A Project Report

on

CREDIT RISK ANALYSIS

carried out as part of the course CS1634 Submitted by

Vidhi Garg

179301227

VI-CSE

in partial fulfilment for the award of the degree

of

BACHELOR OF TECHNOLOGY

In

Computer Science & Engineering

MANIPAL UNIVERSITY
JAIPUR

Department of Computer Science & Engineering,
School of Computing and IT,
Manipal University Jaipur,

June, 2020

CERTIFICATE

This is to certify that the project entitled "***Credit Risk Analysis***" is a bona fide work carried out as part of the course ***Minor Project CS-1634***, under my guidance by ***Vidhi Garg***, student of Bachelor Of Technology (B.Tech.) in Computer Science & Engineering (CSE) at the Department of Computer Science & Engineering , Manipal University Jaipur, during the academic semester *VI of year 2019-20.*

Place:

Date: 16 June 2020 Signature of the Instructor (s

DECLARATION

I hereby declare that the project entitled **"*Credit Risk Analysis***" submitted as part of the partial course requirements for the course Minor project (***CS-1653)*** for the award of the degree of Bachelor of Technology in Computer Science & Engineering at Manipal University Jaipur in the semester during academic year 2019-20, has been carried out by me. I declare that the project has not formed the basis for the award of any degree, associate ship, fellowship or any other similar titles elsewhere.

Further, I declare that I will not share, re-submit or publish the code, idea, framework and/or any publication that may arise out of this work for academic or profit purposes without obtaining the prior written consent of the Course Faculty Mentor and Course Instructor.

Signature of the Student:

Place:

Date: 16 June 2020

Abstract:

Credit scoring is important to loan institutions as it helps evaluate the capability of customers to repay their loans. The goal of credit scoring models is to predict the creditworthiness of a customer and determine whether they will be able to meet a given financial obligation or default on it. Such models allow a financial institution to minimize the risk of loss by setting decision rules regarding which customers receive loan and credit card approvals.

Due to the rapid development of machine learning techniques in computer, various classification methods have been proposed for the implementation of machine learning methods for characterizing the repayment behaviour of customers. In our project, we aim to study and use multiple machine learning models to predict the creditworthiness of a lending club's customer using parameters/variables related to their personal status and financial history.

Table of Contents

Sr No.	Content	Page No.
1.	Introduction	6
	1.1 Motivation	6
2.	Literature Review	7
	2.1 Outcome of Literature Review	8
	2.2 Problem Statement	8
	2.3 Research Objectives	9
3.	Methodology and Framework	9
	3.1 System Architecture	9
	3.2 Algorithms used	12
4.	Work Done	15
	4.1 Results and Discussion	15
	4.2 Individual Contribution	18
5.	Conclusion and Future Plan	18
6.	References	19

1. Introduction

Loan lending has played a significant role in the financial world throughout the years. It is quite profitable and beneficial for both the lenders and the borrowers. It, however, carries a great risk in the domain of loan lending which is referred to as Credit risk.

Industry experts and researchers around the world usually assign individuals with scores known as credit scores to measure the risk and their creditworthiness. Throughout the years, machine learning algorithms are being used to calculate and predict credit risk by evaluating an individual's historical data.

ML's ability to consume vast amounts of data to uncover patterns and deliver results that are not humanly possible otherwise is what makes it unique and applicable to so many fields. This predictive power has now sparked a great interest in the credit risk industry. Unlike many other fields, where ML is well-established and used extensively, credit risk modelling has usually taken a cautionary approach to adopting newer ML algorithms.

1.1. Motivation

Loan lending has been an important part of daily lives for organizations and individuals alike. With the ever- increasing competition in the financial world and due to a significant amount of financial constraints, the activity of taking loan has become more or less inevitable.

Though loan lending is quite beneficial for both the lenders and the receivers and is considered an essential part of the financial organization, it does carry some great risks. This type of risk represents the inability of the receiver to pay back the loan at the designated time which was decided upon by the lender and the borrower, during the loan origination and is referred to as Credit risk.

In today's world, obtaining loans from financial institutions has become a very common phenomenon. Every day many people apply for loans, for a variety of purposes. But not all the applicants are reliable, and not everyone can be approved. Every year, there are cases where people do not repay the bulk of the loan amount to the bank which results in huge financial loss. The risk associated with making a decision on a loan approval is immense. Hence, the idea of this project is to gather loan data and use machine learning techniques on this data to extract important information and predict if a customer would be able to repay the loan or not. In other words, the goal is to predict if the customer would be a defaulter or not.

2. Literature Review

Credit Risk Management has evolved dramatically over the years. In the years 1976-1996, a number of secular forces were responsible for this evolvement. These were i) increase in bankruptcies, ii) disintermediation, iii) competitive margins, iv) declining values of assets, v) increased off-balance-sheet instruments. In response, better credit-scoring systems started to develop, concentrated risks were evaluated and new models were built to measure off-balance-sheet risks.

