Loan Repayment Proposal

Problem Statement and Background

In the lending industry, a company offers loans to lenders in return for a profit. If a lender repays the money, the company makes a profit. On the other hand, if the lender does not pay off the money, the company takes a lost. Therefore, in this proposal, the problem we are trying to address is to predict the loan risk on a given loan application based on previous applications. Machine Learning has become a popular approach in recent years in analyzing data and identifying patterns. With a good model, we will be able to identify patterns from previous data, and accurately predict future data.

Goal

The goal for this proposal is to accurately predict if an applicant will default on the loan based on the applicant's personal information. This will help the company to make decisions on issuing loans, so that the company can reduce the risk and maximized its profit.

Data Cleaning and Exploratory Analysis

There are three csv files that contains numerous numbers of data, each csv files consisting of a column that correlates to either one of the other csv files. The initial approach towards data cleaning is identifying relevant columns that are useful to predicting loan risk and dropping unnecessary columns to reduce the number of columns. After identifying relevant columns, the next step will be to identify missing values in the dataset. Image 1 contains a heatmap in seaborn which illustrates columns with missing values. This step help with analyzing data later by either giving the missing values a default data or dropping the row.

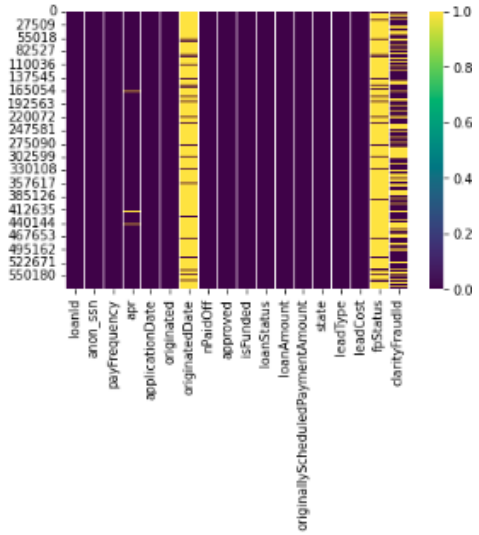

Image 1

Another important step in data cleaning is to drop duplicates so that we do not keep any duplicated rows in the dataset.

After data cleaning, the next step will be to perform Exploratory Data Analysis (EDA). A good start to this will be to look at the size of the dataset using the df.shape. Another important feature in pandas is df.describe() function to obtain a statical summary on each column. Image 2 provides the statistical summary for the numerical values. This helps with identifying outliers in the dataset. Another key feature used in the data analysis was the df.value_counts() method, it gives a good understanding on how concentrated the data are in the dataset and how many unique elements are there in each column of the dataset. The hist() method was also really helpful in giving some visualization of the dataset.
The Exploratory Data Analysis.ipynb file provides the codes and analysis results from the dataset.

	apr	nPaidOff	isFunded	loanAmount	originallyScheduledPaymentAmount	leadCost	hasCF
count	573760.000000	577658.000000	577682.000000	575432.000000	577682.000000	577682.000000	577682.000000
mean	553.080972	0.037887	0.067480	514.245084	1428.897209	7.854389	0.619187
std	110.046159	0.333366	0.250852	320.939929	925.009141	12.853451	0.485587
min	0.000000	0.000000	0.000000	0.000000	-816.710000	0.000000	0.000000
25%	490.000000	0.000000	0.000000	350.000000	1023.640000	3.000000	0.000000
50%	590.000000	0.000000	0.000000	500.000000	1245.250000	3.000000	1.000000
75%	601.000000	0.000000	0.000000	500.000000	1615.660000	6.000000	1.000000
max	705.590000	21.000000	1.000000	5000.000000	19963.630000	200.000000	1.000000

Image 2

Methods

Merging csv files

The initial approach to merging three csv files is to identify important columns and merging the important columns all into one csv files. This step is important in plotting important analysis later. The merged dataset is called MergeData.csv. Besides, it is important to replace Boolean information to 0 and 1 or -1 and 1. This helps will plotting simpler and reduces the number of errors occurring while trying to plot the graphs because integers are easier to match compared to strings.

Filtering

The merged dataset was filtered to use rows that have started underwriting. The reason for this approach is to filter out rows that do not have an underwriting process because we are trying to assess the financial risk and underwriting is the fundamentals of evaluating risk. Not having an underwriting process makes it difficult to assess the risk probability.

