A Comparison Of Binary Classification Methods for Loan Eligibility Prediction

Zhuli Wang
210244908
Dr. Aisha Abuelmaatti
MSc Computing and Information Systems

***Abstract*—With the economy and financial environment being sounded and more focusing on individual activities, there are increasing loan applications which are time-consuming and high cost to tackle manually. This research is an endeavour on applying 4 classification models of machine learning to help improve the automation of the loan process by predicting loan eligibility. The data being studied is from the Small business Administration of America which is a loan-guaranteed government department. The data contains 899164 rows of the loan application. Logistic Regression, Random Forest, Support Vector Machine and Extreme Gradient Booster are applied to the SBA loan data which is being processed by the EDA method. The best performance algorithm is XGboot with 95.7% for Accuracy, 91.3% for Precision, 88.7% for Sensitivity and 93.2% for AUC.**

***Index Terms*—Loan eligibility, Binary classification, Logistic Regression, Extreme Gradient Booster, Random Forest, Support Vector Machine**

I. Introduction

Big data has a huge impact on the financial industry, it changes many businesses' processes largely including the lending process. Nowadays, automation streamlining become the latest financial industry key point. The software application which can achieve the demand in the area of lending and credit assessment for financial customers has became prevailed. To increase efficiency, decision speed and customer experience, the finance institutions keep focussing on business automation [1]. With the participation of automation, the key data that affect the whole lending process are collected and stored by those financial institutions.

With the increasing amount of loan application data being collected, machine learning, a big data technology could be utilized to help improve the efficiency and accuracy of loan eligibility. Loan eligibility has two statuses, approval and disapproval which means it can be effectively converted to binary classification problems in machine learning. A stable and reliable machine learning model would help those institutions accurately predict the decision of a loan application with the automation system. This can help the financial institution achieve total business automation and free the labour force from numerous and tedious auditing tasks then let them focus on the more important job that machine learning and automation could not participate in.

The dataset which is dedicated to research is from Kaggle.com [2], authorized by US SBA (Small Business Administration). US SBA was established in 1953 with the aim of proceeding and helping small businesses in the US credit market [3].

SBA assists small businesses with a loan guarantee program that commits a portion of the loan and repays that portion if the loan defaults. Virtually, SBA is an official guarantor that gives banks confidence to approve the loan application [4].

This dissertation is aiming to create and compare several machine learning models which potentially are the optimal solution to the SBA loan eligibility prediction problem. Find out the optimal parameter for each algorithm then compare which algorithm is the better solution for the data we are learning.

II. Previous Research

Many studies have been conducted on loan eligibility prediction or similar binary classification issue [5] [6] [7] [8]. Since many algorithms can be applied to such binary classification problems, some studies focused on one or two models, and some compared several algorithms. In previous research, not every algorithm can achieve a score well under the same performance metrics. However, some algorithms have achieved average good performance in previous research which will be the focus of my research.

Reference [8] studied loan approval prediction by applying the logistic regression model to the consumer loan data from a Portuguese finance institution. The study concluded that the percentage of correctly classified cases of the logistic regression model was 89.93%, sensitivity is 0.94% and specificity is 99.55% (conduct from confusion matrix). It mentioned that the cutting point of Default = 1 was 0.5, and the prediction results were rounded ($p < 0.5$ round to 0; $p >= 0.5$ round to 1).

Reference [7] in the study compared three algorithms, linear discriminant analysis, logistic regression and random forest based on the data from the Swedish Board of Student Finance (CSN). It introduced several performance metrics to compare each algorithm's advantages and disadvantages. First, the study used a multiclass approach, in accuracy, three algorithms had a good result in the range of 96-97%. Taking into account the unbalanced data, other indicators should be used. In terms of sensitivity, LDA has the worst performance, while LR just lags behind it, and although RF has better performance, it is still not enough. Comparing AUC, LDA reached 66.46%, LR was 70.93% and RF was 67.5%, which means all the models got nearly 70 chances of rightness. Then the researchers used a binary approach and found the LR got an AUC of 0.946 and RF of 0.9764 which were quite good results. RF performance is great out of two algorithms in sensitivity of 96.95% and specificity of 80.58%.

Reference [6] compared six models, logistic regression, decision tree, random forest, gradient-boosted trees, factorization machine and linear support vector machine. In the conclusion, the random forest presented the highest accuracy for 99.619%, AUC for 99.67%, precision for 99.8%, recall for 99.2% and f-score for 99.5%, and decision tree was merely equal to the random forest. Thus, random forest and decision tree were the most efficient and accurate in predicting binary loan default problems. However, LSVM was a cutting line between these six models which deserves further research.

Reference [13] introduced a powerful learning algorithm extreme gradient boosting (XGboost). Reference [14] compared logistic regression and XGboost, according to the conclusion XGboost had better overall performance than logistic regression in accuracy, sensitivity, specificity, and precision with values 99.94%, 99.88%, 99.99% and 99.99% and the value of logistic regression was 87.88%, 82.43%, 93.13% and 92.05%. which means XGboost might potentially perform better than random forest or decision tree studied by AM Uwais, H Khaleghzadeh [6].

According to those previous research, logistic regression is a good baseline for the comparison of several algorithms because it can better reflect the relationship between variables and predictors and generate a mild good prediction. Beyond this, random forest, support vector machine and XGboost have good performance in previous research. This dissertation will focus on the comparison of these four algorithms' performance and propose that XGboost and Random forest have a better performance than Logistic Regression and Support Vector Machine.

