Loan Prediction Project

Ayelene Biju

BBA Big Data & Management

Table of contents

Introduction

CasaBank has been experiencing significant challenges with its manual loan request procedure. Each application requires detailed analysis, contributing to a slow and inefficient process. Recently, the bank has seen a rise in loan applications, increasing the workload for employees and extending response times, which has led to heightened customer dissatisfaction. To address these issues, the head office has requested the development of a model to expediently analyze cases and decide on loan approvals.

This report will discuss the application of various data mining techniques to understand the key factors influencing loan decisions. It will also cover the development of a deep learning model designed to predict whether loans should be approved or rejected. The insights gained from this work and the performance of the model will be presented, highlighting how they can enhance the bank's operations.

Methods – Process

1.1 Dataset analysis

Firstly, the loan history dataset was analyzed among its different categories. Here is a preview of it.

	ID	Age	Experience	Income	ZIP Code	Family	CCAvg	Education	Mortgage	Personal Loan	Securities Account	CD Account	Online	CreditCard
0	1	25	1	49	91107	4	1.6	1	0	0	1	0	0	0
1	2	45	19	34	90089	3	1.5	1	0	0	1	0	0	0
2	3	39	15	11	94720	1	1.0	1	0	0	0	0	0	0
3	4	35	9	100	94112	1	2.7	2	0	0	0	0	0	0
4	5	35	8	45	91330	4	1.0	2	0	0	0	0	0	1
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
4995	4996	29	3	40	92697	1	1.9	3	0	0	0	0	1	0
4996	4997	30	4	15	92037	4	0.4	1	85	0	0	0	1	0
4997	4998	63	39	24	93023	2	0.3	3	0	0	0	0	0	0
4998	4999	65	40	49	90034	3	0.5	2	0	0	0	0	1	0
4999	5000	28	4	83	92612	3	0.8	1	0	0	0	0	1	1

5000 rows × 14 columns

Figure 1: Loan history dataset

The variables are explained as follows:

- ID: A unique identifier for each customer.
- Age: The age of the customer in years.
- Experience: The number of years the customer has professional experience.
- Income: The annual income of the customer (in thousand dollars).
- ZIP Code: The home address ZIP code of the customer.
- Family: The size of the customer's family.
- CCAvg: The average monthly credit card spending (in thousands of dollars).
- Education: An ordinal variable indicating the level of education. This might include values like 1 for undergraduate, 2 for graduate, and 3 for advanced/professional education.
- Mortgage: The value of the house mortgage if any (in thousands of dollars).
- Personal Loan: A binary variable indicating whether the customer was granted a loan by the bank (1 for yes, 0 for no).
- Securities Account: A binary variable indicating whether the customer has a securities account with the bank (1 for yes, 0 for no).
- CD Account: A binary variable indicating whether the customer has a certificate of deposit (CD) account with the bank (1 for yes, 0 for no).
- Online: A binary variable indicating whether the customer uses Internet banking facilities (1 for yes, 0 for no).
- CreditCard: A binary variable indicating whether the customer uses a credit card issued by the bank (1 for yes, 0 for no).

To better understand how the different categories impact the loan decision, a correlation matrix was used to analyze the relationships between variables.