Another filter after that will be to look at the loan status. Only the rows with external collection, internal collection and paid off loan were used in the dataset. The reason for this approach is to predict if the applicants will pay off their loans or fail to pay off their loans. When they fail to pay off their loans, this resorts to external collection and internal collection. This will be explained more in details in the analysis part why these three options were used for the filtered dataset.

Crosstab Table

The crosstab table is an important method in showing the relationship between multiple variables. This helps in visualizing our data in a table format and is a simple task to do to see if there are any relationship between the features before moving to plotting the graph.

Visualization and Analysis

Based on the EDA, after performing the tasks in the method section mentioned above, some meaningful insights were obtained from the dataset. The goal is to see if there are any relationship between features in the dataset with the loan status. Loan status with value of 1 means the loan was paid off while value of 0 means the loan have some form of collection. Image 3 and 4 shows the relationship between the number of loans a client has paid off in the past and the loan status. Based on image 3 and 4, which are scatter plots, as the number of loans a client has paid off in the past increases, the probability that the client will pay off the loan is high. Image 4 shows a logistic regression illustration of the plot, suggesting that when the client have 7 past loans paid off in the past, there is an extremely high probability that the client will pay off the loan.

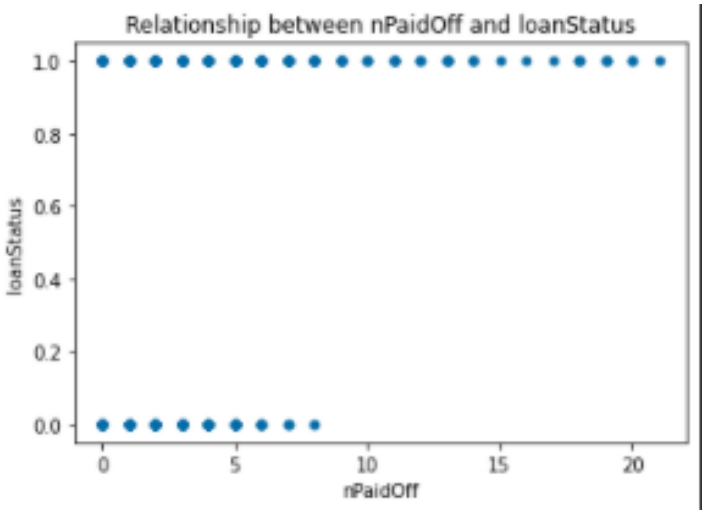

Image 3

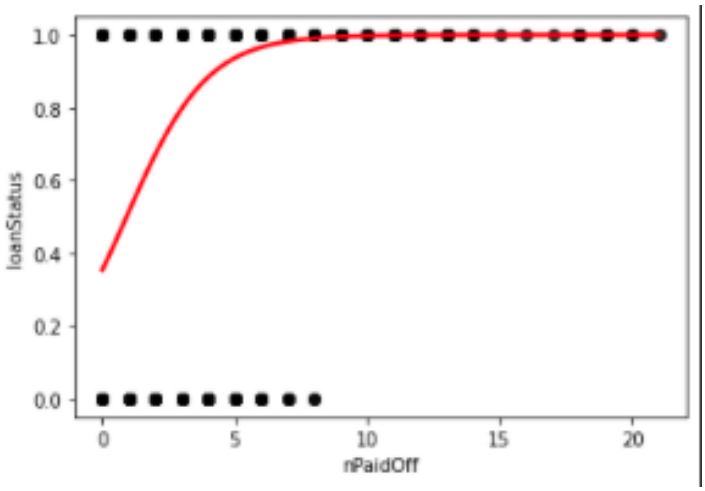

Image 4

Another key insight found while exploring the dataset is the relationship between the number of loans a client has paid off in the past and the errorPayementExist. ErrorPaymentExist means that there exists at least one unsuccessful payment based on its previous payment history from the payment.csv file. The value 1 means there exist at least one unsuccessful payment from the client's payment history and the value 0 means there does not exist any unsuccessful payment, meaning the client paid off the loan without any issue. Based on image 5, the image clearly shows that as the number of loans a client has

paid off in the past increases, the probability that there does not exist any unsuccessful payment also increases.