III. Theory

A. Supervised learning

For the loan approval prediction, there is a clear response variable, therefore supervised learning is the appropriate method to be applied. Supervised learning utilizes a labelled dataset to train an algorithm to classify the given data and make a precise prediction. The prediction variable of this study is 'Default' ('MIS_Status' before feature engineering).

B. Binary classification algorithm

1) Support Vector Machine: The support vector machine works by finding an independent line or a hyperplane which can best separate the data into two parts. To determine the optimal line or hyperplane, mathematically a general line is introduced. The closest points to the general line from both the positive and negative sides are called support vectors. Then construct two lines connecting each side's support vectors which parallel with the general line. The problem now converts to finding the maximum margin of the two support vector lines [9].

2) Logistic Regression: Logistic regression is the algorithm which most often being used to demonstrate the function that indicates the independent variables to dependent variables. Logistic regression is different from linear regression as the outcome of logistic regression is binary. The core theory of logic regression is based on probability, and the equation is shown below:

$$E\left(y\right) = \frac{1}{1 + e^{-\left(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_\rho x_\rho\right)}} \tag{1}$$

$E\left(y\right)$ means the probability of $y = 1$ when given sets of specific independent variables. The boundary of the equation is 0 and 1. β_0 is the linear regression coefficient and $\beta_1 \ldots \beta_\rho$ is the regression coefficient of independent variables. In binary classification, the result probably will be transformed to binary values 0 or 1.

3) Random forest: Random forest is an ensemble method used for classification, regression and other tasks requiring decision trees [10]. Random forest is also called bagging, it created several subsets from a dataset and randomly chooses the subsets with replacement. With more collected trained trees, the prediction averaged from those trees will be more accurate.

In random forest, a Gini Impurity is introduced to describe the feature importance. It is measured by the impurity decrement of a feature split at each node. The feature with the largest decrement is the best performing one. This will also help drop the less important features to optimise the dataset.

4) Extreme gradient boosting: Extreme gradient boosting is a tool to control overfitting using a more regularized model formalization, it has a better performance and faster speed, said by Tianqi Chen, the author of XGboost [11].

The sequential ensemble model, known as boosting, consists of several classifiers which train the data one by one to reduce the residual loss of the model [12]. The classifier of each model will adjust the weight for each parameter based on the prediction correctness and convey it to the next classifier to improve the accuracy. The classifier model can be added until the prediction is correct or reach the limitation of the classifier maximum.

C. Model building

The building flow of the loan approval prediction model consists of several steps: 1) data collection; 2) data preprocess; 3) model fitting; 4) hyperparameter tuning; 5) comparison and conclusion.

1) Data pre-processing: Data pre-processing is the basic and urgent step of machine learning. The raw data for machine learning is always incomplete, imbalanced, clumsy and contains irrelevant information. To make the dataset best fit the learning model, pre-processing is necessary and compulsory.

2) Missing value: The missing value is the most common issue for all machine learning projects. They are usually N/A, null and empty cells.

The simplest way to handle missing value is to drop it, this will lead to a decrement in the sample which causes less representativeness of data information and increases bias in the data set and might cause the absence of key value.

Another way to deal with missing value is imputation, this is replacing the missing value with a substituted value. The substituted value is an estimated value base on other available information.

3) Classifying categorical values: The performance of the machine learning model might become worse when the predictor variable has many unique values. Therefore, these variables with many unique values need to be classified into several related groups with no more than 2 unique values.

A function of pandas, get_dummies [17] is a tool that can automatically detect the unique values and split them into different variables and then the values of those variables will come binary.

4) Feature Engineering: The learning model usually does not perform well with the original features from raw data because most original features are not fitting machine learning criteria. Feature engineering can simplify data sets, speed up data learning and enhance model accuracy.

Feature creations indicate the creation of new variables that will be helpful to the model and the removal of some unnecessary variables. Transformation is reformatting features from raw type to another efficient type, is easy to plot data. Feature extraction is identifying useful information of the features from the data set without misunderstanding the original meaning or twisting the relation. [15].

5) Scaling: Normalization, also known as min-max normalization, is the process of scaling all values in a specific range between zero and one. The modification of the values has no impact on the distribution of features, however, it will exacerbate the effect of the outlier because of lower standard deviation so the outlier should be dealt with before normalization. The formula [16] of normalization is:

$$X' = \frac{X - X_{min}}{X_{max} - X_{min}} \tag{2}$$

X_{max}, X_{min} are the maximum and minimum values of the feature respectively.

Standardization, also known as z-score normalization, is the process of scaling each feature separately by its standard deviation. The formula [16] for standardization is:

$$X' = \frac{X - \ \mu}{\sigma} \tag{3}$$

μ is the mean of feature values and σ is the standard deviation of feature values. When the dataset is not following the Gaussian distribution, normalization is a better choice than standardization, and on the contrary, standardization is better than normalization.

6) Imbalanced data, oversampling and undersampling: For machine learning algorithms, a proper divide of data set is important for the learning process because the bias of data would mislead the learning process. Oversampling and undersampling are two techniques to solve the data imbalance.

Oversampling is mainly duplicating samples from the minority class while undersampling is deleting samples from the majority class.

Random oversampling is the process of randomly choosing samples from the minority classes and copying them to supplement the train set. Therefore, random oversampling might cause overfitting when an instance could be randomly selected several times. It will also lead to an increment in computational cost when the skewed degree is far too high.