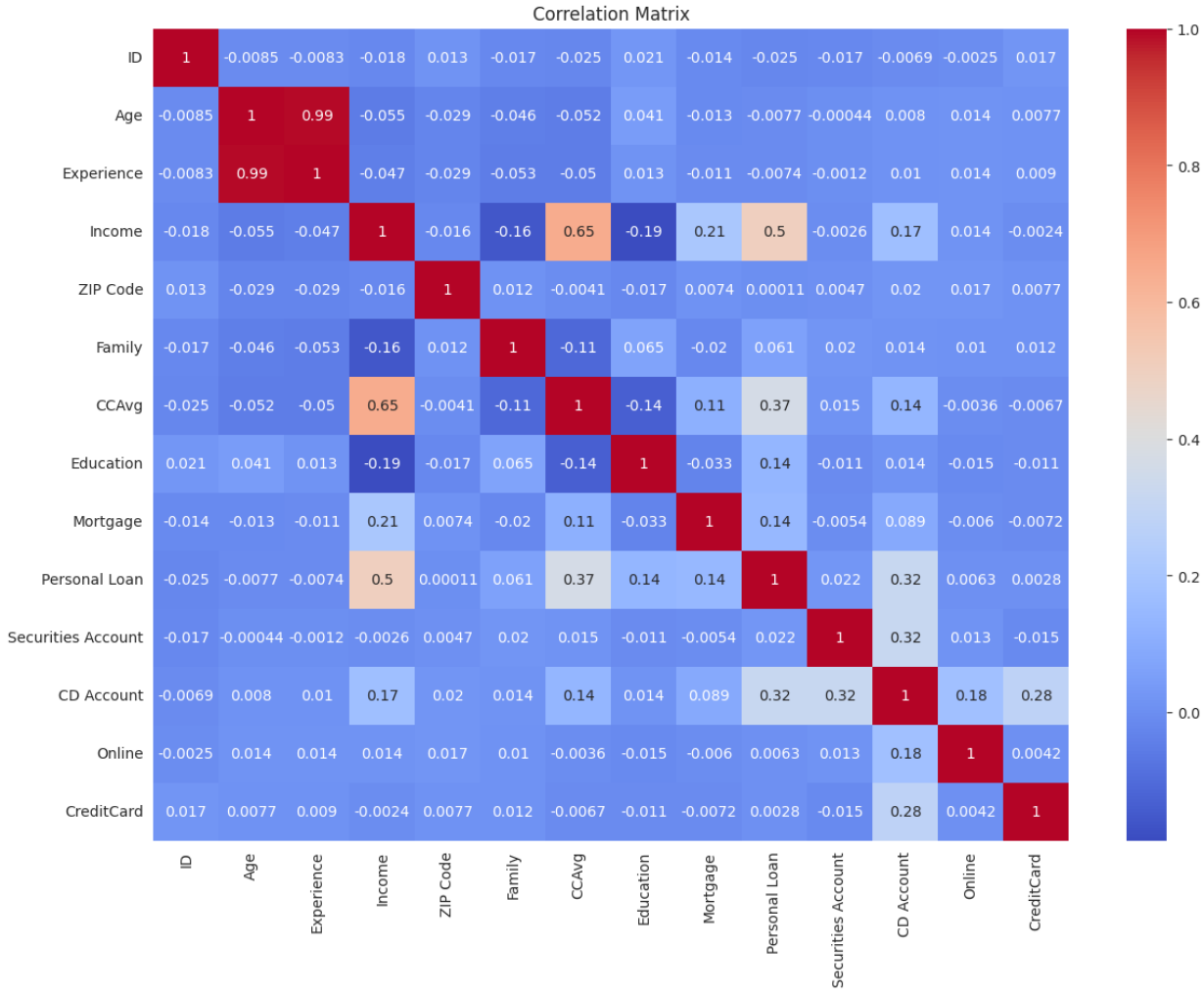

Figure 2: Correlation matrix

From this analysis, it's evident that variables like Income and CCAvg show a strong correlation with the likelihood of accepting a personal loan, whereas other variables such as Experience and Online exhibit a negative or weak correlation. Consequently, we selected variables for our model that have a correlation factor of at least 0.010.

The updated dataset now looks like this:

	Income	Family	CCAvg	Education	Mortgage	Personal Loan	Securities Account	CD Account
0	49	4	1.6	1	0	0	1	0
1	34	3	1.5	1	0	0	1	0
2	11	1	1.0	1	0	0	0	0
3	100	1	2.7	2	0	0	0	0
4	45	4	1.0	2	0	0	0	0
...	...	...	...	...	...	...	...	...
4995	40	1	1.9	3	0	0	0	0
4996	15	4	0.4	1	85	0	0	0
4997	24	2	0.3	3	0	0	0	0
4998	49	3	0.5	2	0	0	0	0
4999	83	3	0.8	1	0	0	0	0

5000 rows × 8 columns

Figure 3: Updated dataset

We wanted to explore more deeply the impact of the variables on the loan decision, so we took the most correlated variables to Personal Loans and compared their influence by visualizing the data points on scatter plots.