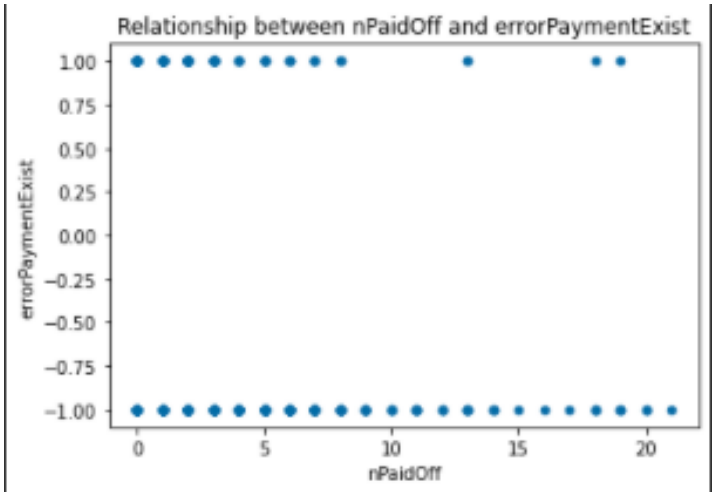

Image 5

Apart from these, another meaningful insight found from the dataset is the relationship between the clear fraud score and the loan status. The plot is a density plot, which is a different plot compared to the previous plots. The previous plots were scatter plots, however image 6 is a density plot. It's a representation of the distribution. Based on image 6, as the clear fraud score increases, the probability that the client will pay off the loan also increases.

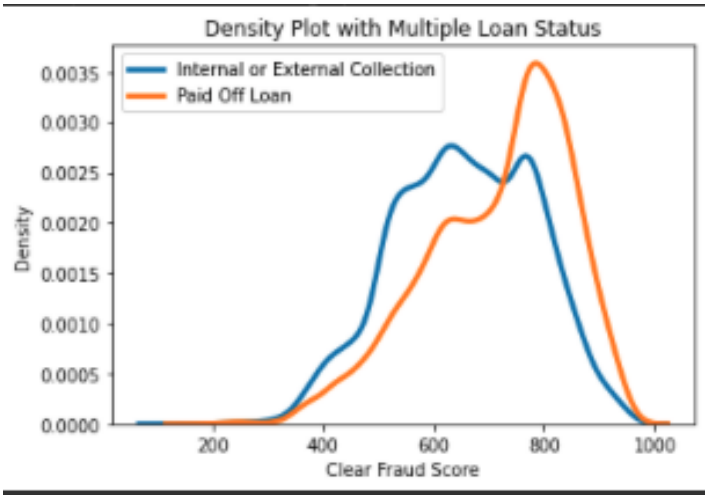

Image 6

Therefore, the loanStatus, clearFraudScore, errorPaymentExist and nPaidOff are important features that will be used in predicting the probability a loan is risky.

Based on the methods and analysis, some important information will be helpful in accurately predicting the probability of a risky loan. Although there is a clearFraudScore column for predicting the client will not default, having credit score information will be beneficial in predicting the probability. The credit score provides an overview of the client credit history and attributes that predict how a client will behave in the future. The purpose of the loan will also provide a meaningful insight in terms of knowing how the money is going to be used by the client. By having the purpose of loan information, clustering will be a useful way of grouping clients and observing any meaningful insights as it will be beneficial for understanding which cluster the client will be to predict the probability. Client's that fall under the same cluster will share similar financial characteristics, making prediction more accurate.

Plan

From the dataset, the first thing to work on will be to split the data to training sample and test samples. There are numerous ways to split the data, the train_test_split method from scikit- learn was used to split the data to 80% training sample and 20% test sample. The reason having a test sample is it acts as held-out sample, meaning the test sample will be used to evaluate the sample. Another popular way to split the data will be the K-folds cross validation method. For the dataset, the train and test sample will consist of rows and columns based on the filtering method mentioned previously. Only 4 features will be used to develop the model. The loanStatus, clearFraudScore, errorPaymentExist and nPaidOff features will be used to train the model. The machine learning algorithm used for the model will be the logistic regression. The reason logistic regression is ideal as it is a classification algorithm used to find the probability that the loan is risky or not risky. In this case the loanStatus feature is what the model will output, either 1 or 0. The accuracy of the logistic regression is approximately 66%. Another popular approach will be the support vector classification algorithm. The accuracy of the support vector classification is approximately 64%. However, the logistic regression provided a more accurate accuracy.

Conclusion

From the dataset, there are many features that is important in predicting a risky loan. Logistic regression is a better choice in terms of trying to predict a risky loan compared to other classification algorithms. The applicants that usually pay off their loan have a high clearFraudScore, a -1 value for errorPaymentExist and a high nPaidOff. These characteristics will be used to predict if a loan is risky.