# SBA Loan Approval Analysis & Prediction

TEAM 13

You Li NUID: 002824033

Yanqi Ma NUID: 002774114

## Ⅰ. Introduction

1. **Background**

Since its establishment in 1953, the U.S. Small Business Administration (SBA) has been dedicated to promoting and aiding the development of small enterprises within the U.S. credit market. Small businesses are a crucial source of job innovation in the United States, and the growth and development of these entities have significant social value by generating employment opportunities and reducing unemployment. The SBA mitigates bank risk and stimulates loan issuance to small businesses through its loan guarantee program. Nevertheless, the proportion of default on these guaranteed loans has been a contentious issue, sparking a broad discussion on the role of government in the credit market.

1. **Motivation**

Historically, the small business loan guarantee program has propelled many startups to success, such as FedEx and Apple Computer. However, there have been numerous instances of small businesses and startups defaulting on their SBA-guaranteed loans, which has been a focal point of controversy. This has led to two perspectives: one believes that credit markets can operate efficiently without government intervention, while the other argues that the social benefits of job creation by businesses receiving government-guaranteed loans far exceed the costs associated with defaulted loans. Therefore, it is imperative to explore the effectiveness of the SBA loan guarantee program and its impact on banking decisions.

1. **Goal**

The objective of this project is to classify SBA-approved loans into “Paid In Full” and “Charged-Off” categories through machine learning, aiming to provide banks with a more precise framework for risk assessment to optimize the loan approval process and enhance the overall efficiency of the small business credit market. Specifically, we want to assist the Small Business Administration (SBA) Loan Guarantee Program and banks in accurately pinpointing which borrowers are at a higher risk of default. If time-permitted, we want to analyze the main factors influencing the default rate of guaranteed loans, and assess the program's effectiveness in creating jobs and promoting social benefits.

## Ⅱ. Methodology

1. **Data Cleaning**

Identify and correct errors and inconsistencies in the data to improve its quality. This includes:

* + - * **Removing Duplicates**: Check for and eliminate any duplicate records that may skew the model training.
      * **Handling Missing Values**: Identify columns with missing values and decide on a strategy for handling them (e.g., imputation, deletion).
      * **Converting Data Types**: Ensure that each column is of the correct data type (e.g., numeric, categorical). Implement data type conversion when necessary.
      * **Addressing Outliers**: Looking for any anomalies in the data that might need to be addressed.

1. **Exploratory Data Analysis (EDA)**

Begin EDA by examining the distribution and relationships between factors such as loan sizes and industry types through visualizations and statistical summaries. This helps identify key factors influencing loan risk, spot anomalies, and challenge assumptions, guiding hypothesis and model development.

1. **Feature Engineering**

Use one-hot encoding to convert categorical data into numerical data, create new features based on the characteristics and assumptions of the dataset, standardize the values, and explore feature interactions to enhance predictive performance. We use the chi-squared test to select features, pinpointing the 10 features that are most closely associated with our label. This process prepares the data for machine learning, thereby improving the accuracy of predictions for loan approvals.

1. **Model Selection and Training**

For such a classification problem that aims to predict a binary outcome (loan will be Paid in Full or Charged-Off), several machine learning models could be suitable. Here are some models that we intend to employ for this project:

* **Random Forest**: Random Forests are particularly effective for predicting borrower defaults because they can process complex datasets with many features, naturally handle imbalanced data, and maintain high accuracy by averaging multiple decision trees to reduce overfitting.
* **Logistic Regression**: Logistic regression estimates the probability that a given input point belongs to a certain class. It could effectively handle binary classification of loan Paid In Full or Charged-Off.
* **Support Vector Machines (SVM)**:SVM excels in high-dimensional spaces by finding the optimal boundary between classes, using the kernel trick for non-linear separation, making them highly effective for distinguishing between approved and denied loans with complex feature relationships.

1. **Model Evaluation and Tuning**

Evaluate model performance using appropriate metrics (e.g., accuracy, precision, recall, F1 score) and use techniques like cross-validation for more robust evaluation. Perform hyperparameter tuning to optimize the model's performance.