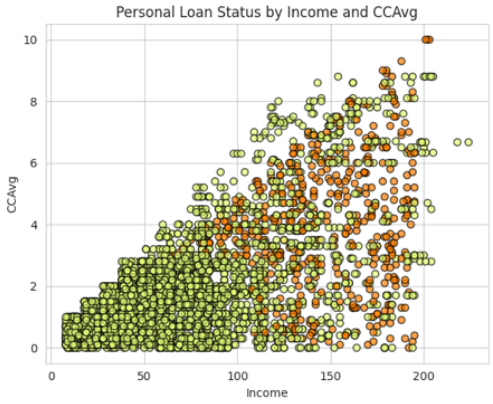

Figure 4.1: Loan by Income and CCAvg

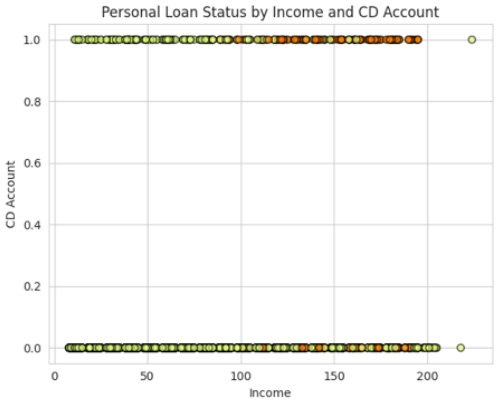

Figure 4.2: Loan by Income and CD Account

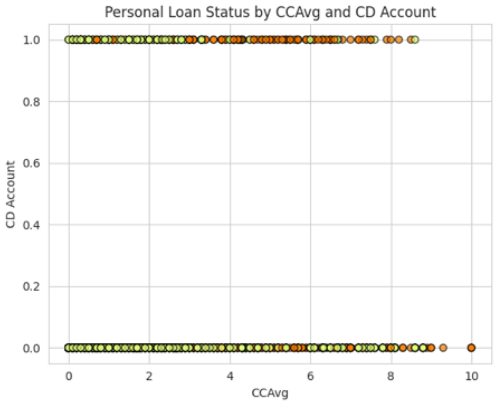

Figure 4.3: Loan by CCAvg and CD Account

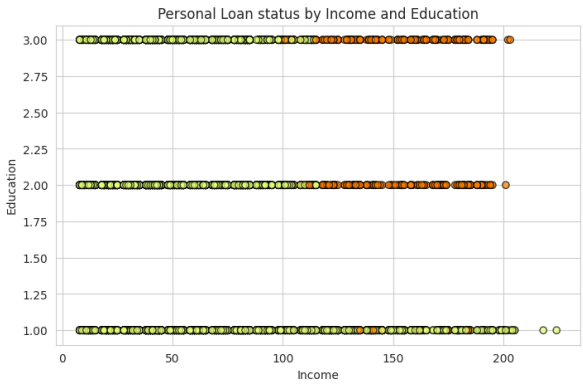

Figure 4.4: Loan by Income and Education

Figure 4.5: Loan by Income and Mortgage

Figure 4

From this, we can see how the highest correlated factor (Income) compares to the other factors. The colour scale shows if a personal loan was granted (Orange) or not (Yellow). Common observations include having a CD account with an income higher than 100 gives you a better chance of being granted a loan by the bank (Fig 4.2). Similarly having an education >=2 shows a higher chance of being granted a loan (Fig 4.4). Fig 4.1 on the other hand does not give a good result which can be seen in the clustering below. These give some kind of insights into the criteria on which people can pay a loan or not and whether the bank should grant them a loan or not.

