

Loan Analysis Predicting Defaulters

Mudit Manish Agarwal¹, Harshal Mahendra Shirke², Vivek Prafullbhai Vadhiya³,
Manya Gidwani⁴

^{1,2,3}Student, Department of Information Technology, Shah & Anchor Kutchhi Engineering College, Mumbai, India

⁴ Professor, Department of Information Technology, Shah & Anchor Kutchhi Engineering College, Mumbai, India

Abstract - Due to the advancements in the domain of Artificial Intelligence and Data Science, its utilization is becoming more common in every possible domain. Nowadays, the majority of the industries make use of AI and its applications in some or the other way. Taking the advantage of the field of Data Science results in creating effective and modern applications, products irrespective of the domain. One of the industries where the application of AI and Data Science is proving to be effective is the Finance Industry commonly known as the Banking Sector. Banks face severe losses due to the loan defaults made by the client and hence to overcome this problem, there lies a need to create a credit risk scoring model which can analyze and predict the loan defaults. Hence, with the help of Machine Learning, we aim to create a Loan Default Analysis model which could predict the loan defaults and integrate the model into a web application for the user for easy usability.

Key Words: Loan Default, Machine Learning, data mining, prediction, web application.

1. INTRODUCTION

Due to the Covid-19 Pandemic, there is a huge loss of capital caused to the banking sector, financial institutions, small scale finance companies, etc. Nearly a year and a half, everything has been shut down thereby leaving people with no source of income. Due to this reason, there has been a significant increase in the loan defaults made by the client. Now, in this new normal, there is a need especially for the banks to strengthen their loan sanctioning system. Since banks may face huge losses due to the defaults made by their clients which increases the rejection rate of the loan applicants. This affects the bank's overall reputation and also at the same time, due to the rejection of new loan applicants it causes huge financial loss to the bank since, the most common type of unsecured loans are debt consolidation, credit-card loans, student loans, and personal loans [1].

The Loan Analysis Predicting Defaulters (LAPD) is an attempt at creating a better credit risk scoring model which can correctly identify which applicant will be a defaulter in the future. This is done by analyzing the historic data and identifying the patterns. Such a model would minimize credit risk and prevent the clients who are capable of repayment

from getting rejected. Various classification algorithms such as Logistic Regression, Decision Tree, Random Forest, etc. have been applied to build various models and compare them to find out the best accuracy [4]. The source of the dataset is Kaggle which is a highly imbalanced dataset. Data Balancing techniques such as SMOTE, SMOTE ENN have been implemented to balance the dataset. Important features such as loan_amount, term, home_ownership, issue_date_year, etc have been extracted on which the target attribute depends for prediction. Once the model creation and comparison are done, the model which gives better accuracy is integrated into the website for the end-user

2. BUSINESS PROBLEM

Banks may suffer significant losses as a result of customer defaults, which raises the rejection rate of loan applications. To address this issue, we require a more accurate credit risk assessment algorithm that can predict which applicants will default in the future. This will be accomplished by analyzing previous data and discovering trends. This methodology would reduce credit risk and prevent clients who are capable of repayment from being turned down [3].

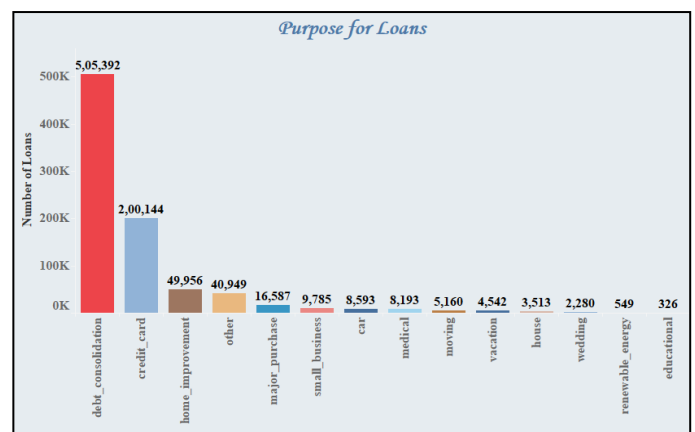


Fig. 2.1 Purpose of loan

3. DATA DESCRIPTION

The Data Set was taken from Kaggle The dataset includes entire loan data for all loans granted between 2007 and 2015, including current loan status (Current, Late, Fully Paid,

etc.) and most recent payment information. There are 8,55,969 items in our dataset, with 73 attributes including the target variable. Furthermore, the dataset is extremely imbalanced, with 46467 entries of failed loans. This dataset contains a variety of attributes, including category, numeric, and date data. The number of defaulters increased significantly between 2012 and 2014, with the LIBOR Scandal, Hurricane Sandy, and Hostess Files for Bankruptcy being some of the primary factors. The major reason for a borrower's request for a loan is the loan purpose. Debt consolidation is the most common reason for taking out a loan, followed by credit card debt.

