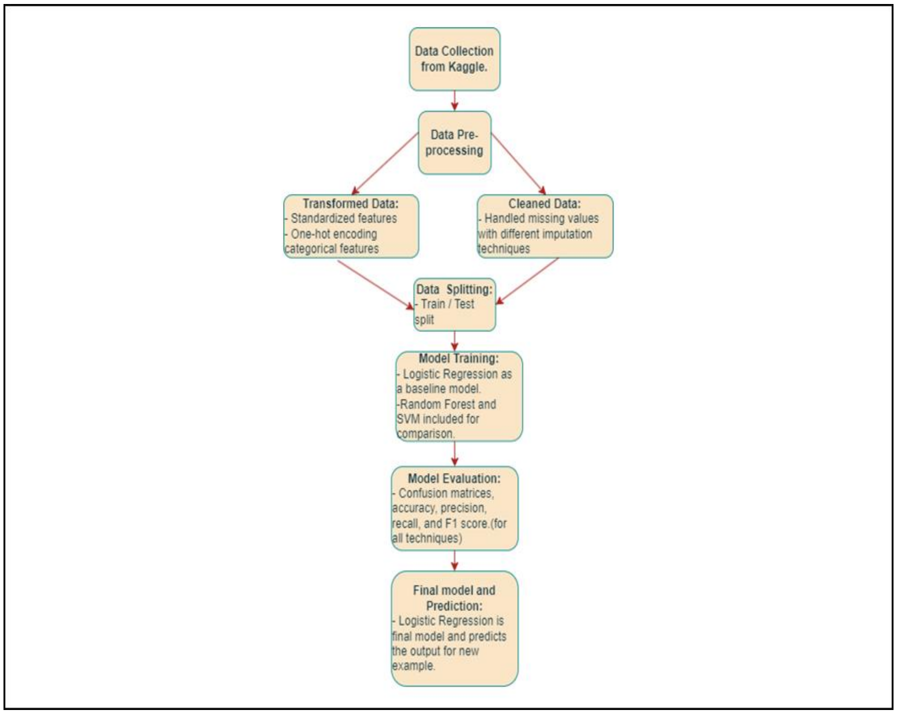
2 Methodology

2.1 Design & Pipeline

The pipeline diagram below shows a comprehensive machine-learning architecture for a Linear Regression task using a Loan Dataset from Kaggle. Initially after data collection, the pipeline proceeds with data preprocessing, where missing values are added to the dataset, and new features are created combining old features with making the dataset more symmetrical to improve the model. Further, the features are standardized, and categorical features are encoded using one-hot encoding to prepare the data for the model.

The data is then split into two sets: a training set and a test set, which is a standard practice to evaluate the model's performance on unseen data. Once these steps are completed successfully multiple models are trained for comparison and Logistic Regression is used as a baseline model. While Random Forest and Support Vector Machine (SVM) are also trained for a more robust comparison.

After that, the models are then trained for comparison based on the following criteria: Confusion matrices are constructed to understand the true positive, false positive, true negative, and false negative rates. Based on the results Accuracy, precision, recall, and F1 score are calculated for all techniques to measure the models' performances. The Logistic Regression model is finalized and used to predict the output for new examples. This suggests that, among the models tested, Logistic Regression was chosen as the most suitable for deployment based on its performance metrics.


Data Collection from Kaggle.
Data Pre-processing
Transformed Data:
- Standardized features
- One-hot encoding categorical features
Cleaned Data:
- Handled missing values with different imputation techniques
Data Splitting:
- Train / Test split
Model Training:
- Logistic Regression as a baseline model.
-Random Forest and SVM included for comparison.
Model Evaluation:
- Confusion matrices, accuracy, precision, recall, and F1 score.(for all techniques)
Final model and Prediction:
- Logistic Regression is final model and predicts the output for new example.