1.2 Clustering

To explore data mining techniques, a cluster analysis was performed as well to determine if more insights could be extracted.

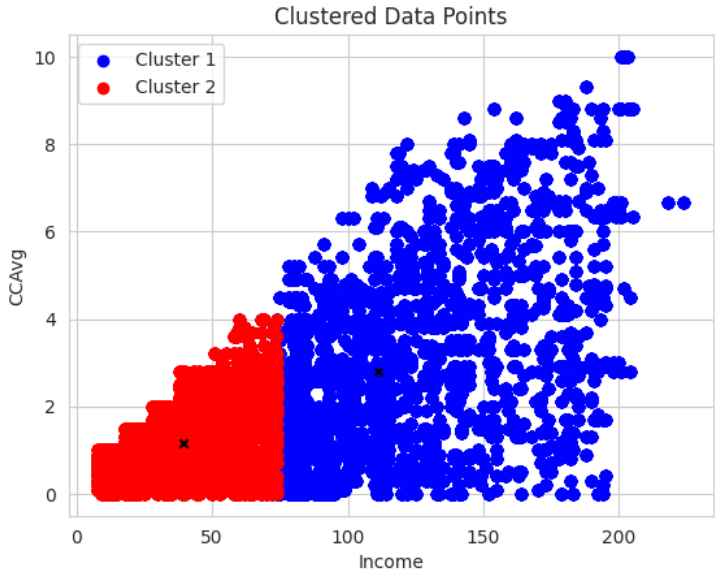

Figure 5

Clustering Fig4.1, using two of the most correlated variables, we can see that there is no clear cluster formed that reflects the loan decision of the bank.

1.3 Neural Network Model

After understanding the data and trying to get insights from clustering, the main Neural Network model was developed to predict the loan decision outcomes. A not-so-complex algorithm was created in Python thanks to the facilities that the frameworks Sklearn and Pytorch offer.

We start by preprocessing our data, which involves scaling numerical features like income and converting categorical features like education into a format the model can understand i.e. Data Standardization. This step ensures our data is uniform and optimized for training.

We split our preprocessed data into training and testing sets. The training set helps our model learn, while the testing set evaluates how well our model can predict new, unseen data.

We then use PyTorch, a machine learning library, to define our data's structure and how it should be processed during model training. We create batches of data, which helps in optimizing the learning process, making it faster and more efficient.

Our model, called LoanPredictionNN, is composed of layers of neurons. It starts with an input layer, which takes in features, followed by hidden layers that help uncover patterns in data, and finally, an output layer that gives the probability of an individual being granted. We use activation functions like ReLU to help the model learn the relationships, and Sigmoid in the output layer to reduce the predictions into a probability range (0 to 1).

For training the model, we use a process called backpropagation with an optimizer and a loss function. In each training epoch (a full cycle through all data), we adjust the model weights to minimize the error between our model's predictions and the actual data. We track both training and testing loss to see how well our model is learning and generalizing to new data.

After training, we evaluate the model's performance on the test set, focusing on accuracy, which tells us the percentage of predictions our model got right.

We then go on to measure the precision which is the key indicator to measure the performance of our model.

After the model was trained several times, it was time to test it and see the results.

2. Results and conclusions

2.1 Confusion Matrix

When deploying the Neural Network, the results were displayed in the following confusion matrix.