Some important features from the dataset

- loan_amnt - Amount of money requested by the borrower.
- int_rate - The interest rate on the loan.
- grade - Loan grade with categories A, B, C, D, E, F, G.
- annual_inc - Borrowers annual income.
- purpose - The primary purpose of borrowing.
- installments – Monthly amount payments for the opted loan.
- term – duration of the loan until it's paid off
- to – the ratio of your gross monthly income that goes to paying your monthly debt payments [6].

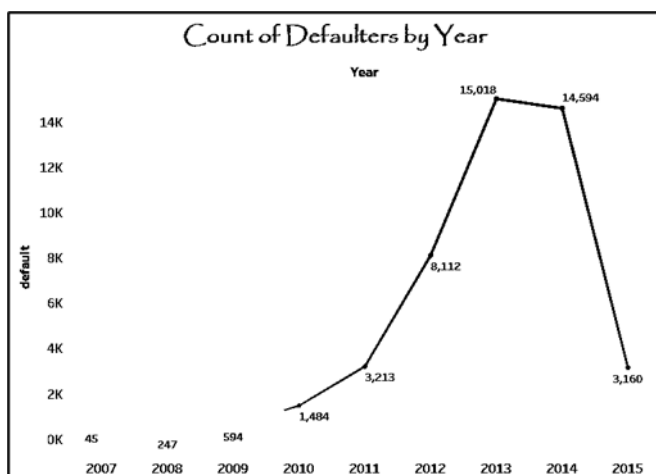


Fig. 3.1 Understanding the dataset

4. DATA PRE-PROCESSING

The data were inspected, cleaned, and prepared as follows before data mining model analysis: Changing the names of the variables — We changed the names of the variables to suit our needs and domain expertise. Retrieved the values of variables such as emp length and term, which had values in a string from which we extracted the numeric component.

E.g.: -'10+ years' -> 10;

'2 years' -> 2;

'1 year' -> 1;

'<1 year' -> 0;

We did the same thing for the term with string values.

For example: - 36 months -> 36

60 months -> 60

Values from one variable are replaced with values from another variable. Based on the application type, the income value of the borrower is replaced with the income value of the co-borrower. If the application type is INDIVIDUAL, the borrower's income will be preserved, but if the application type is JOINT, the current value of the borrower's income will be substituted with joint income. while joint income is the sum of the borrower's and co-income. borrower's Dti/ratio inc exp borrower and dti joint/ratio inc exp joint follow the same technique.

4.1 Label Encoders

The Label Encoder is a method for converting category data to numeric variables. We used the Label encoder approach to categorize the aforementioned variables, which are categorical. We have two functions in this technique: fit and transform.

Label Encoder's Steps:

1. Obtaining the one-of-a-kind values
2. Putting them in ascending order is a good idea.
3. Starting with 0,1,2, and so on, mapping the values.
4. All three processes will be completed by the Fit function.
5. The data values in the data frame will be replaced by the transform function.

4.2 Train Test Split

Splitting the data into train and test in the ratio 7:3 where the training part contains 70 % of the data and Testing part contains 30% of the data Train Size – 598978 records with 8 features Test Size – 256991 records with 8 features. The Dataset was highly Imbalanced with variable 0 (Non – Defaulters) = 809502 and 1 (Defaulters) = 46467. In order, the balance this day out we have used two techniques which are Smote and Smote ENN [3].

