2022

Credit-Risk Model

DATA PRE-PROCESSING

TASNEEM HOQUE

Contents

Executive Summary

The goal is to construct a Credit Risk Model that will measure the probability of “default” for every personal account. Data contains all the information of loan transaction which includes distinct ID for each borrower, date, loan amount, state, term, interest rate and some other primary information. Some interesting observations include the issue date which all belonged to the same year for this dataset.

The goal is to have as many numerical or Boolean values as possible. This will make calculation easier going forward. All categorical data needs to be kept down to the least different types of categories possible. Since the data set belongs to an affiliate bank based in the United States, the value in dollars needs to be converted to EURO. Every categorical variable needs to be quantified. Similarly for columns we only care about whether they provide positive or negative connotations and change them accordingly with values of 0’s and 1’s. Furthermore, when we're measuring credit worthiness, we need to be extremely risked averse and distrustful of any unavailable data, therefore, for any missing values we assume the worst-case scenario.

We are against deleting any data; therefore, any missing data would need to be filled in. Any column that doesn’t play a significant role or can be re-calculated from existing data is safe to delete.

Objective

We will be cleaning the “load-data” dataset in order to be used at a later stage to create a predictive model. The objective is to obtain a clean and processed loan data set that can be handed over to the next people in the analytical chain. We must be careful of how we decide to fill-in for missing data as these data are manually filled in and there is no incentive to withhold data.

Data Summary

For this pre-processing we will be using the “loan-data” dataset. It consists of 1,000 records which are from 2022 of the various loan processed by the borrower in US. The following are the attributes of the dataset:

Attributes	Description
ID (id)	A unique Letter of Credit assigned ID for the loan listing.
Issue Date (issue_d)	The month which the loan was funded.
Loan Amount (loan_amnt)	The listed amount of the loan applied for by the borrower. If at some point in time, the credit department reduces the loan amount, then it will be reflected in this value.
Loan Status (loan_status)	Current status of the loan.
Funded Amount (funded_amnt)	The total amount committed to that loan at that point in time.
Term (term)	The number of payments on the loan. Values are in months and can be either 36 or 60.
Interest Rate (int_rate)	Interest rate on the loan.
Installment	The monthly payment owed by the borrower if the loan originates.
Grade	LC assigned loan grade.
Sub Grade (sub_grade)	LC assigned loan sub grade.
Verification Status (verification_status)	Indicates if the borrower’s income was verified by LC, not verified, or if the income source was verified
URL	URL for the LC page with listing data.
State Address (addr_state)	The state provided by the borrower in the loan application.
Total Payment (total_pymnt)	Payments received to date for total amount funded.

Table 1: Attributes of the ‘loan-data’ dataset.

Observations:

The following table briefly describes the observation made on each of the attributes in the "loan-data" data set:

Attributes	Description
ID (id)	• The code is numeric, consists of 6 to 8 numbers.
Issue Date (issue_d)	• All the issue dates are from the same year, hence, only the month information is important to retain. • There are _ records of issue dates missing.
Loan Amount (loan_amnt)	• Loan amount is of type integer, with no decimal values. • There are _ records of loan amount missing.
Loan Status (loan_status)	• Current status of the loan can range from current, late, grace period or fully paid.
Funded Amount (funded_amnt)	• Has integer values • They are equal to the loan amount.
Term (term)	• This is a string with an integer followed by a string "month"
Interest Rate (int_rate)	• The rates are in percentage values, usually more than 1 but some less than 1.
Installment	•
Grade	• Letter grades ranging from A to G
Sub Grade (sub_grade)	• Letter grades with a number assigned from 1 to 5.
Verification Status (verification_status)	• Verification status is either 'Verified', 'Source Verified' or 'Not Verified'.
URL	• This is a URL to the borrower's profile; they all have the unique customer ID at the end.
State Address (addr_state)	• It is always a two-letter code indicating 1 of the 50 states.
Total Payment (total_pymnt)	• This is a floating-point number.

Table 2: Observations made on the attributes of the 'loan-data' dataset.

Limitations:

The limitation was with opening the csv file because there was error with the int_rate. The values were corrupted with question marks that replaced zeros in front of the decimal places. As a result, the file could not be opened using pandas.

This issue was overcome using text to columns feature in excel. This transformed all the strings to numbers then reformatted the numbers to be decimals.