Random undersampling is the process of randomly choosing samples from the majority classes and deleting them. This can make the decision boundary between majority and minority instances harder to learn and cause a loss in classification performance. It is important to understand random undersampling might delete valuable information and cannot be detected or preserved.

7) Data splitting: The data set should be separated into a train set and a test set. The training set is the base for estimating parameters, comparing models and so on. The test set is used to estimate a final, neutral assessment of the model's prediction [18].

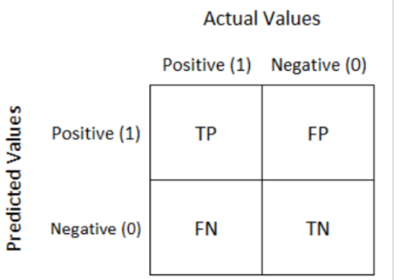
To prevent underestimation and overestimation, the data should be split randomly especially when data is imbalanced. Normally a percentage of twenty to twenty-five could be assigned to the test set and a percentage of seventy-five to eighty assigned to the train set.

D. Hyperparameters tuning

There is a commonly used tool called GridSearchCV [19]. Given a set of the grid of parameters, it can go through those parameters and search for the best combination. It is a cross-validation method: the model and the parameters are required to be input and then the best parameter values are extracted and the prediction is made.

E. Performance metric

The confusion matrix, Fig.1, is an intuitive metric to calculate the correctness and accuracy of a model. It is a combination of prediction and actual values. It is very efficient for calculating the value of Recall, Accuracy, Precision and AUC.



Predicted Values	Actual Values	Positive (1)	Negative (0)
	Positive (1)	TP	FP
	Negative (0)	FN	TN

Fig. 1. Confusion Matrix [20]

Accuracy is the calculation result of all the true predictions among all predictions. It will effectively evaluate a model when the target class is nearly balanced. The equation is shown in (4).

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \tag{4}$$

Precision is the calculation result of true positive predictions among all the positive predictions. It will effectively evaluate a model when false positive predictions are concerned. The equation is shown in (5).

$$Precision = \frac{TP}{TP + FP} \tag{5}$$

Recall is the calculation result of true positive predictions among true positive predictions and false negative predictions. It will effectively evaluate a model when false negative predictions are concerned. The equation is shown in (6).

$$Recall = \frac{TP}{TP + FN} \tag{6}$$

F1 score is the harmonic mean of precision and recall, reaching maximum when precision is equal to recall. The extreme variable will be punished in the F1 score. The equation is shown in (7).

$$F1 = 2 \times \frac{Precision \times Recall}{Precison + Recall} \tag{7}$$

The Receiver Operator Characteristic (ROC), Fig.2, is a probability curve plotted by the FPR as the x-axis against TPR as the y-axis. The equation is shown in (8).

$$TPR = \frac{TP}{TP + FN}; FPR = \frac{FP}{TN + FP} \tag{8}$$

AUC shown in Fig.2, measures the entire area under the ROC curve. It shows the probability of a random positive prediction having a higher rank than a random negative prediction. AUC score zero when a model's predictions are all wrong, and scores one when a model's predictions are all correct.

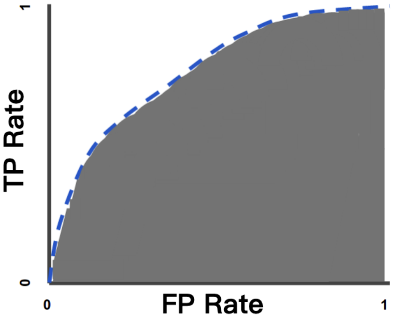


Fig. 2. Area Under ROC [21]

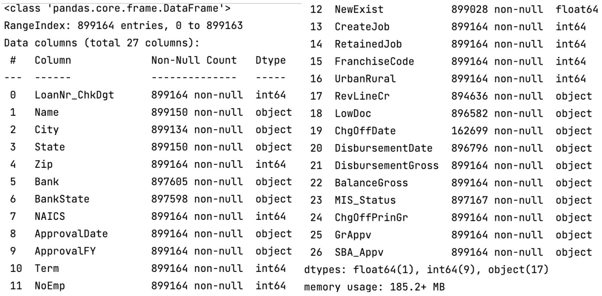
IV. Method

A. Data overview

The data consists of 899164 rows of loan applications and 27 columns for variables. The meaning of each variable is presented in Fig. 3. And the data overview is shown in Fig. 4.

Variable Name	Description
LoanNr_ChkDgt	Identifier Primary key
Name Borrower	name
City Borrower	city
State	Borrower state
Zip Borrower	zip code
Bank Bank	name
BankState	Bank state
NAICS	North American industry classification system code
ApprovalDate	Date SBA commitment issued
ApprovalFY	Fiscal year of commitment
Term Loan	term in months
NoEmp	Number of business employees
NewExist	1 = Existing business, 2 = New business
CreateJob	Number of jobs created
RetainedJob	Number of jobs retained
FranchiseCode	Franchise code, (00000 or 00001) = No franchise
UrbanRural	1 = Urban, 2 = rural, 0 = undefined
RevLineCr	Revolving line of credit: Y = Yes, N = No
LowDoc	LowDoc Loan Program: Y = Yes, N = No
ChgOffDate	The date when a loan is declared to be in default
DisbursementDate	Disbursement date
DisbursementGross	Amount disbursed
BalanceGross	Gross amount outstanding
MIS_Status	Loan status charged off = CHGOFF, Paid in full =PIF
ChgOffPrinGr	Charged-off amount
GrAppv	Gross amount of loan approved by bank
SBA_Appv	SBA's guaranteed amount of approved loan