4.3 Balancing the dataset

4.3.1 SMOTE

Imbalanced classification entails creating prediction models for datasets with a significant class imbalance. When working with unbalanced datasets, the difficulty is that most machine learning approaches will overlook the minority class, resulting in poor performance, even though performance on the minority class is often the most

significant. Oversampling the minority class is one way to deal with unbalanced datasets. Duplicating instances in the minority class is the easiest way, but these examples don't provide any new information to the model [7]. Instead, new instances may be created by synthesizing old ones. The Synthetic Minority Oversampling Technique, or SMOTE for short, is a kind of data augmentation for the minority population. Imbalanced classification entails creating prediction models for datasets with a significant class imbalance. When working with unbalanced datasets, the difficulty is that most machine learning approaches will overlook the minority class, resulting in poor performance, even though performance on the minority class is often the most significant. Oversampling the minority class is one way to deal with unbalanced datasets. Duplicating instances in the minority class is the easiest way, but these examples don't provide any new information to the model. Instead, new instances may be created by synthesizing old ones. The Synthetic Minority Oversampling Technique, or SMOTE for short, is a kind of data augmentation for the minority population.

4.3.2 SMOTE ENN

This method, developed by Batista et al. (2004), combines the ability of SMOTE to generate synthetic examples for minority classes with the ability of ENN to delete some observations from both classes that are identified as having different classes between the observation's class and its K-nearest neighbor majority class. The following is a description of the SMOTE-ENN procedure.

(SMOTE begins) Select data at random from the minority class. Calculate the distance between the randomly generated data and the k closest neighbors. Add the result to the minority class as a synthetic sample by multiplying the difference by a random value between 0 and 1. Repeat steps 2-3 until the required minority class proportion is achieved. (SMOTE comes to an end) (ENN begins) Calculate K to be the number of closest neighbors. If K can't be determined, it'll be 3. Find the observation's K-nearest neighbor from the dataset's other observations, then return the majority class from the K-nearest neighbor. If the observation's class and its K-nearest neighbor's majority class are not the same, the observation and its K-nearest neighbor are removed from the dataset. Repeat steps 2 and 3 until each class has the required proportion of students. (The ENN comes to a close.) [3].

After using the techniques the data was balanced with equal parts using smote and in the ratio 6:4 by using smote ENN.

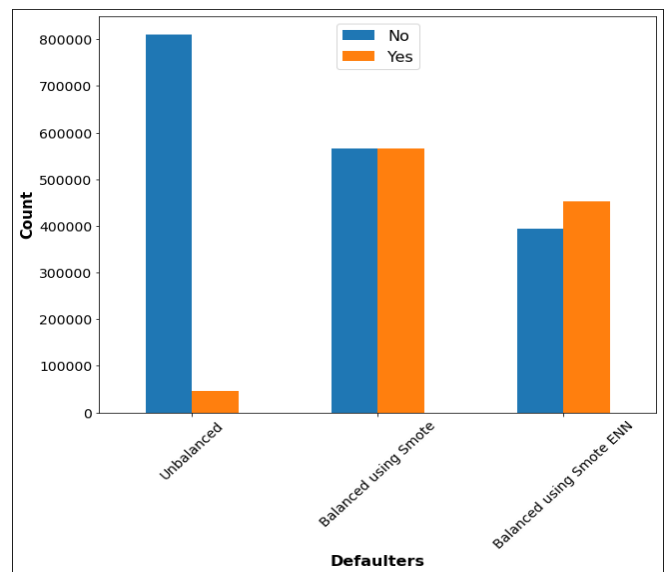


Fig. 4.1 Dataset after balancing

4.4. Algorithms

4.4.1 Logistic Regression

The method of modeling the likelihood of a discrete result given an input variable is known as logistic regression. The most frequent logistic regression models have a binary result, which might be true or false, yes or no, and so forth. Multinomial logistic regression can be used to model situations with more than two discrete outcomes. Logistic regression is a handy analytical tool for determining if a fresh sample fits best into a category in classification tasks. Because components of cyber security, such as threat detection, are classification issues, logistic regression is a valuable analytic tool [2].

4.4.2 Decision Tree

For classification and regression, Decision Trees (DTs) are a non-parametric supervised learning approach. The objective is to learn basic decision rules from data attributes to develop a model that predicts the value of a target variable. A tree is an approximation of a piecewise constant. Decision Trees are a form of supervised machine learning in which the data is continually split according to a parameter (you describe what the input is and what the related output is in the training data). Two entities, decision nodes, and leaves can be used to explain the tree. The decisions or ultimate outcomes are represented by the leaves. And the data is separated at the decision nodes. At first, we consider the entire training set to be the root. Categorical feature values are desired. If the values are continuous, they must be discretized before the model can be built. Records are distributed recursively based on attribute values. As the root of the internal node, we apply statistical approaches to rank characteristics.