2.2 Dataset

The dataset used for training the models was obtained from Kaggle which is a platform for data science competition and datasets. This data is collected from Dream Housing Company by Kaggle. The data set was not clean and had a lot of missing values in it. For data pre-processing we handled missing values using mean imputation for numerical features like Loan Amount and most frequent imputation for categorical features like Gender, Married, Self-employed. And standardized numerical features to ensure they are on similar scales, as well as changed datatypes of few features to numbers from string to be used in training the model. Lastly, did Label Encoding for categorical features and then applied one-hot encoding for them, which converts categorical variables into binary columns, preserving their information without introducing ordinal relationships, thus providing model compatibility. Performed some steps of feature engineering, where we created new features like EMI by calculating ratio of Loan Amount and Loan Amount Term, combined both type of income to form Total Income and created Balance Income feature by getting difference between Total Income and EMI * 1000, to provide insights into the financial situation of the individual. To address skewness in the numerical features like Loan Amount, Applicant Income, Co-applicant Income, Log transformation was applied to improve model performance. Adjusted the Dependents column values to convert '3+' to the numeric value 3 and also changed other values to number. At last, we dropped a few features for making dataset simple and improve model performance. By using such imputation techniques, we ensured that the models are trained on complete and representative data.

2.3 Model Training

Inputs are preprocessed features, including numerical features (e.g., standardized income, loan amounts) and one-hot encoded categorical features (Gender, Marriage status, Education). However, all the models were trained using almost similar inputs.

Logistic Regression: It is a linear model suitable for binary classification tasks, predicting whether a loan application is approved or rejected. The output is binary prediction indicating loan approval (1) or rejection (0). This model had accuracy of 81.46%. Training process: The model is trained using the logistic regression algorithm. During training, the model optimizes its coefficients to find the best linear decision boundary that separates the classes based on the input features.

Random Forest: Random Forest Classifier, an ensemble learning method that builds multiple decision trees during training. It provides binary prediction as output i.e.: if loan is approved or not. This model has accuracy of 75.76%. Training process: The model is trained by constructing multiple decision trees and combining their predictions. Each tree is trained on a subset of the data and features, contributing to an ensemble model.

SVM: It is a supervised learning algorithm used for classification tasks. The output is like other classification used for the project that being yes or no for loan approval. This model has lowest accuracy of 69.65%, compared to other techniques used so far for this project. Training process: The model is trained to find the hyperplane that best separates the classes in the feature space. Support vectors, which are data points close to the decision boundary, influence the positioning of the hyperplane.

Initially the plan was to train model on same set of features, with aim of comparing their performances based on metrics like accuracy, precision, recall, and F1 score. Multiple models were considered initially to explore their strengths and weaknesses. Random Forest and SVM were included for comparison to identify the model that best suits the characteristics of the dataset and the problem at hand. However, the Logistic regression model was performing better than other ones, thus making it a base line model for our project.

In addition, studies from Project Part 2 made us change how we get the data ready and how we set up our models and influenced the consideration of specific features during preprocessing and the exploration of hyperparameter settings. We learned about the specific details of the data and how different features (like income or education) might be important. This new understanding influenced how we prepared the data before using it to train our machine learning models.