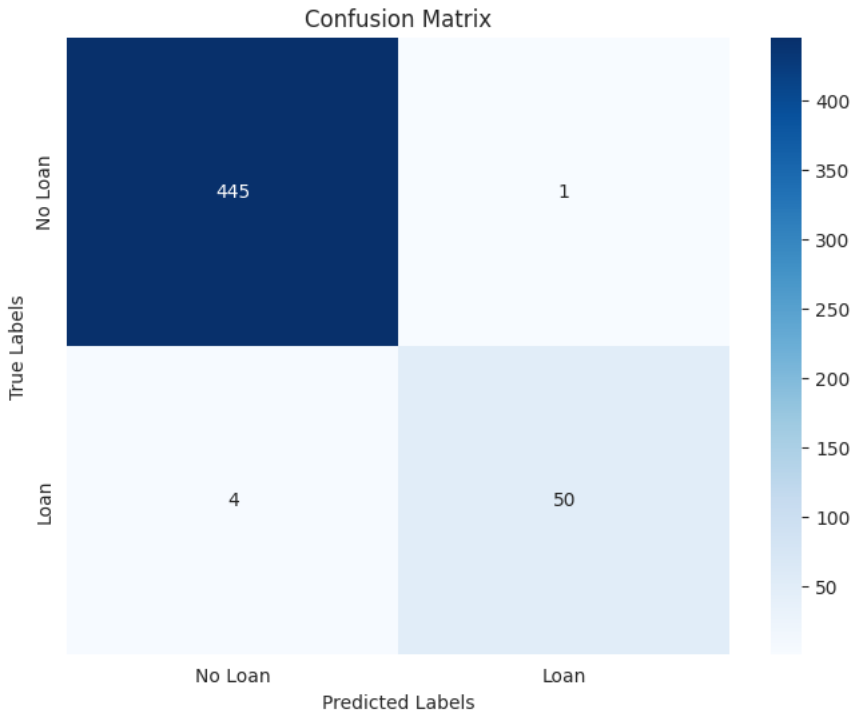

Figure 6: Confusion matrix

The model predicted 50 true positives, 1 false positive, 445 true negatives and 4 false negatives. This means that based on the actual results, the model correctly predicted 50 accepted loans and 445 denied loans, while for one case it accepted one loan when it should have not done it and denied 4 loans when it should have accepted them.

This model has an accuracy of 99% which is considerably good but most importantly, a precision of 98.03%. As this is a loan prediction model, it is crucial to avoid false positives as much as possible. This means, predicting a case as accepted when it should not be, which could lead to potential losses and risks for the bank.

2.2 Loss function

The loss function computes the error of the model on each epoch. It was computed for both the training and test sets.

