Module 3 Project | Lending Club

Trang Tran

CPS Analytics, Northeastern University

ALY6020 | Predictive Analytics

Professor Prashant Mittal

Nov 27, 2023

Introduction

LendingClub, the world's largest peer-to-peer lending platform, revolutionizes the financial landscape by connecting borrowers and investors in an innovative online marketplace. The analysis focuses on Q1 2018 data, with the main objectives of building logistic regression and Support Vector Machine (SVM) models to predict High-Low and High-Medium loan amounts in two responsible variables. We delve into the significance of variables in the logistic model, shedding light on their predictive contributions. Finally, we provide a comparison of accuracy metrics across classification models, including overall accuracy, precision, and recall, guiding our recommendation for LendingClub based on these critical performance indicators.

Data Description and Preprocessing

The original 'LoanStats' dataset contains 96829 observations with 52 columns of features. Whereas there are 38 numerical variables and 14 categorical variables. However, 'emp_length' representing the employment length in years is in the wrong type of data. It can be a good numerical feature to predict the loan amount, so we extracted the text string and converted this

```
# Print only the categorical columns
listofcat = list(df.select_dtypes(include = ['object']))
listofcat
```

```
['addr_state',
 'disbursement_method',
 'emp_length',
 'grade',
 'home_ownership',
 'initial_list_status',
 'loan_amnt_cat',
 'loan_amnt_HighLow',
 'loan_amnt_HighMedium',
 'loan_status',
 'pub_rec',
 'sub_grade',
 'term',
 'title']
```

```
# convert the 'emp_length' variable into integer type

# extract the text
df['emp_length'] = df['emp_length'].str.extract('(\d+)')

# Convert the extracted numeric part to integer
df['emp_length'] = pd.to_numeric(df['emp_length'], errors='coerce')
```

Figure 1: Categorical variables

variable into an integer type. It is noted that possible values in this feature are between 0 and 10 where 0 means less than one year and 10 means ten or more years.

In part 1, we perform feature outlier analysis and eliminate up to 1% percentile of rows each based on the following columns: ‘dti’, ‘annual income’, and ‘delinq_2yrs’. These three features witnessed a huge transformation after removing the outliers (Figures 2).

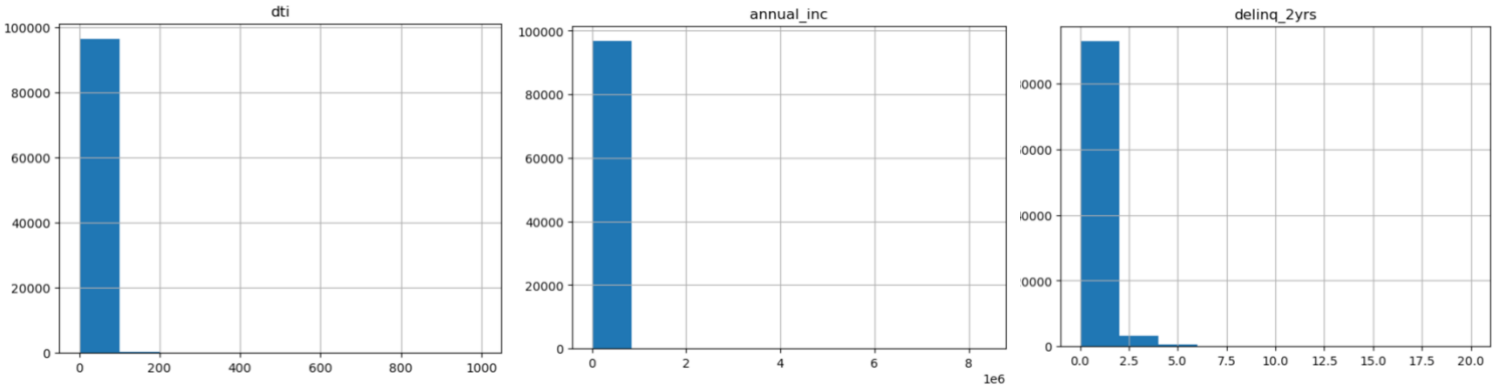

Figure 2: Before removing outliers

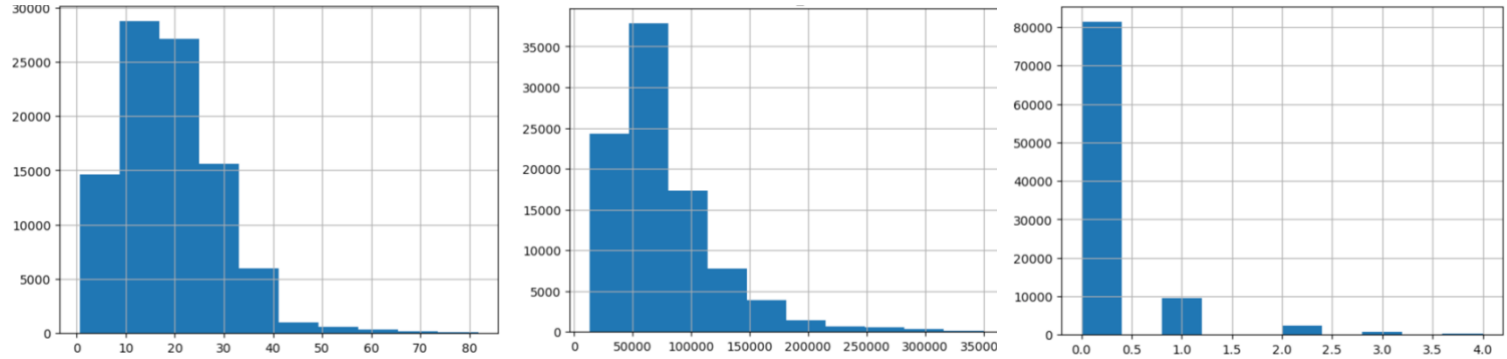

Figure 2: After removing outliers

Logistic Regression Models

Among the categorical variables shown below in Figure 1, we did one-hot encoding of three variables 'disbursement_method', 'home_ownership', 'pub_rec'. This process results in the creation of binary "dummy" variables that represent each category within the original categorical variables. Besides, we dropped off several variables that are bad predictors to the model,