4.4.3 Ada Boost

The AdaBoost algorithm, short for Adaptive Boosting, is a Boosting approach used in Machine Learning as an Ensemble Method. The weights are re-allocated to each instance, with larger weights applied to improperly identified instances. This is termed Adaptive Boosting. In supervised learning, boost is used to decrease bias and variation. It is based on the notion of successive learning. Each succeeding student, except the first, is produced from previously grown learners. In other words, weak students are transformed into strong students [6]. With a little modification, the AdaBoost method operates on the same idea as boosting. Adaptive Boosting is an effective ensemble approach for both classification and regression issues. It is most commonly used to solve categorization difficulties. It outperforms all other models in terms of model correctness, which can be verified by following the steps in order. To apply the boost and implement AdaBoost, one can start with decision trees and then move on to random forests. As we progress through the steps outlined above, accuracy improves. The weight-assigning approach used after each iteration distinguishes the AdaBoost algorithm from all other boosting algorithms, which is its strongest feature.

4.4.4 Random Forest

Random forest is a supervised machine learning algorithm that is commonly used to solve classification and regression issues. It creates decision trees from several samples, using the majority vote for classification and the average for regression. One of the most essential characteristics of the Random Forest Algorithm is that it can handle data sets with both continuous and categorical variables, as in regression and classification. For classification difficulties, it produces superior results. Random Forest Algorithm Characteristics-

- 1) It outperforms the decision tree algorithm in terms of accuracy.
- 2) It is a useful tool for dealing with missing data.
- 3) Without hyper-parameter adjustment, it can provide a fair forecast.
- 4) It overcomes the problem of decision tree overfitting.
- 5) At the node's splitting point in every random forest tree, a subset of characteristics is chosen at random.

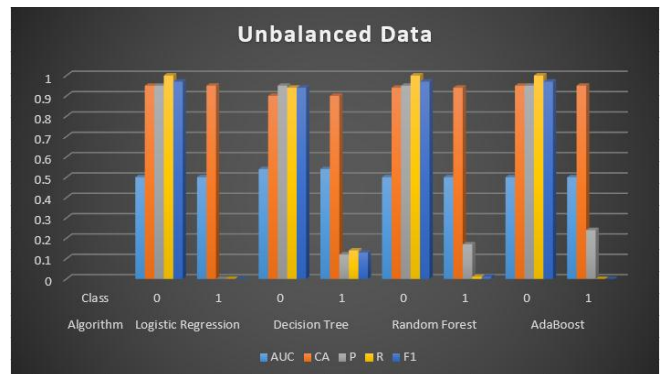


Fig. 4.2 Comparison of Accuracy of Unbalanced Data

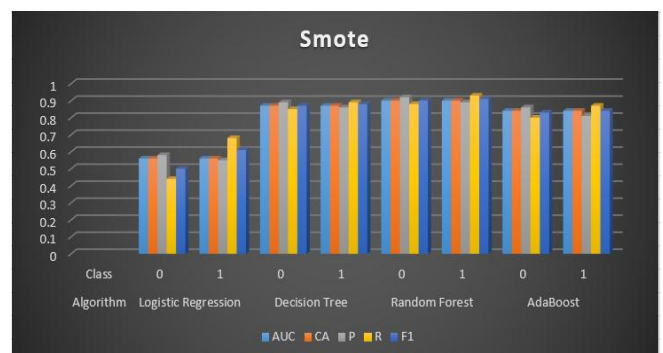


Fig. 4.3 Comparison of Accuracy of Balanced Data using Smote

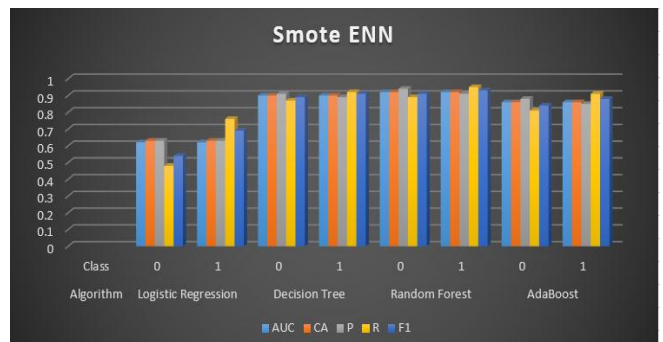


Fig. 4.4 Comparison of Accuracy of Balanced Data using Smote ENN

5.5 Pickle

For serializing and de-serializing a Python object structure, the Python pickle package is utilized. Pickling an object in Python allows it to be stored on a disc [3]. Pickle works by first "serializing" the item before writing it to file. Pickling is a Python function that converts a list, dict, or other Python object into a character stream. The assumption is that this character stream provides all of the data required to recreate the object in another Python function.