Fig. 3. data meaning reference [2]



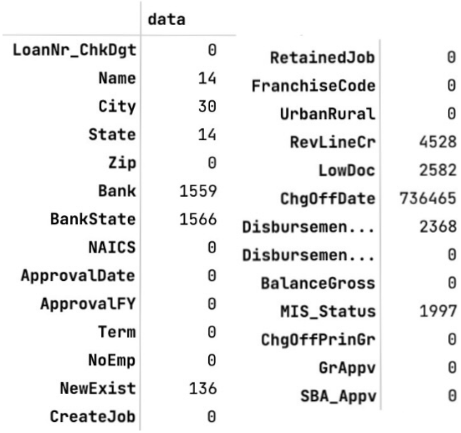
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 899164 entries, 0 to 899163
Data columns (total 27 columns):
 #   Column             Non-Null Count   Dtype
---  ------             --------------   -----
 0   LoanNr_ChkDgt      899164 non-null  int64
 1   Name               899150 non-null  object
 2   City               899134 non-null  object
 3   State              899150 non-null  object
 4   Zip                899164 non-null  int64
 5   Bank               897605 non-null  object
 6   BankState          897598 non-null  object
 7   NAICS              899164 non-null  int64
 8   ApprovalDate       899164 non-null  object
 9   ApprovalFY         899164 non-null  object
 10  Term               899164 non-null  int64
 11  NoEmp              899164 non-null  int64
 12  NewExist           899028 non-null  float64
 13  CreateJob          899164 non-null  int64
 14  RetainedJob        899164 non-null  int64
 15  FranchiseCode      899164 non-null  int64
 16  UrbanRural         899164 non-null  int64
 17  RevLineCr          894636 non-null  object
 18  LowDoc             896582 non-null  object
 19  ChgOffDate         162699 non-null  object
 20  DisbursementDate   896796 non-null  object
 21  DisbursementGross  899164 non-null  object
 22  BalanceGross       899164 non-null  object
 23  MIS_Status         897167 non-null  object
 24  ChgOffPrinGr       899164 non-null  object
 25  GrAppv             899164 non-null  object
 26  SBA_Appv           899164 non-null  object
dtypes: float64(1), int64(9), object(17)
memory usage: 185.2+ MB
```

Fig. 4. Raw Data Overview

B. Pre-processing

1) Null value: Null values of ChgOffDate are not a concern because the prediction is focusing on whether the loan was charged off, not when did charge off happened.

The null values of other rows shown in Fig.5 need to be removed entirely rather than imputing them for reasons. First, the amount of other null values is quite small compared to the whole dataset. Second, imputation is far more difficult for the data spanned long period and the exact circumstance of some variables is hard to learn. For example, the variable NewExist is potentially an important feature, but it is hard for us to assume whether it is a new business or an existing business.

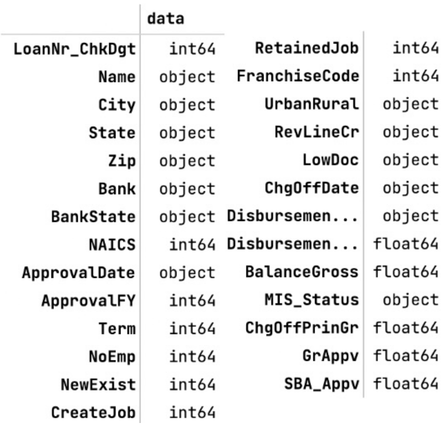


	data
LoanNr_ChkDgt	0
Name	14
City	30
State	14
Zip	0
Bank	1559
BankState	1566
NAICS	0
ApprovalDate	0
ApprovalFY	0
Term	0
NoEmp	0
NewExist	136
CreateJob	0
RetainedJob	0
FranchiseCode	0
UrbanRural	0
RevLineCr	4528
LowDoc	2582
ChgOffDate	736465
Disbursemen...	2368
Disbursemen...	0
BalanceGross	0
MIS_Status	1997
ChgOffPrinGr	0
GrAppv	0
SBA_Appv	0

Fig. 5. Null value counts

2) *Feature formatting:* The currency fields, DisbursementGross, BalanceGross, ChgOffPrinGr, GrAppv, and SBA_Appv are recognized as object rather than float. ApprovalFY should be integer rather than object. Therefore, the data type needs to be changed.

Also, some other variables' data types are changed, the result was shown in Fig.6.



	data
LoanNr_ChkDgt	int64
Name	object
City	object
State	object
Zip	object
Bank	object
BankState	object
NAICS	int64
ApprovalDate	object
ApprovalFY	int64
Term	int64
NoEmp	int64
NewExist	int64
CreateJob	int64
RetainedJob	int64
FranchiseCode	int64
UrbanRural	object
RevLineCr	object
LowDoc	object
ChgOffDate	object
Disbursemen...	object
Disbursemen...	float64
BalanceGross	float64
MIS_Status	object
ChgOffPrinGr	float64
GrAppv	float64
SBA_Appv	float64

Fig. 6. Data types

3) *Feature extraction:* Extract the first two digits of the variable NAICS to create a new column Industry base on the NAICS number reference.

Convert variable FranchiseCode, NewExist to binary flag fields IsFranchise and NewBusiness.

Convert variables RevLineCr and LowDoc to binary flag fields.

Set the target variables to Default based on MIS_Status. 1 stands for charge off and 0 stands for paid in full.

Set a feature AppvDisbursed, which indicates the difference between variable DisbursmentGross and GrAppv.

4) *Feature creation:* Create variable DaysToDisbursment which indicates the time from loan approval to funds disbursement. This may potentially reveal the importance of capital circulation in loan prediction.

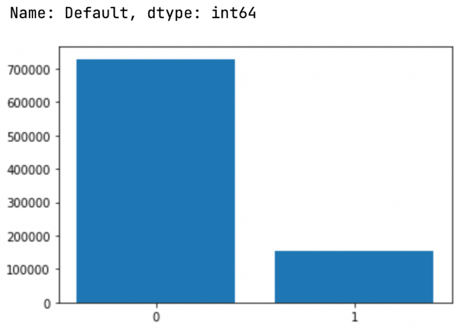
Create variable StateSame which indicates whether the location state of the loan-providing bank is the same as the location of the loan applicant. This may potentially reveal the applicant's location preference of a bank in their loan approval decision-making.

Create variable SBA_ApprPct which indicates the SBA guaranteed loan amount percentage of the total loan application amount. This may potentially reveal the future default risk of a loan application which is concerned by both SBA and the bank.

5) *Drop column:* LoanNr_ChkDgt, NameBank, ChgOffDate, NAICS, NewExist, FranchiseCode, ApprovalDate, DisbursementDate, SBA_Appv and MIS_Status are dropped.

6) *Imbalanced data:* There are 358,558 cases paid fully and 98,382 cases are charged-off. The paid fully cases are 78.5% of the total cases and charged-off cases are 21.5% of the total cases. The imbalance result of the data set with a ratio of approximately 1:3.7. The distribution of the target variable has shown in Fig.7

Random undersampler function [22] was adopted to undersampling the data set.