## Ⅲ. Description of the Dataset

1. **Description**

This dataset from the U.S. Small Business Administration (SBA) comprises 899,164 records and 27 columns, totaling 179.43MB, detailing various aspects of SBA loans.

These are columns from the dataset:

1. LoanNr\_ChkDgt: Identifier Primary key
2. Name: Borrower name
3. City: Borrower city
4. State: Borrower state
5. Zip: Borrower zip code
6. Bank: Bank name
7. BankState: Bank state
8. NAICS: North American industry classification system code
9. ApprovalDate: Date SBA commitment issued
10. ApprovalFY: Fiscal year of commitment
11. Term: Loan term in months
12. NoEmp: Number of business employees
13. NewExist: 1 = Existing business, 2 = New business
14. CreateJob: Number of jobs created
15. RetainedJob: Number of jobs retained
16. FranchiseCode: Franchise code, (00000 or 00001) = No franchise
17. UrbanRural: 1=Urban, 2=rural, 0=undefined
18. RevLineCr: Revolving line of credit: Y = Yes, N = No
19. LowDoc: LowDoc Loan Program: Y=Yes, N=N
20. ChgOffDate: The date when a loan is declared to be in default
21. DisbursementDate: Disbursement date
22. DisbursementGross: Amount disbursed
23. BalanceGross: Gross amount outstanding
24. MIS\_Status: Loan status charged off = CHGOFF, Paid in full =PIF
25. ChgOffPrinGr: Charged-off amount
26. GrAppv: Gross amount of loan approved by bank
27. SBA\_Appv: SBA’s guaranteed amount of approved loan

It is worth noting that there are some specialized features in the loans within this dataset. For example, RevLineCr(Revolving line of credit) refers to a type of credit arrangement that allows the borrower to access funds up to a pre-approved limit, repay, and borrow again. LowDoc(LowDoc Loan Program) stands for "Low Documentation Loan Program," which is designed to simplify and expedite the loan application process for smaller loan amounts, making it easier for small businesses to access financing. NAICS refers "North American Industry Classification System" code. This is an industry code system used to classify North American businesses. Table 1 [[1]](#endnote-1)showed description of the first two digits of NACIS.

|  |  |
| --- | --- |
| Sector | Description |
| 11 | Agriculture, forestry, fishing and hunting |
| 21 | Mining, quarrying, and oil and gas extraction |
| 22 | Utilities |
| 23 | Construction |
| 31-33 | Manufacturing |
| 42 | Wholesale trade |
| 44-45 | Retail trade |
| 48-49 | Transportation and warehousing |
| 51 | Retail trade |
| 52 | Finance and insurance |
| 53 | Real estate and rental and leasing |
| 54 | Professional, scientific, and technical services |
| 55 | Management of companies and enterprises |
| 56 | Administrative and support and waste management and remediation services |
| 61 | Educational services |
| 62 | Health care and social assistance |
| 71 | Arts, entertainment, and recreation |
| 72 | Accommodation and food services |
| 81 | Other services (except public administration) 92 Public administration |

Table 1

Finally, MIS\_Status is selected as the target feature as an indicator of loan performance.

1. **Data Source**

This is the Dataset source link:

<https://www.kaggle.com/datasets/mirbektoktogaraev/should-this-loan-be-approved-or-denied>

## Ⅳ. Results and Analysis

1. **Data Cleaning and Feature Engineering**
2. **Removing Duplicates and Handling Missing Values**

Our project begins with data cleaning, which involves addressing missing values and data type conversion. Since there’s no duplicate in our dataset, we directly proceed to the missing value part. Figure1 shows the information about our missing values.

图片包含 图形用户界面

描述已自动生成

Figure 1

For the label “Mis\_Status”, we drop rows with those missing values. For features "Name", "City", "State", "Bank", "BankState", "NewExist", "RevLineCr", "LowDoc", "DisbursementDate", the number of missing values is relatively small in comparison to the overall sample size; hence, we have decided to remove rows containing these missing values to maintain data integrity.