Parameters		AUC	CA	P	R	F1
Algorithm	Class					
Logistic Regression	0	0.62	0.63	0.63	0.48	0.54
	1	0.62	0.63	0.63	0.76	0.69
Decision Tree	0	0.9	0.9	0.91	0.87	0.89
	1	0.9	0.9	0.89	0.92	0.91
Random Forest	0	0.92	0.92	0.94	0.89	0.91
	1	0.92	0.92	0.91	0.95	0.93
AdaBoost	0	0.86	0.86	0.88	0.81	0.84
	1	0.86	0.86	0.85	0.91	0.88

Fig. 4.5 Parameters for Evaluation

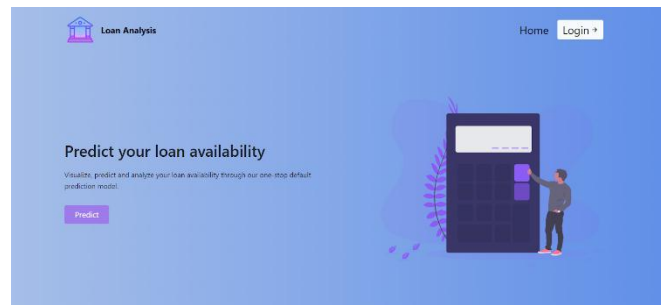


Fig. 5.1 Home Page

After applying these algorithms, we were able to achieve an accuracy of 92 % which was given by Random Forest using Smote ENN method after which we used the pickle library to store the model and flask to integrate it with our webpage.

5. WEB TECHNOLOGIES

For our front-end part, we have used a Python framework called Flask which is used to make a webpage. For our project, we integrated our ML model into a website which is used to get rich output and visualization.

5.1 Flask:

Flask is a Python-based web application framework. The Werkzeug WSGI toolkit and the Jinja2 template engine are the foundations of Flask. Both are Pocco initiatives.

And for the database part, we have used MySQL and also we used the session to authenticate our login system with bank users.

5.2 MYSQL:

MySQL is a quick, easy-to-use relational database management system (RDBMS) that is utilized by many small and large enterprises. MySQL AB, a Swedish business, is responsible for its development, marketing, and support.

For our ML learning model, we used the Pickle python library for integration with the flask webpage also we show the graphs generated from the ML model into our webpage for Easy to understand our users.

Normally Flask is used HTML templates to render the webpage and for CSS we have used bootstrap and tailwind CSS for validation JS function. Also, in the footer, we mention our contributors' details. Total 4 webpages are there on our website which is Home, Login, Register, and Predict.

Screens on our website:

Home Page:

On this page, we make a basic webpage which consists of Navbar, Body, and Footer.

Login Page:

On this page for successful login, users have to give their perfect credentials as per the registration. After successful login bank users can use predict calculator or ML model for their client/customers.

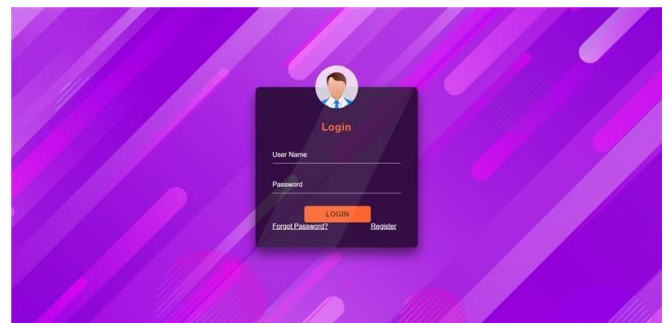


Fig. 5.2 Login Page

Register Page:

On this page, we ask for Basic details of users for registration with the database.