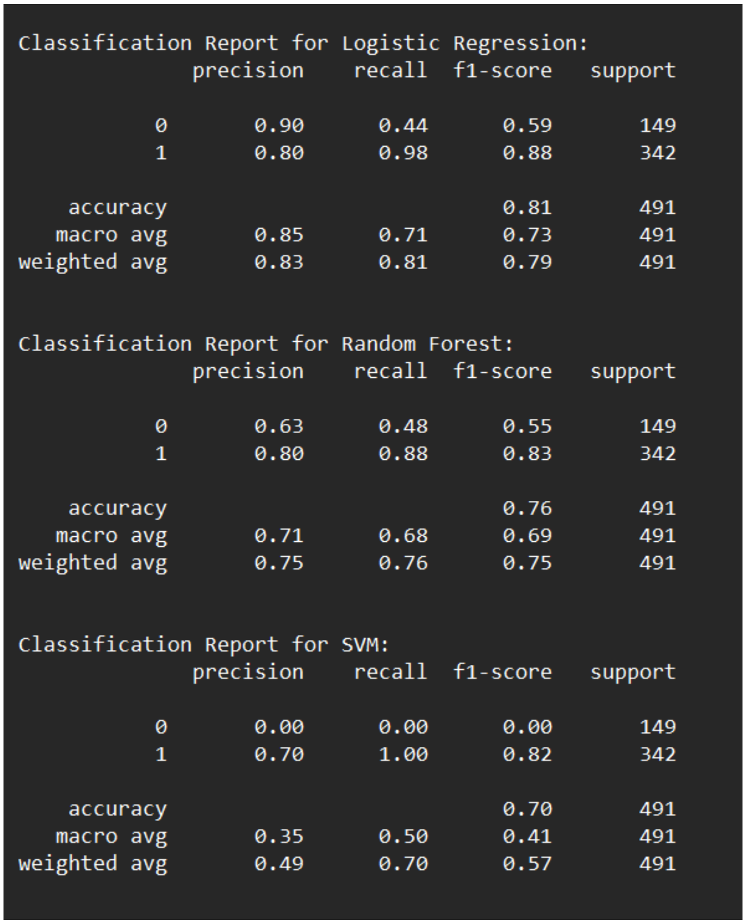
2.4 Prediction

After training the model, we can use it to predict the output for new input example, for that we need to take features of the given example and pre-process them in similar way we did to train the model, that is same steps would be applied for handling missing values, feature engineering and encoding categorical values. After processing the features, the model uses them to make prediction. Thus, giving an output of whether the loan application is likely to be approved or rejected.

3 Results

3.1 Evaluation

The logistic regression model seems to have a decent overall accuracy. The recall is relatively high at 89.1%, suggesting that the model is effective at capturing the positive instances. The precision indicates that when it predicts a positive outcome, it is correct about 81.46% of the time. The accuracy obtained from Random Forest is somewhat moderate. It has decent precision and recall values. Thus, indicating that the performance of the model is reasonably well on test data. The strong recall (sensitivity) of the SVM model suggests that it can recognize every positive case. But at 69.6%, accuracy is not very high, and precision is not relevant (division by zero in the denominator). This suggests that the SVM model is making predictions for the positive class without providing a balance in terms of precision. Overall, after comparing the models we have chosen Logistic Regression as our main model. It predicts more accurately compared to other models.



```
Classification Report for Logistic Regression:
              precision    recall  f1-score   support

           0       0.90      0.44      0.59       149
           1       0.80      0.98      0.88       342

    accuracy                           0.81       491
   macro avg       0.85      0.71      0.73       491
weighted avg       0.83      0.81      0.79       491


Classification Report for Random Forest:
              precision    recall  f1-score   support

           0       0.63      0.48      0.55       149
           1       0.80      0.88      0.83       342

    accuracy                           0.76       491
   macro avg       0.71      0.68      0.69       491
weighted avg       0.75      0.76      0.75       491


Classification Report for SVM:
              precision    recall  f1-score   support

           0       0.00      0.00      0.00       149
           1       0.70      1.00      0.82       342

    accuracy                           0.70       491
   macro avg       0.35      0.50      0.41       491
weighted avg       0.49      0.70      0.57       491
```

3.2 Result

Logistic Regression:

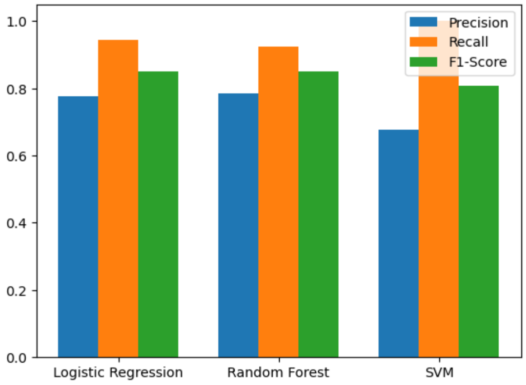
Testing set: achieved an accuracy of approximately 81.46%

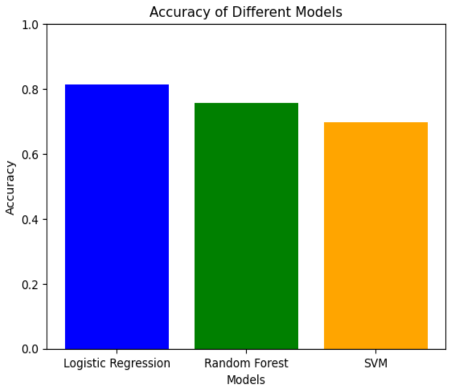
Random Forest:

Testing set: achieved an accuracy of approximately 75.76%

SVM:

Testing set: achieved an accuracy of approximately 69.65%





4 Discussion

4.1 Implications

The outcomes of our loan prediction application indicate that we have substantially accomplished our main objective. Based on the input features, the model shows impressive accuracy in predicting whether a loan applicant will be granted or rejected. This is in line with our main goal of giving financial institutions a trustworthy tool to quickly determine a loan applicant's eligibility.