A particular focus is on the "ChgOffDate" column, which indicates the date when a loan is declared in default. Upon comparing the number of missing entries in "ChgOffDate" with the count of loans marked as "PIF" (Paid in Full), we noticed a close correlation. This similarity led us to hypothesize that the missing values in "ChgOffDate" are primarily due to loans that were fully repaid, as such loans would not have a default date. Based on this insight, considering the direct relationship between "ChgOffDate" missing values and fully repaid loans, we decided to remove this column from our dataset to streamline our analysis and focus on more impactful variables. After the previous steps mentioned, all the missing values in our dataset have been properly handled.

1. **Drop Features That Are Apparently Irrelevant to Our Prediction**

In the next stage, we implemented some feature selection. We dropped features that are apparently irrelevant to our prediction, such as "LoanNr\_ChkDgt" and "Name".

We scrutinized columns such as “ChgOffPrinGr” for charged-off amounts and “BalanceGross” for gross amounts outstanding. We recognized that these features materialize only after a company is marked as default (loan status charged off), indicating a potential for data leakage if included in our model. To maintain the integrity of our predictive analysis and prevent overfitting, we chose to exclude these variables.

Furthermore, to ensure our model’s predictive power is not biased by temporal factors and can reliably forecast future defaults, we eliminated time-related attributes like "ApprovalDate," "ApprovalFY," and "DisbursementDate." This aligns with our goal of creating a time-agnostic model that evaluates the likelihood of a company paying in full or charging off, free from the influence of specific time periods or fiscal years.

To effectively handle the geographic information for the company, we've decided to streamline the data by focusing on the 'State' attribute. Due to the high number of unique values, we will remove 'City' and 'Zip' from our analysis to prevent issues with excessive categorical data representation. Similarly, we will exclude the 'Bank' feature and retain only the 'BankState'.

Figure 2 shows the dataset info after we removed some unnecessary columns.

表格

描述已自动生成

Figure 2

1. **Convert Data Types and Create New Columns**

In the subsequent stage of data preprocessing, we converted specific attributes to string data types for enhanced interpretability; this included the "NewExist" and "UrbanRural" features.

Because there's some kind of overlap between "DisbursementGross" and "GrAppv". We decide to keep the "DisbursementGross". Considering that the amount of the loan payment should be a float instead of a string, we used code to remove the “$” sign from the original string type and convert it into a float type.

For the "NAICS" codes, we established a new column titled "Industry," which categorizes businesses according to the two-digit North American Industry Classification System codes (Table 1) by assigning them descriptive industry sector names. (Code showed in Figure 3)

Regarding "FranchiseCode," we introduced a binary column named "IsFranchise" to distill the original information into a simplified form, facilitating more straightforward analysis in subsequent stages. (Code showed in Figure 4)

To assess the relationship between the borrower's location and the bank's location, we will introduce a new column that indicates whether the borrower and the bank are in the same state[[2]](#endnote-2). (Code showed in Figure 5)

图形用户界面, 文本

描述已自动生成

Figure 3

图形用户界面, 文本, 应用程序

描述已自动生成

Figure 4

图形用户界面, 文本, 应用程序

描述已自动生成

Figure 5

The next step is to filter out irregular entries in 'RevLineCr' and 'LowDoc'. We removed all entries in the 'RevLineCr' column that did not correspond to the standard 'Y' (Yes) or 'N' (No) values, as evidenced by the before-and-after counts of value frequencies. A similar process was carried out for the 'LowDoc' column, again keeping only the 'Y' and 'N' entries and discarding any others (code shows in Figure 6 and Figure 7). This approach ensures that both columns now contain consistent binary indicators, which are likely more relevant for any subsequent analysis, such as machine learning modeling or data visualization, while reducing noise and potential errors arising from non-standard entries.

图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成

Figure 6

图形用户界面, 文本, 应用程序

描述已自动生成

Figure 7

After data cleaning and feature engineering, we process and fill missing values, delete duplicates, standardize the data and filter noise to a certain extent, and obtain a relatively high-quality data set to improve the accuracy of the model. The processed data set now has 610187 samples and 16 features, details is shown in the Figure 8.