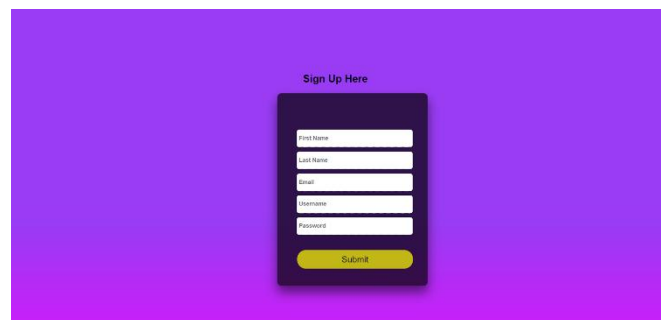


Fig. 5.3 Registration Page

Predict Page:

On this page we made one predicting calculator field for our ML model.

After filling all input fields we give the output as per the ML model results and Accuracy.

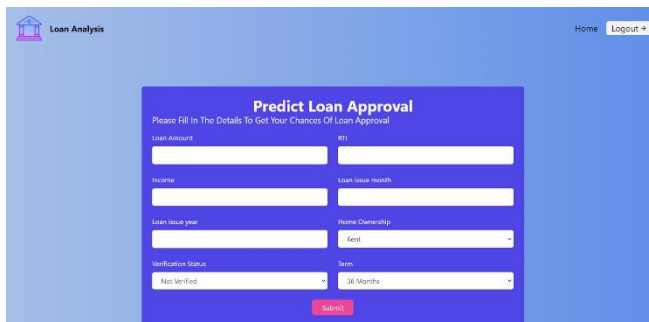


Fig. 5.4 Predict Page

6. FUTURE SCOPE

Since the domain of the project (LAPD) revolves around data science and machine learning, it is possible to add Natural Language Processing (NLP) based chat-bot which would make the project more interactive and would help the user in navigating the website or would recommend or resolve queries regarding the usage of the application. The chat-bot can be integrated either by using python language or by using the Dialogflow platform powered by Google. In our project (LAPD) we have developed a full-fledged website for the end-user to predict the loan default. An alternative to the website can be a mobile application which would prove much handier for the user. Cross-Platform technology frameworks such as Flutter which is also backed by Google can be brought into consideration for developing the application [4]. Apart from this, currently, the dataset which we have used is restricted to a particular organization. If it is possible to collect a wider range of data, it would not only act as a knowledge base for the model but also help the model in understanding and predicting the loan default correctly indirectly resulting in getting better accuracy

7. CONCLUSION

Due to a sudden halt caused by the COVID-19 pandemic, it has affected almost every possible business of industries thus leading to a financial dip. Many financial and banking firms both large scale and small scale are the prime sectors where the pandemic has largely impacted. The loan defaulters rate rose high because of the lockdown and not many were able to pay back. Hence, this invoked the need of developing a system for predicting the loan defaults thereby strengthening the loan sanctioning process.

This LAPD is thus an attempt to design and develop a credit risk scoring model which could analyze and predict the possibility of the loan default and thus reduce the rejection rate of the new loan applicants and capital loss faced by the bank which they would have made by sanctioning the loan to the clients

REFERENCES

- [1] Ahmad Al-qerem, Ghazi Al-Naymat, Mays Alhasan, "Loan Default Prediction Model Improvement through Comprehensive Preprocessing and Features Selection", published.
- [2] P. Maheswari, CH. V. Narayana, "Predictions of Loan Defaulter - A Data Science Perspective", published.
- [3] Lin Zhua, Dafeng Qiu, Daji Ergua, Cai Yinga, Kuiyi Liub, "A study on predicting loan default based on the random forest algorithm", published.
- [4] Hafiz Ilyas Tariq Aziz, Asim Sohail, Uzair Aslam, "Loan Default Prediction Model Using Sample, Explore, Modify, Model, and Assess (SEMMA)", published.
- [5] Mehul Madaan, Aniket Kumar, Chirag Keshri, Rachna Jain and Preeti Nagrath, "Loan default prediction using decision trees and random forest: A comparative study", published.
- [6] Harish Puvvada, Vamsi Mohan Raminedi, "Loan Default Prediction", published.
- [7] Haotian Chen, Ziyuan Chen, Tianyu Xiang, Yang Zhou, "Data Mining on Loan Default Prediction", published.