```
Name: Default, dtype: int64
```

Fig. 7. Target variable distribution

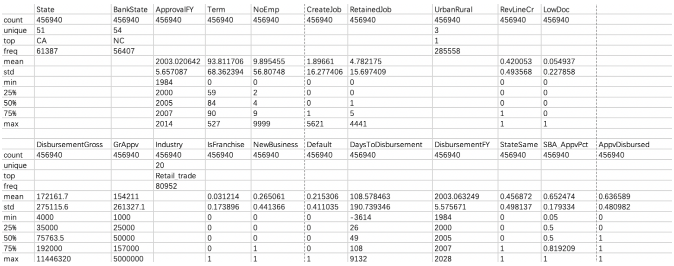
7) *Data scaling and splitting:* Sklearn.preprocessing.StandardScaler [23] is adopted to standardize the data set.

The portion train set is set to 0.75 and test set is 0.25 by sklearn.model_selection.train_test_split function [24].

C. Data analysis

As shown in Fig.8, there are some notable information.

- The average month of Term is 93.8 which means the loan terms are usually long.
- 42% of loans are revolving line of credit and only 5.5% of loans are LowDoc program loan.
- 78.5% of loans in the data set were paid fully.
- 45.7% of loans were serviced by banks in the same state.
- The average percentage of SBA loan guaranteed amount was 65.2%



	State	BankState	ApprovalFY	Term	NoEmp	CreateJob	RetainedJob	UrbanRural	RevLineCr	LowDoc
count	456940	456940	456940	456940	456940	456940	456940	456940	456940	456940
unique	51	54						3		
top	CA	NC						1		
freq	61387	56407						285558		
mean			2003.020642	93.811706	9.895455	1.89661	4.782175		0.420053	0.054937
std			5.657087	68.362394	56.80748	16.277406	15.697409		0.493568	0.227858
min			1984	0	0	0	0		0	0
25%			2000	59	2	0	0		0	0
50%			2005	84	4	0	1		0	0
75%			2007	90	9	1	5		1	0
max			2014	527	9999	5621	4441		1	1

	DisbursementGross	GrAppv	Industry	IsFranchise	NewBusiness	Default	DaysToDisbursement	DisbursementFY	StateSame	SBA_AppvPct	AppvDisbursed
count	456940	456940	456940	456940	456940	456940	456940	456940	456940	456940	456940
unique			20								
top			Retail_trade								
freq			80952								
mean	172161.7	154211		0.031214	0.265061	0.215306	108.578463	2003.063249	0.456872	0.652474	0.636589
std	275115.6	261327.1		0.173896	0.441366	0.411035	190.739346	5.575671	0.498137	0.179334	0.480982
min	4000	1000		0	0	0	-3614	1984	0	0.05	0
25%	35000	25000		0	0	0	26	2000	0	0.5	0
50%	75763.5	50000		0	0	0	49	2005	0	0.5	1
75%	192000	157000		0	1	0	108	2007	1	0.819209	1
max	11446320	5000000		1	1	1	9132	2028	1	1	1

Fig. 8. Data description

D. Correlation analysis

From the heat map Fig.9, it can easily learn that GrApprv and DisbursementGros have a positive relation; RevLineCr have a negative relation with both SBA_AppvPct and AppvDisbursed; SBA_AppvPct have a negative relation with both ApprovalFY and DisbursementFY.

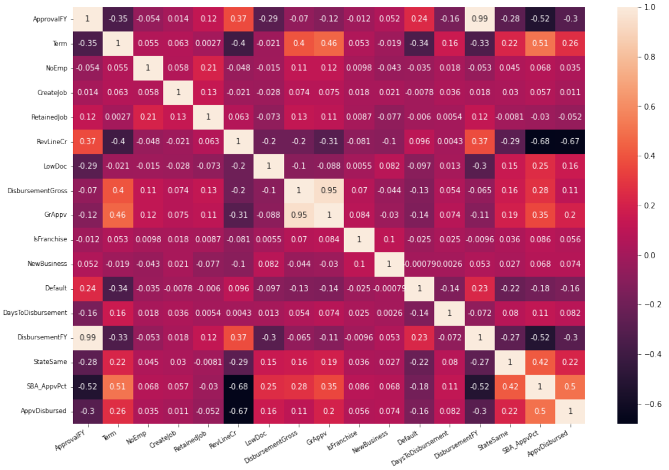


Fig. 9. Correlation Heat Map

E. Fitting models

This research adopts classification models from the scikit-learn machine learning python package and the XGboost package from dmlc. The fitting models are sklearn.linear_model.LogisticRegression, sklearn.ensemble.RandomForestClassifier, and sklearn.svm.SVC and XGBClassifier.

The key parameter for logistic regression is 'C' and 'penalty', for Random Forest is 'n_estimators', 'max_depth' and 'min_samples_split', for SVM is 'C', 'gamma' and 'kernal', for XGboost is 'booster', 'eta', 'gamma' and 'max_depth'.

V. Result

A. Logistic Regression

According to the result (Fig. 10, Fig. 11, TABLE I), the best performance for logistic regression model is the default parameter, 'C' = 1 and 'penalty' = l2, with no random undersampling. The best parameter for undersampling model is the same as the default. The accuracy of the default setting is 86.7% while the accuracy of the undersampling model is 81.8%.

B. Random Forest

According to the result (Fig. 12, Fig. 13, TABLE II), the best performance for random forest model is the default parameter, 'n_estimators' = 100, 'max_depth' = none and 'min_samples_split'=2, with no random undersampling. The best parameter for undersampling model is 'n_estimators' = 60, 'max_depth' = 13 and 'min_samples_split'=50. The accuracy of the default setting is 95% while the accuracy of the undersampling model is 93%.

C. XGboost

According to the result (Fig. 14, Fig. 15, TABLE III), the best performance for extreme gradient booster is the default parameter, 'booster' = gbtree, 'eta' = 0.3, 'gamma'=0 and 'max_depth'=6, with no random undersampling. The best parameter for undersampling model is 'booster' = gbtree, 'eta' = 0.2, 'gamma'=0 and 'max_depth'=7. The accuracy of the default setting is 95.7% while the accuracy of the undersampling model is 94.5%.