表格

描述已自动生成

Figure 8

1. **Exploratory Data Analysis (EDA)**

Through preliminary data processing, we got a general understanding of the data set and asked some questions:

* Does the higher the SBA guarantee mean the company has better credit and is more likely to repay the loan in full?
* Will more disbursement amount create more jobs?
* How are companies applying for SBA loans geographically distributed, and what does that mean?
* How does the term of the loan affect whether the company pays it back?
* Will whether the location of the borrowing company and the bank be consistent will affect whether the company will repay the loan in the end? How to explain?
* Does the greater a company's ability to create new jobs, the greater the likelihood of repayment?
* Is there any correlation between whether a company has a franchise business code and whether it will repay the loan? If so, what advice can this correlation provide banks when approving loans?

To further explore the connections between data and answer these questions, we first do Univariate Analysis for numeric features and discrete features respectively. Then visualize their distribution through code to discover the structure of the data more clearly.

1. **Univariate Analysis for Numeric Features**

图表

描述已自动生成

Figure 9

The histogram (Figure9) shows that the distribution of the number of employees is severely shifted to the left. Severe bias can affect the prediction accuracy of trained models. Although the data is seriously biased, its distribution is consistent with reality. Some large companies with many employees will also apply for loans through the SBA, although the number of these large companies is small. However, considering that our project mainly focuses on loans to small companies, we define companies with more than 60 employees as not meeting the definition of small companies and remove them from the data set. By plotting the histogram again (Figure 10), we can see that the distribution of the number of employees has become normal.

figure11 shows the number of jobs created is primarily distributed very unevenly, concentrated around 20. This reflects reality, as small companies have limited capacity to create many new positions. To address the issue of uneven data distribution, we set the upper limit for new jobs created by small companies at 20. After removing data that does not meet this criterion and redrawing, the histogram (figure12) is obtained. Although the data distribution is still heavily skewed to the left, it has been somewhat improved. Further adjustments will be made prior to training specific models, based on the characteristics of the models.

图表, 直方图

描述已自动生成

Figure 10

表格

描述已自动生成

figure 11

图表

描述已自动生成

figure 12

Same as "JobCreated", we do the same thing with “RetainedJo” and remove data that has retained jobs more than 20.

For “DisbursementGross”, we find it from figure13 that the distribution of it is not balance, but it is consistent with the actual situation. Although SBA is committed to assisting the development of small businesses, there are still a small number of large companies that apply for loans through its policies and assistance. Since this project focuses on loans to small businesses, we removed data samples with loan amounts greater than 1,000,000 because we assume that small businesses will not be approved for loans with too high amounts due to their small size. After that, we replotted the histogram as figure14 shows.

图表

描述已自动生成

figure 13

图表, 直方图

描述已自动生成

figure 14

Same situation for “SBA\_Appv”, so we do the same thing to it and didn’t find more special information about it.

1. **Univariate Analysis for Discrete Features**

We have several discrete features in dataset, such as “State”, “BankState”, “NewExist”, “SameState”, “UrbanRural”, “RevLineCr”, “LowDoc”, “Industry” and “IsFranchise”. Because not all features present useful information, so we will only elaborate on some of the parts that need explanation.

First, we check the distribution of our label “Mis\_Status”. The visualization shows in figure15 From this figure, we can find that the total amount of Paid-In-Full data was much larger than the amount of Charged-Off data, making the distribution of data very imbalanced. This imbalance could potentially lead to a model that demonstrates strong predictive accuracy for companies that Paid-In-Full. However, its performance in identifying defaulting ("Charged-Off") companies is notably poorer. To address this issue, we will seek ways to solve this problem in the subsequent model training part.

图表, 条形图

描述已自动生成

figure 15

Then we do the same thing for other discrete features but did not find any further information that merited deeper investigation.

1. **Bivariate Analysis**

Based on the box plot (figure16), we make the hypothesis longer-term loans have a higher likelihood of being paid in full, whereas shorter-term loans might be more susceptible to being charged off.

As we can see most of the companies do not create any job (figure 12), we try to categorize this feature into "WhetherCreateJob" to find more information.

From the results of figure17, it can be summarized that in most cases, regardless of whether a job was created or not, most of the records are in the "PIF" status. Moreover, instances of job creation are relatively rare, which may suggest that job creation is not common within this dataset. Additionally, for those cases where jobs were indeed created, there is a slightly higher proportion in the "CHGOFF" status compared to those where no job was created, which answered one of our questions before EDA.

