Predicting A Loan Outcome

By: Jahnvi Panchal, Andrew Lam, Sam Le, and Roy Lee

Problem Statement

The problem our group will be working on will be a loan outcome prediction which will help an user predict if their loan will get approved or denied based on many types of factors. Some of the factors will be their employment, credit score, debt-to-income ratio, and much more. By using AI, the application will identify the correlations between the borrower characteristics and loan outcomes which will then predict if the loan application will get approved or denied.

What is the problem that you are trying to solve?

This application will be a good tool for users to help them decide about loan approvals, and the different factors that are important in deciding about loans. We are trying to solve the problem of whether or not a loan will be approved.

If it is not a common game, or a variant of a common game, describe the game. If the game is popular (chess, checkers, Rubik's cube), don't include a description.

We have chosen not to do a game, it is an AI Loan Approval tool.

Approach

Our approach will be to gather data from the Home Mortgage Disclosure Act (HMDA). This website supplies a CSV in order for us to use to train a dataset. We will then use this dataset to apply the KNN algorithm. This algorithm will essentially be comparing pre-existing loans and seeing where the loan stands in terms of approval or denial. Outside of this we will be using Python as a language.

What is the general approach – algorithms, representations

We will be using the K-Nearest Neighbor (KNN) algorithm. This algorithm uses the proximity of data points to make predictions. It calculates the distance between the test data and the trained points. For our specific algorithm we will be calculating this using the euclidean distance.

Description of the software

We used various software such as PyCharm. The software itself to predict the loan takes in a data set. The data set itself takes in a set of inputs such as loan type, property type, purpose, occupancy, amount, gender, income, etc. Once you input the data of the application that you wish to have predicted for, the software will output an approval at 84% accuracy.

Describe the components of the software (e.g., front-end, GUI, AI algorithms). It will help to include a diagram illustrating the components and the input/output relationship between them.

The algorithm used was K nearest neighbor which is a classification algorithm by looking at the closest K neighbors in all possible feature dimensions to determine the target classification.

Programming language, any libraries used, any data

Programming Language - PyCharm, Jupyter Notebook
Libraries - SciKit, numpy, matplotlib, seaborn
Data - Public Domain HMDA Data [Link in References]

Layout of the code

The majority of the code is cleanly laid out in a Jupyter ipynb file. It's stepwise and each step is accurately described. We first take in the hawaii csv and then we organize the data into columns. After this we separate the columns into types of approval or denial. Following this we remove any null values and then encode into categories. After organizing the data sets we use sklearn to train the data set and generate a KNN K value with various evaluations.

What does each file/folder do? Make it understandable to another CS student.

Files

Hawaii.csv - This is an organized dataset from the HMDA Public information for loan approvals or denials.

Loan.ipynb - This is the code itself for training the data set and evaluating precision.

Model - This is the trained model itself after being run through the system.

Test.ipynb - This is a test input to test data and see what result we achieve from given input.

Evaluation

Model evaluated to be 84% accuracy with decent precision and recall on loan originated. However precision for loan being denied is hovering around 50% and below 10% for recall. This is due to the dataset containing a much smaller number of loans being denied versus those originated.

Describe any evaluation of the software. E.g., did you play against your game-playing program? How well did it play?

We did not choose a game; however, as a loan approval software our software definitely could have

had improvements. It was approving loans that we ourselves deemed that should have been denied. We achieved 84% however as high as this looks there could definitely be improvements.

Quantify performance where possible – number of wins/losses, time taken

We achieved 84% test accuracy among a very large set of data, roughly 20,000+ data sets of various loan types. The time taken to train the set took around 4-5 hours. The time to use the loan approval software itself is only around 5 minutes.

Conclusion and future work

At the end of the project, our group realized that this project truly helped us learn more about AI implementation in daily lives. Our group wanted to go further than just creating a game which is why we decided to make a loan predictor application for users to apply to in their lives. For future work, our group believes we can increase the accuracy of the test, such as making it go higher than 84%. This would help it have better accuracy and make the loan approval tool be more accurate for users to use in their lives.

Summarize the main lessons learned from the project.

From this project, our group learned to use Python better to make a loan prediction application. First, we had to choose an application that could be different from others, and then we realized that we have to figure out the best algorithm to apply. Our group had to research more about the KNN algorithm as we decided on that one. Afterwards, we had to find the best dataset online to keep the best accuracy. The coding was difficult for us as we faced many issues, and had to keep researching to solve the small errors we kept getting as we moved along. After solving each issue, we got to the accuracy of 84% which we realize we can still improve as we keep working on the project in the future.

Describe ideas for improvement

For future purposes we believe we can increase the accuracy of this test. As of right now there is still a margin for error as a certain amount of loans will be marked as approved whereas they might not be approved. At 84% we would hope to increase accuracy and correct the loan approval tool to be absolutely accurate.

References

Any papers/tutorials/datasets that you used

HMDA Public Data for Training
https://www.consumerfinance.gov/data-research/hmda/historic-data/?geo=hi&records=all-records&field_descriptions=labels

Github for our Project
https://github.com/pas-trop-de-zele/loan-predictor-knn