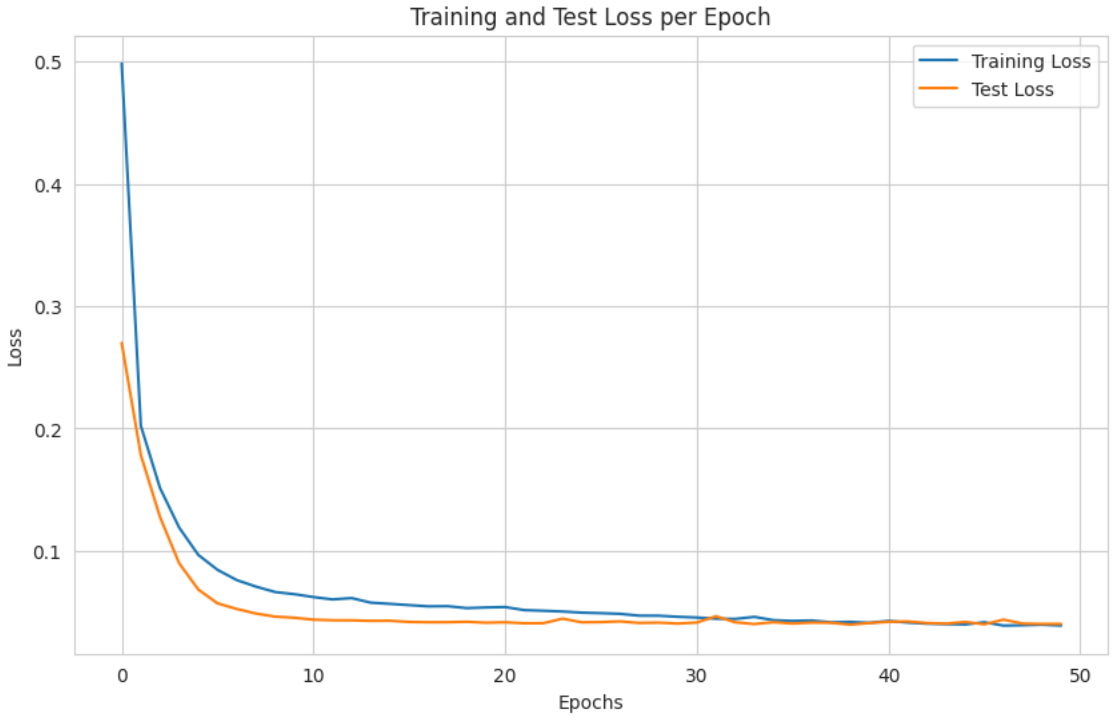

Figure 7: Loss function

The graph indicates that initially, the training error was 0.5 and the test error was approximately 0.28 at the first epoch. Subsequently, there was a rapid decline in error, illustrating swift learning by the model. From epoch 20 onwards, the model's performance began to stabilize, achieving an optimal level. Additionally, the training and test sets displayed comparable performance, with a minimal difference between them, showcasing the model's effective performance on the test set.

2.3 ROC Curve

Another measure that supports the good performance of the model is the ROC curve. This displays the proportion of actual positive cases that are correctly identified by the model (True positive rate) against the proportion of actual negative cases that are incorrectly classified as positive among different thresholds.

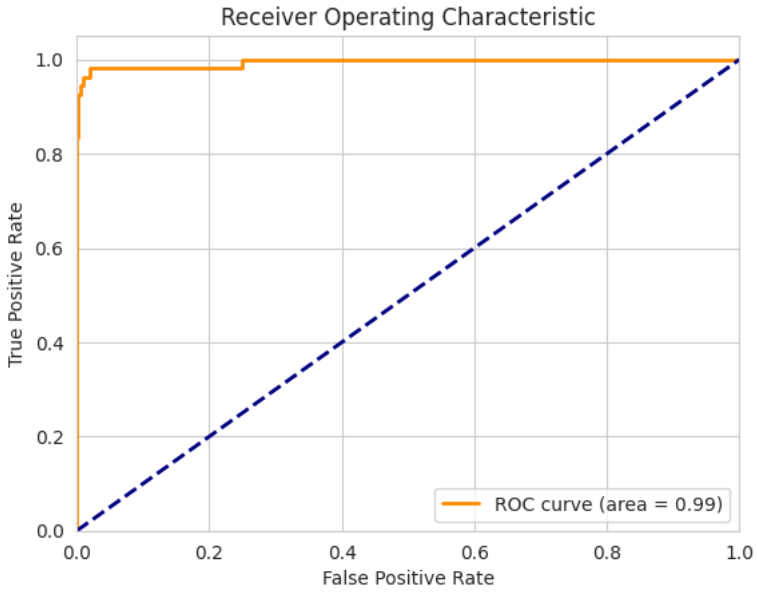

Figure 8: ROC curve

From the ROC curve displayed in the graph, it's evident that the model has an outstanding classification performance with an AUC (Area Under the Curve) of 0.99. This near-perfect AUC score highlights the model's exceptional ability to distinguish between the two classes with high accuracy.

2.4 Conclusion

Overall, a neural network model was built effectively to predict loan decisions using the data of customers at CasaBank. An analysis of the data was performed to get insights of which are the factors that influence more on the loan decision by computing a correlation matrix and scatter plots comparing the variables. After, a cluster was performed but it did not give further insights, so it was dismissed to pass to the creation of the neural network. By training it and testing it, it showed outstanding results like a precision of 98%, an optimal learning rate, and performance demonstrated with the loss function and supported by the ROC curve graph that showed that the model responded adequately among different thresholds with an almost perfect area under the curve of 0.99. This shows that the model is reliable, and the bank can implement it in their operations to achieve an automation of loan decision procedures.