图表, 箱线图

描述已自动生成

figure 16

When we check the relationship between “MIS\_Status” and “DisbursementGross”, we found that loans with higher disbursement amounts are more likely to be paid in full, whereas loans with lower disbursement amounts are more likely to be charged off (figure18).

图表, 条形图

描述已自动生成

figure 17

图表, 箱线图

描述已自动生成

figure 18

After checking the relationship between “MIS\_Status” and “DisbursementGross”, we found that companies with higher SBA’s guaranteed amount of approved loan are likely to pay in full (figure19). This result can be interpreted to mean that the higher the SBA guarantee amount, the lower the probability that the company will default on the loan. The SBA guarantee amount appears to influence whether a company repays its loan. This also validates a hypothesis we had at the start of our Exploratory Data Analysis (EDA). We find that the SBA's guarantee for small business loans has a positive effect, better aiding in the development of businesses and influencing the likelihood of a company repaying its loan.

图表, 箱线图

描述已自动生成

figure 19

Through visual analysis of the relationship between "SameState" and "Mis\_Status", we discovered that if the bank and the borrower company are in different states, there tends to be a higher proportion of unpaid loans (figure 20). This effectively answers another question we had at the beginning of our EDA. Based on this result, we have reason to believe that SBA-approved loans where the bank and borrower company are in the same location are more conducive to reducing the risk of bad debt.

图表, 条形图, 箱线图

描述已自动生成

figure 20

Next, we analyzed the relationship between "IsFranchise" and "Mis\_Status" (figure 21). We discovered that borrower companies without a franchise code have a higher proportion of non-repayment. To some extent, the franchise code signifies a company's reputation and background. This suggests that banks, before approving loans, can focus on examining whether the borrowing business has a franchise code and review the company's historical operational data through that code to mitigate the risk of the loan.

图表, 箱线图

描述已自动生成

figure 21

After completing the bivariate analysis between the discrete features and the target feature, we also explored the relationships between some numerical variables but did not find any strong correlations or additional information that warranted further exploration.

1. **Modeling**
2. **Data Preprocessing**

We initiate the data preprocessing stage by applying one-hot encoding to transform categorical features into a numerical format for the convenience of subsequent operations. This task was accomplished utilizing the OneHotEncoder from scikit-learn's preprocessing module. The encoder was configured to generate a binary column for each category within the designated categorical fields. These binary vectors denote the presence (with a "1") or absence (with a "0") of each category, effectively translating qualitative values into a structured quantative format. Employing OneHotEncoder not only facilitates the interpretative capabilities of our models but also strengthens the integrity and precision of our predictions (figure 22).

In the next step of our data preparation process, we have delineated the features and target for our predictive modeling. We generate our feature matrix by excluding the 'MIS\_Status' column, which we have designated as the target variable, and set 'MIS\_Status' column as the target array. This separation between features and target is a preparatory step essential for supervised learning algorithms.



figure 22

Subsequently, the dataset is partitioned into training and testing sets to facilitate model validation and to ensure that our model can generalize well to unseen data. The train\_test\_split function is employed for this division, with 80% of the data reserved for training and 20% for testing. The random\_state parameter is set to 42 to guarantee reproducibility of the split.

The final step in our preprocessing involved the numeric transformation of the target variable's categorical labels. By invoking LabelEncoder, we transformed these labels into a numeric array, avoiding the risk of computational discrepancies during the model training phase. This transformation was applied to both the training and testing targets, resulting in the generation of y\_train and y\_test.

With the successful execution of these preprocessing strategies, our dataset is now well-prepared for the application of Random Forests model.

1. **Random Forest**

Transitioning into the model training phase, we've implemented the RandomForestClassifier. Because of the imbalanced nature of our dataset, we've set the class\_weight parameter to 'balanced'. This approach ensures the classifier assigns a greater importance to minority classes during training, ensuring a more proportionate influence of each class on the model's learning process.

Following the configuration of our model, we proceeded to fit the model with our training data (X\_train, y\_train). Upon training completion, we utilized the trained model to predict outcomes on the test set (X\_test). The model's predictive power was quantified by calculating the accuracy score, which in this case is an impressive 93.55%. Such a high level of accuracy suggests that our model is adept at making correct predictions most of the time.