D. Support Vector Machine

According to the result (Fig. 16, Fig. 17, TABLE IV), the best performance for extreme gradient booster is the default parameter, 'C': 1, 'gamma': 'scale' and 'kernal'='rbf', with no random undersampling. The best parameter for undersampling model is 'C': 10, 'gamma': 0.01 and 'kernal'='rbf'. The accuracy of the default setting is 90.2% while the accuracy of the undersampling model is 86.6%.

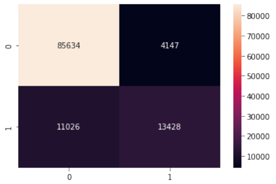


Fig. 10. Confusion Matrix for LR

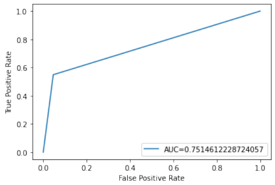


Fig. 11. ROC Curve for LR

TABLE I
LOGISTIC REGRESSION

	Precision	Recall	F-score	Support
0	0.886	0.954	0.919	89781
1	0.764	0.549	0.639	24454
accuracy			0.867	114235
macro avg	0.825	0.751	0.779	114235
weighted avg	0.860	0.867	0.859	114235

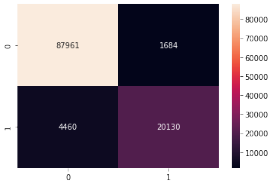


Fig. 12. Confusion Matrix for Random Forest

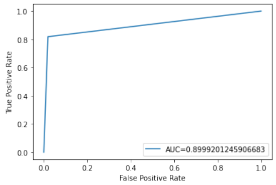


Fig. 13. ROC Curve for Random Forest

TABLE II
RANDOM FOREST

	Precision	Recall	F-score	Support
0	0.95	0.98	0.97	89645
1	0.92	0.82	0.87	24590
accuracy			0.95	114235
macro avg	0.94	0.90	0.92	114235
weighted avg	0.95	0.95	0.95	114235

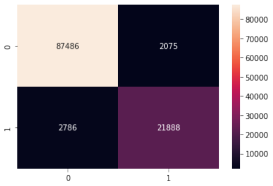


Fig. 14. Confusion Matrix for XGboost

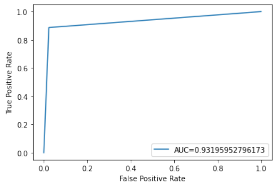


Fig. 15. ROC Curve for XGB

TABLE III
XGBOOST

	Precision	Recall	F-score	Support
0	0.969	0.977	0.973	89561
1	0.913	0.887	0.900	24674
accuracy			0.957	114235
macro avg	0.941	0.932	0.937	114235
weighted avg	0.957	0.957	0.957	114235

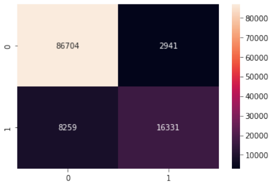


Fig. 16. Confusion Matrix for SVM

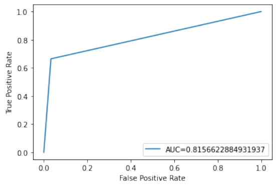


Fig. 17. ROC Curve for SVM

TABLE IV
SUPPORT VECTOR MACHINE

	Precision	Recall	F-score	Support
0	0.913	0.967	0.939	89645
1	0.847	0.664	0.745	24590
accuracy			0.902	114235
macro avg	0.880	0.816	0.842	114235
weighted avg	0.899	0.902	0.897	114235

VI. DISCUSSION AND CONCLUSION

A. Conclusion

The performance of four classification models as shown in TABLE V. For SBA loan applicant data, XGboost and RF have a better performance than LR and SVM. This result validates my proposal which based on conclusions from research [6] [7] [8] [13] [14] that XGboost and Random Forest have better performance than LR and SVM.

Comparing different circumstances of data preprocessing and parameter tuning, as shown in TABLE VI, the performance of each algorithm has a varying degree of decrement. The most severe algorithm is Logistic Regression and then SVM. XGboost and Random Forest do not change severely. All the parameter tuning makes the performance of algorithms become worse.

According to the feature importance of XGboost, as shown in Fig.18, it can learn that the term of the loan is a critical impact on whether it will go default or not, since in reality the longer a loan's term the higher possibility of default. The term of the loan will also decide the amount of the fund lending which is another critical impact on loan default so as the DisbursementGross. DaysToDisbursement also suggests importance in prediction because the longer the loan takes to disburse, the more likely the loan goes to default. ApprovalFY suggests that if the lender could not get the loan as soon as possible, that means there must be a potential risk the bank to consider carefully.

Although the official format and requirements of a loan application are varied in different institutions, the basic principle for loan default control and eligibility approval is generally the same. The correlation of key features from the SBA data set will also have the same impact on other business-based loan predictions. Therefore, XGboost and Random forest can also perform well on other loan application data not only on SBA loan application data.

TABLE V
PERFORMANCE OF FOUR CLASSIFICATION MODELS

	XGboost	RF	LR	SVM
Accuracy	95.7%	95.0%	86.7%	90.2%
Precision	96.9%	95.0%	88.6%	91.3%
Recall	97.7%	98.0%	95.4%	96.7%
F-score	97.3%	97.0%	91.9%	93.9%
AUC	93.2%	90.0%	75.1%	81.6%

TABLE VI
COMPARISON OF FOUR CLASSIFICATION MODELS ON ACCURACY

	Before Undersampling	After Undersampling	Paramteter Tuning After Undersampling