When outcomes fail to meet up to expectations, a number of things might lead to less-than-ideal performance. These could be the existence of outliers, poor feature relevance, or poor data quality. Furthermore, if the training data does not sufficiently reflect the variety of loan applications that are made in the real world, the model's performance can be compromised.

4.2 Strengths

For our model, it has many different data, and it is not certainly linearly separable, so we use Logistic Regression for better results. Logistic Regression is commonly used for prediction and classification problems, which requires output to be any value between two definite output values such as (yes/no) or (true/false). Logistic Regression, by design, is influenced by all data points in the training set and while our dataset has multiple features and medium-large number of samples which cannot be easily separable therefore, we would want our model to be not heavily influenced by a few outliers.

4.3 Limitations

Logistic Regression can lose effectiveness when dealing with a lot of features (high dimensionality). It might result in overfit models or models that are difficult to train computationally also if the dataset is imbalanced such as data with a significantly higher number of approved loans compared to rejected ones would provide an inaccurate output.

4.4 Future Directions

Getting a more comprehensive dataset is a priority. A larger and more diverse dataset would improve the application. We will continue to iteratively train the model on various datasets, including ones with more features, to enhance it even further. The objective is to rectify any current constraints, enhance the model's ability to anticipate outcomes, and guarantee that it can continue to adjust to changing financial environments. Subsequent versions might investigate enhancing the application's capabilities. This can entail adding dynamic features that record changes in the economy over time, investigating different modeling strategies, or integrating advanced analytics.

The utilization of a Neural Network for the purpose of computing Loan Eligibility is a viable option that has the potential to yield greater accuracy, provided that it is executed appropriately and given access to a larger dataset for training. However, due to the limited availability of data in the dataset and the highly sensitive nature of the information contained therein, obtaining the necessary data has proven to be a significant challenge. Nonetheless, efforts are being made to address this issue in the future.

5 Additional Questions

5.1 Useful feedback

Feedback highlights the critical need for a larger data set in refining our Loan Eligibility Prediction model. Larger data sets strengthen statistical reliability by ensuring the model learns from a more diverse set of examples, minimizing bias from smaller data sets. Improved generalization, critical for accurate prediction of new loan applications, can be achieved with expanded data sets. Furthermore, the call for clear visualization is equally important. Visual results, such as the confusion matrix or ROC curve, serve as powerful tools for conveying complex model performance metrics. They enhance interpretation, facilitate stakeholder engagement and shared understanding. Combining a larger dataset with insightful visualizations aligns with our commitment to building a robust, generalizable, and transparent Loan Eligibility Prediction model.

5.2 Changes

We have added graphical representation of the results, as recommended by the peer reviews. Additionally, we have engineered and modified the features to achieve higher accuracy and improved the format layout for better representation. Also, we have elaborated on the features and their usage to provide a more comprehensive understanding of the methodology employed.

References

[1]. Loans dataset: Loan Eligible Dataset (kaggle.com)

[2]. Similar application 1: Mortgage Loans Calculators Mortgage Tools and Calculators | CIBC

[3]. Similar application 2: Student Loans Eligibility Platforms StudentAid BC

[4]. Similar application 3: Credit Scoring System How to Check Your Credit Score - RBC Royal Bank

[5]. Regression: What is Regression? Definition, Calculation, and Example (investopedia.com)

[6]. Suggestion: Prediction of Loan Approval in Banks Using Machine Learning Approach by Viswanatha V, Ramachandra A.C, Vishwas K N, Adithya G :: SSRN

[7]. https://neptune.ai/blog/ml-pipeline-architecture-design-patterns