In the next step, we inspect the classification report(figure 23), which provides more detailed information of the model's performance through metrics such as precision, recall, and F1-score. The report reveals good precision and recall for both classes, with weighted averages exceeding 90%. We gave particular attention to the recall score of class 0. This metric is especially significant as it reflects the model's capability in accurately identifying all potential “Loan Status Charged Off” cases. As shown in the classification report, our model's recall for “Loan Status Charged Off” cases (class 0) is lower at 0.75 compared to 0.98 for 'Paid in Full' cases. This is likely due to the imbalance in our dataset. To address this, we will undertake hyperparameter tuning to improve the detection of 'charge off' instances.

A screenshot of a computer

Description automatically generated

figure 23

1. **Hyperparameter Tuning for Random Forest**

Hyperparameter tuning involves iterating through a set of hyperparameters to find the combination that enhances our model’s performance, to improve the recall for the “Loan Status Charged Off” case. Due to limited time and computation resourses, we only take “n\_estimator”, the number of trees in the forest, and “max\_depth”, maximum number of levels in each decision tree, into consideration.

After conducting the grid search, the optimal hyperparameters identified were 'max\_depth' of 20 and 'n\_estimators' of 100. When the RandomForestClassifier was retrained with these parameters, it yielded a notable improvement in recall for “Loan Status Charged Off”, up to 0.83 from the previous 0.75, without significant decrease in the overall accuracy (figure 24).

A screenshot of a computer

Description automatically generated

figure 24

This improved recall for “Loan Status Charged Off” suggests that the model is now more powerful to identify potential defaults. While there is a minor trade-off in overall accuracy, dropping to approximately 90.17%, the gain in recall for class 0 represents a targeted improvement aligned with our purpose of dectecting companies in default.

1. **Feature Selection**

Before applying models like SVM and Logistic Regression, we implemented feature selection to filter the most influential features in out dataset. Based on results in EDA, features such as "Term," "DisbursementGross," "SBA's guaranteed amount of approval loan," "SameState," "UrbanRural," "RevLineCr," and "LowDoc" were identified as having strong relationships with our target variable, "MIS\_Status". To ensure our models focus on the most pertinent predictors, we utilized the SelectKBest method with the chi2 statistic, which gauges the degree of association between each feature and the target variable.

Employing the chi2 function allows us to compute the Chi-squared statistic and retain the top k features that exhibit the strongest relationships. To determine the optimal number of features to keep ('k'), we carried out cross validation across a range of 'k' values. For SVM, we found that the model achieved the best performance in terms of recall when 'k' was set to 15 (figure 25). Similarly, for Logistic Regression, the ideal number of features was equal to 10 (figure 26). This approach to feature selection ensures that the models concentrate on the variables most predictive of our target outcome and help to avoid overfitting.

图形用户界面, 文本, 应用程序

描述已自动生成

figure 25

图形用户界面, 文本, 应用程序

描述已自动生成

figure 26

1. **Feature Normalization**

Before we proceed with implementing Support Vector Machine (SVM) and Logistic Regression models, it's essential to address the uneven distribution of our feature values. Given that our data exhibits significant variance in feature scales and that these two algorithms are particularly sensitive to such disparities, normalization becomes a critical preprocessing step.

For this problem, we utilize the StandardScaler from the scikit-learn library. This scaler adjusts our features so that they assume a distribution with a mean value of 0 and a standard deviation of 1. By standardizing the range of our features, we ensure that no single feature with a broad range can dominate the model's behavior.

1. **SVM**

Following the normalization of features to address scale sensitivity, we fitted a Support Vector Machine (SVM) model with a linear kernel due to its efficiency in handling larger datasets compared with SVC.

Once the model was trained, we proceeded to evaluate its performance on the test set (figure 27). The accuracy score is approximately 0.839. However, the classification report reveals a precision of 0.76 and a very low recall of 0.23 for the minority class (class 0), indicating that the model is not effectively identifying most of the 'charge off' cases. In contrast, for the majority class (class 1), the model shows a high precision of 0.84 and a recall of 0.98, suggesting it's proficient at identifying 'Paid in Full' cases.