XGboost	95.7%	94.5%	94.6%
RF	95.0%	93.0%	92.0%
SVM	90.2%	86.6%	84.2%
LR	86.7%	81.8%	81.8%

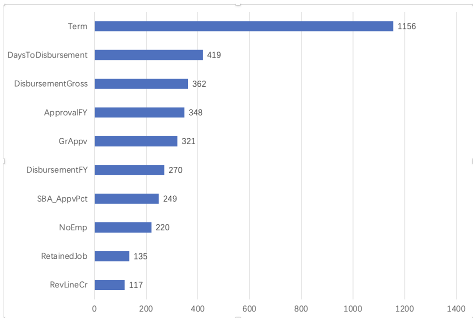


Fig. 18. Feature Importance of XGB

B. *Discussion*

After the training process of each classification model, the results are presented and revealed many issues. The main issue for this study is the imbalanced data did not affect the prediction accuracy and precision as anticipation which should improve the prediction performance for each algorithm.

For these four classification algorithms, the random undersampling data set all perform badly compared to the imbalanced data set. The deviation between the random undersampling data set and imbalanced data set for logistic regression is 5.65%, random forest is 2.1%, XGboost is 1.25% and Support vector machine is 3.99%. Random forest and XGboost have a mild impact from the undersampling while logistic regression and SVM have a greater impact. This is because the mechanism of these two algorithms might have negative effects from the defect of undersampling.

As mentioned in the theory part, random undersampling has two direct defects, cannot draw a fairly panoramic of data set and might drop valid samples with valuable information from the majority class. It is possible that logistic regression and SVM can learn a relatively explicit data set with an imbalance which is distinguishable better than a random undersampling data set. In SVM, an imbalanced data set will draw a midway learning boundary between the negative and positive support vector, its orientation will be almost the same as the ideal hyperplane but quite far from it which is not classified best. With random undersampling, the learning boundary might intersect with the ideal hyperplane which means a bias will exist in the prediction. Furthermore, according to the imbalance distribution of the target variable, the majority class is paid fully which means the sample from this class is very important for the prediction, then trimming the majority class will make the classification become inaccuracy.

The reason for XGboost's great performance is that it combines the predictions of multiple base learners to generate an overall prediction for each sample. Also, the sequential model helps reduce the loss generated by the previous model in each iteration. Likely, Random forest also performs well because it consists of several classifiers which overfitted random samples to generate predictions and an overall prediction will be voted from the predictions of each classifier. These ensemble methods can effectively process structured data sets with a complex relationship between features and the target variable. This is also the reason more and more research adopt Random Forest and Extreme Gradient Booster as the primary choice in deep learning.

Acknowledgment

I would like to express my gratitude first and foremost to my research supervisor Mrs Aisha Abuelmaatti from the Queen Mary University of London. She patiently guides me on how to start my research and points out a possible direction for me. I would like to express my sincerest thanks to all people who gave their time to help me in the production of this dissertation, for their expertise and support. I also want to thank my parents for them keep encouraging me and caring for my health.

References

[1] Doug, P. (2018). How Automation Can Improve Your Loan Origination Process. [online] www.moodysanalytics.com. Available at: https://www.moodysanalytics.com/articles/2018/maximize-efficiency-how-automation-can-improve-your-loan-origination-process.

[2] Mirbek, T. (2020). Should This Loan be Approved or Denied? [online] www.kaggle.com. Available at: https://www.kaggle.com/datasets/mirbektoktogaraev/should-this-loan-be-approved-or-denied.

[3] Wikipedia Contributors (2019a). Small Business Administration. [online] Wikipedia. Available at: https://en.wikipedia.org/wiki/Small_Business_Administration.

[4] Li, M., Mickel, A. and Taylor, S. (2018). 'Should This Loan be Approved or Denied?': A Large Dataset with Class Assignment Guidelines. Journal of Statistics Education, 26(1), pp.55–66. doi:10.1080/10691898.2018.1434342.