including: 'addr_state', 'grade', 'initial_list_status', 'loan_amnt_cat', 'loan_status', 'sub_grade', 'term', 'title', 'installment', 'int_rate', 'loan_amnt'. The new dataset has a total of 47 columns. Afterward, we map response values 'loan_amnt_HighLow' and 'loan_amnt_HighMedium' to numerical values of 0 and 1.

With the dummy variables in place, we can now distinguish between Features (45 columns) and Target variable ('loan_amnt_HighLow' or 'loan_amnt_HighMedium') in two separate Logistic Regression models. We need to pay attention to the NA values in each target variable in each model and make sure to align y (target variable) with X (features) after dropping rows of NAs. After splitting the data set into the train and test sets with a ratio of 80%-20% respectively, fitting the Logistic Regression model on the train set and running the predicting process on the test set, we have the results of two models as follows:

```
Accuracy:  0.9889785251675945
Precision:  0.9904425242210002
Recall:  0.9968375280010542
F1 Score:  0.9936297366519997
Confusion Matrix:
 [[1139   73]
 [  24 7565]]
```

Figure 4: Logistic Regression Model on High-Low response

```
Accuracy:  0.9084598698481562
Precision:  0.8915863840719332
Recall:  0.9164246105096382
F1 Score:  0.9038348850836643
Confusion Matrix:
 [[7717  844]
 [ 633 6941]]
```

Figure 4: Logistic Regression Model on High-Medium response

Figure 3 shows evaluation metrics on the Logistic Regression Model of the high-low response. All metrics are roughly 99%, indicating good results of the classification model. Meanwhile, figure 4 presents lower results (~90%) of the high-medium response classification model. Overall, these logistic regression models performed well in classifying loan amount responses for the Lending Club.

While ‘sci-kit learn’ does not directly compute p-values of logistic regression models, we still can display the coefficients of the logistic model and interpret the significance of variables based

on the magnitude and sign of the coefficients. Figure 5 below provides details of coefficients in the ‘High-Low’ response model.

```
Intercept: -0.00035430181423326735
all_util: -0.017772134416078722
annual_inc: -3.823520385053507e-05
avg_cur_bal: -9.143899487408759e-05
delinq_2yrs: -5.129910951051577e-05
dti: -0.005751468833328869
emp_length: -0.001586574521993379
inq_fi: -0.0002839672811746936
inq_last_6mths: -0.0001541874309633287
max_bal_bc: -4.768873439315704e-05
mort_acc: -9.13700945006883e-05
num_accts_ever_120_pd: -0.00015231971688034094
num_actv_bc_tl: -0.0007867709868588918
num_actv_rev_tl: -0.0013393024699652754
num_bc_sats: -0.0009611678631688237
num_bc_tl: -0.0013772044596879137
num_il_tl: -0.0014319229498592023
num_op_rev_tl: -0.0019905720034976786
num_rev_accts: -0.0029439240540536076
num_rev_tl_bal_gt_0: -0.0013358611676339864
num_sats: -0.002665111427890751
open_acc: -0.0026706550864711385
open_acc_6m: -0.00029425930858725314
open_rv_12m: -0.0004610545824895254
out_prncp: 0.0008037907920209672
pub_rec_bankruptcies: -4.1356246419099036e-05
revol_bal: 2.530564484080191e-05
revol_util: -0.00012394621576170868
tot_cur_bal: 2.5949225508444856e-05
tot_hi_cred_lim: -1.4413652821229 46e-05
total_acc: -0.004486540940756769
total_bal_ex_mort: -8.5045002286808e-06
total_bc_limit: -2.0987745683809534e-05
total_cu_tl: -0.00023921866622853032
total_il_high_credit_limit: -2.1779795038222424e-06
total_pymnt: 0.0005637060203933871
total_rec_int: -0.0011047347868530573
disbursement_method_Cash: -0.00033137210636776494
disbursement_method_DirectPay: -2.2929707865503e-05
home_ownership_ANY: 0.0
home_ownership_MORTGAGE: -7.025416246681395e-05
home_ownership_OWN: -5.480954669833066e-05
home_ownership_RENT: -0.0002292381050681224
pub_rec_0: -0.0003119093601765158
pub_rec_1: -4.223398207740882e-05
pub_rec_2 or more: -1.5847197934281067e-07
```

Figure 5: Coefficients of 'High-low' logistic regression model

Support Vector Machine Models

Firstly, we implement the grid search which helps to search for the best combination of hyperparameters for the model without actually running the models individually.

(I could not compute the result of the cross-validation code section. It was loading for several hours so I gave up on this part.)

References

ALY6020.70767.202415. Module 3. Slides-Sample-Programs-Other Documents.

P3_SVM Sample_program.jpynb