A number of numbers on a white background

Description automatically generated

figure 27

The initial results showed that the model was better at predicting the “Paid in Full” cases, which is a common issue when dealing with unbalanced classes. In our dataset, 'Paid in Full' cases were over four times more common than 'charge offs', leading the model to favor the majority class.

To tackle this imbalance, we set the class\_weight to 'balanced', which helped the model pay more attention to the “Loan Status Charged Off”. After this adjustment, although the overall accuracy (figure 28) dropped slightly, the model's ability to identify “Charge Off” cases improved significantly, which is crucial for identifying accurately.

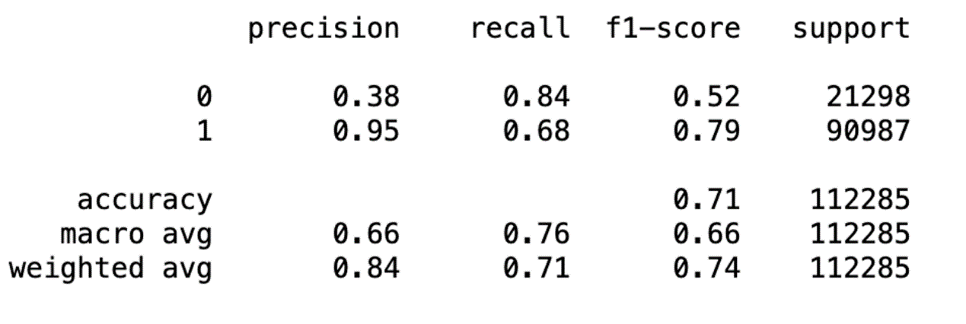


figure 28

1. **Hyperparameter Tuning For SVM**

After applying GridSearchCV to tune the hyperparameter 'C' for a LinearSVC model, the results showed an unexpected decrease in recall for class 0 (the 'Charged Off' loans), dropping from 0.84 to 0.41. While the accuracy and recall for class 1 saw some improvement, this outcome does not align with our objective of improving the model's ability to identify potential defaults, as a high recall for class 0 is critical in the context of loan default prediction.

With limited time and computational resources, and considering the extensive training time required for SVM models, we are currently unable to make immediate adjustments to the hyperparameter tuning process. We will further investigate this part in the future.

A screenshot of a computer

Description automatically generated

figure 29

1. **Logistic Regression**

Due to the widespread application of logistic regression in classification problems, especially its remarkable effect in binary classification, we have chosen the logistic regression model to train and predict on our dataset. Initially, we establish our model using the LogisticRegression class from sklearn.linear\_model. The resulting model exhibits an accuracy of 84%, which seemingly indicates good performance. However, we observed a significant imbalance in the prediction of positive and negative classes.

A number of numbers on a white background

Description automatically generated

figure 30

The occurrence of this situation is largely due to the presence of class imbalance in our dataset, which was also observed during our earlier EDA (figure 15). To address this issue, we set the class\_weight attribute to 'balanced', which will automatically adjust the weight of each category according to the frequency of the labels in the data. This means that less frequently occurring categories (Charged-off) will be assigned higher weights, while more frequently occurring categories (Paid-In-Full) will be assigned lower weights, with the expectation of countering the impact of class imbalance during the training process. Subsequently, the model is trained (fit) on the training set X\_train and y\_train, and the trained model is used to predict on the test set X\_test, resulting in y\_predicted\_lg.

A number of numbers in a row

Description automatically generated with medium confidence

figure 31

After training model, we conducted an accuracy analysis on the model. Using accuracy\_score and classification\_report, we found that the model's accuracy decreased from the previous 84% to 68%, but the recall has improved, indicating that the adjustment of the target weights was effective.

However, bias still exists, despite having used class\_weight='balanced' to mitigate the problem. The model tends to predict samples as class 0, leading to the omission of some true cases of class 1. Therefore, we are considering further optimization of the model.

1. **Hyperparameter Tuning for Logistic Regression**

We optimized the model considering imbalance, with the objective of maximizing recall. We employed grid search for cross-validation, adjusting hyperparameters.

The parameter grid, param\_grid\_lg, included several candidate values for the regularization strength C and two types of penalty parameters, using 5-fold cross-validation (cv=5), and with recall (scoring='recall') as the performance evaluation metric.