[5] Madaan, M., Kumar, A., Keshri, C., Jain, R. and Nagrath, P. (2021). Loan default prediction using decision trees and random forest: A comparative study. IOP Conference Series: Materials Science and Engineering, 1022(1757-899X), p.012042. doi:10.1088/1757-899x/1022/1/012042.

[6] Uwais, A. (2021). Loan Default Prediction Using Spark Machine Learning Algorithms. [online] Available at: http://ceur-ws.org/Vol-3105/paper30.pdf.

[7] Oskarsson, E. (2021). Machine Learning Model for Predicting the Repayment Rate of Loan Takers. [online] DIVA. Available at: https://www.diva-portal.org/smash/record.jsf?pid=diva2%3A1563211&dswid=717.

[8] Costa e Silva, E., Lopes, I.C., Correia, A. and Faria, S. (2020). A logistic regression model for consumer default risk. Journal of Applied Statistics, 47(13-15), pp.2879–2894. doi:10.1080/02664763.2020.1759030.

[9] Wikipedia Contributors (2019b). Support-vector machine. [online] Wikipedia. Available at: https://en.wikipedia.org/wiki/Support-vector_machine.

[10] Wikipedia Contributors (2019a). Random forest. [online] Wikipedia. Available at: https://en.wikipedia.org/wiki/Random_forest.

[11] Shah, I. and Pachanekar, R. (2020). Introduction to XGboost in Python. [online] Quantitative Finance & Algo Trading Blog by QuantInsti. Available at: https://blog.quantinsti.com/xgboost-python/.

[12] Sumanta (2021). XGboost by Heart. [online] AlmaBetter. Available at: https://medium.com/almabetter/xgboost-by-heart-b494a471845e [Accessed 5 Aug. 2022].

[13] Odegua, R. (2020). Predicting Bank Loan Default with Extreme Gradient Boosting. arXiv:2002.02011 [cs, q-fin]. [online] Available at: https://arxiv.org/abs/2002.02011v1 [Accessed 5 Aug. 2022].

[14] Dwidarma, R., Permai, S.D. and Harefa, J. (2021). Comparison of Logistic Regression and XGboost for Predicting Potential Debtors. [online] IEEE Xplore. doi:10.1109/AiDAS53897.2021.9574350.

[15] Patel, H. (2021). What is Feature Engineering — Importance, Tools and Techniques for Machine Learning. [online] Medium. Available at: https://towardsdatascience.com/what-is-feature-engineering-importance-tools-and-techniques-for-machine-learning-2080b0269f10.

[16] Bhandari, A. (2020). Feature Scaling — Standardization Vs Normalization. [online] Analytics Vidhya. Available at: https://www.analyticsvidhya.com/blog/2020/04/feature-scaling-machine-learning-normalization-standardization/.

[17] pandas.pydata.org. (2022). pandas.get_dummies — pandas 1.2.4 documentation. [online] Available at: https://pandas.pydata.org/docs/reference/api/pandas.get_dummies.html.

[18] Kuhn, M. and Johnson, K. (2021). Feature engineering and selection: a practical approach for predictive models. [online] Boca Raton: Chapman & Hall/Crc, pp.3.3 Data Splitting. Available at: http://www.feat.engineering/data-splitting.html.

[19] Scikit-learn.org. (2019). sklearn.model_selection.GridSearchCV — scikit-learn 0.22 documentation. [online] Available at: https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html.

[20] Agrawal, S.K. (2021). Evaluation Metrics For Classification Model — Classification Model Metrics. [online] Analytics Vidhya. Available at: https://www.analyticsvidhya.com/blog/2021/07/metrics-to-evaluate-your-classification-model-to-take-the-right-decisions/.

[21] Google Developers. (2022). Classification: ROC Curve and AUC — Machine Learning. [online] Available at: https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc?hl=zh_cn [Accessed 5 Aug. 2022].

[22] Imbalanced-learn.org. (2022). RandomUnderSampler — Version 0.9.0. [online] Available at: https://imbalanced-learn.org/stable/references/generated/imblearn.under_sampling.RandomUnderSampler.html.

[23] Scikit-learn.org. (2019b). sklearn.preprocessing.StandardScaler — scikit-learn 0.21.2 documentation. [online] Available at: https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html.

[24] Scikit-learn.org. (2018). sklearn.model_selection.train_test_split — scikit-learn 0.20.3 documentation. [online] Available at: https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.train_test_split.html.