Outcome:

In the "loan-data" dataset we will be making slight modification to allow easy use of the data. They will all have the original **1000 datasets**. We will be removing columns like grade and URL which are redundant. They can be re-generated from existing data if required. We will also rename columns to better reflect their purpose. We will also include new columns to show the currency in both USD and EURO. The final version of this dataset will include the following columns with all original information retained:

1. **id**
2. **issue_date**
3. **loan_amnt_USD**
4. **loan_amnt_EUR**
5. **loan_status**
6. **funded_amnt_USD**
7. **funded_amnt_EUR**
8. **term_months**
9. **installment_USD**
10. **installment_EUR**
11. **total_pymnt_USD**
12. **total_pymnt_EUR**
13. **exchange_rate**
14. **issue_date**
15. **loan_status**
16. **term_months**
17. **sub_grade**
18. **verification_status**
19. **state_address**

Design/Method/Approach

The following steps were performed to pre-process the existing data:

1. **Dataset Selection:**
    - We begin by selecting the “loan-data” dataset that contains all the loan transaction details.
    - We format the column in the excel file with int_rates to change them to floating point variables that can be manipulated.
    - Then we review the source data to understand the content and make appropriate changes.
    - Then we categorize the data to format consistently and convert to numbers or dummy variables were possible.
2. **Renaming columns:**
    - We rename columns to better reflect their purpose.
    - Issue_d renamed to issue_date
    - addr_state renamed to state_address
    - term renamed to term_months
3. **Changing data to numbers:**
    - Term data was converted to month only by removing ‘month’ string and converting to integer.
    - Issue_date was changed to just months by removing the whole date since all records are from the same year.
    - Changed loan_status to 1 or 0, classified current, default, Fully Paid, In grace period and issued as good or 0 and Charged Off and late as bad or 1.
    - Changed verification_status to 1 or 0, classified Not Verified to 1 and Verified or Source Verified to 0.
    - Changed interest values to between 0 and 1 from percentage values.
4. **Missing data:**
    - Assumed worst-case scenarios for all missing data
5. **Redundant data:**
    - Removed data that can be re-generated from existing columns
6. **Conversions:**
    - Made currency conversions from USD to EUR using the average of opening, closing, high and low prices of stock.

Detailed Process

Renaming Columns:

Columns were renamed to better reflect their purpose in the database. Here issue_d was renamed to issue_date, addr_state renamed to state_address and term renamed to term_months, after removing strings from the values and filtering to numbers.

Changing Text to Numbers:

Several data that were text was converted to numbers. Such data includes the issue_date, the original formatting for this column was date: 2022-10-15, since they are all dates from the same year and has the same date, they are now just numbers with month as an integer: 5.

loan_status is changed to 0 or 1 depending on the classified category of good or bad, respectively. If the category is Current, Default, Fully Paid, In grace period, it is marked good and if the category is Charged Off or late it is marked as bad. For ex. 'Current' became '0'.

verification_status is changed to 0 or 1 depending on whether the income was verified or not, respectively. Not verified is classified as Verified and Verified or Source Verified is classified as verified. For ex. 'Verified' became '0'.

Interest_rates were originally strings with formatting issues on zeros before the decimal point, it is now fixed to be a floating-point type that has values between 0 and 1 instead of percentages. For ex. '13.33' became '0.1333'.

term column is changed to integer that contain just the term which is either 30 or 65, and stripped of the ending 'months'. For ex. '30 months' became '30'.

Handling Missing Data:

Missing data are found in the columns issue_date, loan_amnt, loan_status, funded_amnt, term_months, installment, grade, sub_grade, int_rate, verification_status, state_addr and total_pymnt. All missing data are set to imagine the worst-case scenario.

The missing data in issue_date is simply given a value of 1, indicating that the day the loan was issued is the earliest it can be issued.

loan_amnt and funded_amnt are set to the highest value in the column, which assumes that the borrower has taken out maximum loan possible keeping in mind the worst-case scenario.

term_months is set to minimum of the value in the column, which mean the time that the borrower has to pay the money back is minimum, assuming worst-case scenario.

Missing installment values are replaced by the loan_amnt multiplied by the int_rate divided by the term which is how much they have to pay every month to pay off the loan.

grade column is removed, missing sub-grades are calculated from the grades, assuming worst level of the grade, for ex. If grade was B, then the sub-grade assumed will be B5. If grade is missing, then replace it with the worst possible sub_grade which is 36.

Missing int_rate were set according to the sub_grade level because int_rate was high when sub_grade was high and int_rate was low when sub_grade was low.

verification_status is set to 'Not Verified' or 1, assuming worst-case scenario.

state_addr is set to the most common state in the column which is 'CA'.

total_pymnt is set to 0 for all empty cells, assuming worst-case scenario.

Removing Redundant Data:

Redundant data is all data that can be generated from the existing table or is of no use during later use of the database. The data that were redundant were the grades column and the url column. The grades column was used to predict sub_grades that were empty, however, there was no use of the grades data once sub_grades was taken care of.

Converting Currencies:

Currencies were converted where necessary from USD to EURO based on a new column called exchange_rate. Exchange_rate is a rate with which the currency is converted based on the issue_date of the particular loan statement. The columns that were converted to Euro are loan_amnt_EUR, funded_amnt_EUR, installment_EUR and total_pymnt_EUR.

Conclusion

All the changes to the data are made to the original data. The code written to make said changes are in Python using the Pandas library. Additional file used to read the currency conversion data to convert all currencies in USD to EURO based on their issue dates.

Reference

No external sites referred to.

Appendix

1. Link to notebook: Tasneem_Hoque_Assignment1 - Jupyter Notebook