The results revealed: the best parameters identified by grid search were C: 100 and penalty: 'l2'. The classification\_report indicated that the model’s accuracy on the test dataset was approximately 68%, which is a general performance indicator, suggesting that about 68% of predictions were correct. For class 0, the precision was 35%, recall was 83%, and the F1 score was 50%. This suggests that the model tended towards a high recall rate at the expense of precision, which could lead to many samples belonging to class 1 being incorrectly predicted as class 0. For class 1, the precision was 94%, recall was 65%, and the F1 score was 77%. Despite the high precision, the lower recall implies that the model missed some actual samples of class 1.

A number of numbers in a row

Description automatically generated with medium confidence

figure 32

These results suggest that in dealing with imbalanced data, the model may overemphasize predicting samples as the majority class (class 0). Since the model aims to maximize recall, this situation is not surprising, but it could lead to an increased number of false positives, thereby lowering precision.

To further improve the balance of the model, we chose to expand the range of the parameter grid, as there may still be better parameter combinations that have not yet been explored. Interestingly, after expanding the parameter grid range, the performance of the obtained model was consistent with that before the adjustment.

A number of numbers on a white background

Description automatically generated

figure 33

Through analysis, we find that this could be due to several reasons.

1. The search range is still not broad enough, but considering the current computing power of our computer, continuing to expand the range will increase the model's computational load, which the computer cannot handle. This will require further exploration when conditions allow in the future.
2. The model may have reached a performance bottleneck: For the given dataset and feature set, the model may have achieved the best performance it can. In this case, adjusting the hyperparameters may not lead to an improvement in performance.
3. Perhaps the parameters we have chosen are already good, so hyperparameter tuning may not bring significant improvement.

Which specific reason it is still requires further learning and exploration on our part.

After Hyperparameter Tuning, we draw a Confusion Matrix (figure 29). For confusion matrix, it provides actual numbers for true negatives (TN), false positives (FP), true positives (TP), and false negatives (FN). The matrix has many true positives (indicating a high number of correct predictions for the positive class) and true negatives but also a considerable number of false negatives, explaining why the recall for class 1 is not very high.

图表, 树状图

描述已自动生成

figure 29

# Ⅴ. Conclusion

In this project, we investigated the effectiveness of the U.S. Small Business Administration (SBA) Loan Guarantee Program, aiming to predict the repayment status of loans—either "Paid In Full" or "Charged-Off"—using machine learning techniques. Our study began with a comprehensive data cleaning process, removing duplicates and handling missing values to enhance data integrity. We engaged in exploratory data analysis to understand the factors influencing loan outcomes and utilized feature engineering to improve predictive performance.

Several machine learning models, including Random Forest, Logistic Regression, and Support Vector Machines, were employed to address this binary classification problem. The models were evaluated based on accuracy, precision, recall, and F1 score, and refined through hyperparameter tuning. The Random Forest model showed promising results with an accuracy of 0.94 and a recall of 0.83 in identifying the companies that will not repay the loan.

However, our model still has many shortcomings. The significant skewness of the label in our dataset has resulted in unbalanced predictions from the model. To address this issue, we have adjusted the hyperparameters through grid search and set the class weight to 'balanced' in hopes of assigning appropriate weights to the two values of the target feature. However, overall, the efforts we have made so far are not enough; further analysis is required. For example, our model could readjust thresholds to pay more attention to minority class in the future, improve feature selection again, and expand the hyperparameter grid to further tune the model.

Also, our findings suggest that the SBA guarantee significantly influences loan repayments, with a higher guarantee amount reducing the likelihood of default. Additionally, loans involving borrowers and banks in the same state showed a lower default rate, highlighting the importance of geographical factors in loan performance.

For bank, before approving loans, can focus on examining whether the borrowing business has a franchise code and review the company's historical operational data through that code to mitigate the risk of the loan.

References

1. [North American Industry Classification System (NAICS) U.S. Census Bureau](https://www.census.gov/naics/?58967?yearbck=2022) [↑](#endnote-ref-1)
2. [SBA Loan Approval Analysis (kaggle.com)](https://www.kaggle.com/code/kevinm6720/sba-loan-approval-analysis) [↑](#endnote-ref-2)