In 1970s, financial institutions(FIs) relied on banker "expert" systems to assess credit-risk on the basis of Character, Capital, Capacity, Collateral (4Cs). It was a highly subjective approach. (Sommerville and Taffler 1995) Eventually, accounting ratio based credit-scoring systems came into play and four methodological approaches were used: i) linear probability model, ii) logit model, iii) probit model, iv) discriminant analysis model. Martin (Journal of Banking & Finance 1997) used both logit and discriminant analysis to predict bank failures where 23 banks failed. Even though these multivariate models were efficient, they accounted multiple assumptions and hence called out for better approaches. Through this came "risk of ruin" models. These were similar to Option Pricing Models(OPM)(Black Scholes 1973) where a relationship between the beginning period market value of assets, outside debt and volatility of market value of assets is calculated. The use of volatility instead of variability questioned the efficiency of this model. Another class of models were explored, that used the yield spread and implied forward rates to know markets expectation of default. A capital based model called the mortality rate model (Altman 1989) was used by rating agencies to predict probability of default using past data on bond defaults. A newer approach used neural network analysis where the assumption of data to be linear was dropped to get better results but it showed no significant difference in results as compared to linear discriminant structure. Off-balance-sheet instruments were predicted using above models but they were less in number. The prediction for concentrated risks was implemented using migration analysis where the risk rating for a pool of loans was analysed.

Since pioneering work of Markowitz (1959), portfolio theory was applied for predictions.

The annual expected return was calculated by EAR = YTM – EAL where:

EAR: Expected annual return

YTM: Yield-to-Maturity

EAL: Expected Annual Loss

These helped in providing a rating to banks (AAA, AA, A, BBB, BB, B, CCC etc.)

Jumping straight to 2019, multiple advanced machine learning algorithms are applied to datasets in order to predict credit-risk in today's world. High speed algorithms are used in order to make these predictions. One of these, and currently one of the best algorithms is "XGBoost Algorithm". XGBoost stands for e**X**treme **G**radient **B**oosting and it is an effective implementation of gradient boost tree. The concept of Boosting is to eradicate past errors with every iteration. At every step, an algorithm is applied to boost the entire model by resolving the previous algorithm's gaps. XGBoost algorithm was used and compared to Logistic Regression model (ICCSE 2019) where it showed better speed and discriminant capability.

2.1. Outcome of Literature Review

Intense amount of research has been done on the issue of credit-risk management and numerous models and algorithms have been applied to data in order to achieve most accurate results since mid-90s but everything had its own drawbacks, ranging from theoretical approaches to multiple assumptions and inaccuracy of models in every field.

Even XGBoost, which is considered to be one of the best algorithms in many aspects, has its own disadvantages: i) It only works with numeric features, ii) Leads to overfitting if hyperparameters are not tuned properly, iii) The complexity of the final model is very high, iv) With respect to the results shown in ICCSE 2019, the stability of the XGBoost model is comparatively low. In laymen terms, there is a scope of improvement even after years of research and work.

2.2. Problem Statement

Since every model has a tendency to be improved and till date, credit-risk prediction has space for betterment, it has motivated us to study all past

developments and add our contribution to this field of machine learning. The aim of this project is to analyze data and apply multiple machine learning algorithms and compare their performance for a particular Lending Club Dataset.

2.3. Research Objectives

To study various methods of Data Analysis: Using these methods, data cleaning can be performed in order to obtain the required and desired dataset.

Data Visualization: Plots and graphs describing and portraying the relationship between different variables selected after the cleaning of dataset. Some methods can also be used for feature selection leading towards a better understanding of the model and better accuracy.

Application of Machine Learning algorithms: Learning and application of various machine learning algorithms, comparing their results and calculating precisions of all the algorithms for one particular dataset on the basis of different parameters.

3. Methodology and Framework

- Data Collection
- Data Cleaning
- Feature Selection
- Data Segmentation
- Model Training
- Results and model assessment

3.1. System Architecture

1. **Data Collection:**

Data obtained by Kaggle platform, contains information about borrowers from 2007-2015. This particular dataset is being used because most of the loans from that period have already been repaid or defaulted on. The file is a matrix of about 890 thousand observations and 75 variables.

2. **Data cleaning and transformation:**

- The first issue is to know if the columns are filled with useful information or are mostly empty. The empty and redundant columns and rows are deleted.

- All values of the target variable and segregated into two categories so that they can be provided with a numeral that represents their category (0 or 1).

- Number of missing values are calculated for every column and then filled using the python function fillna().

- The number of values in each category are brought down to similar figures so there is an equal ratio of categories.

- Columns containing string values are encoded using label encoding. Later, the entire dataset is converted to float for a consistent calculation.

- Columns like “member id” are explicitly deleted off the dataset as they hold no importance towards an accurate of prediction of a loan defaulting.

3. **Feature Extraction:**

- The most important part of the dataset is the variables that the dataset comprises of.

- For the target variable, only two fields (Fully Paid, Default) are being used as the two classes.

- A correlation matrix is being produced in order to visualize the correlation between two variables.

  Value closer to 1 suggests a strong positive correlation (directly proportional)
  Value closer to 0 suggests a weak correlation
  Value closer to -1 suggests a strong negative correlation (inversely proportional)

- This correlation matrix is then used to find the correlation of every variable with the target variable in order to recognize the important features of the dataset.

- The entire dataset is then scaled using a Standard Scaler. It is important because it helps scale the values of the entire dataset within a particular range.

4. **Data Segmentation:**

- The dataset is divided into two parts: Independent and dependent variables.

  Independent variables are used as parameters that help in determining the target variable's value.
  Dependent variable or target variable is the variable that is supposed to be predicted.

  In this project, the loan status (Fully Paid or Default) is the target variable which predicts whether a loan will completely be paid by the borrower or not.

- Then the data is divided into: Training and Test data.

  Training data is used by the machine learning model to learn the behaviour of the dataset.
  Test data of independent variables is used to predict the value of target variable for each row of the test data and for its comparison with the actual values of the test data.

5. **Model development and application:**

Preparing the model consisting of multiple machine learning algorithms to predict if the customer would be a defaulter or not.

6. **Results:**

The accuracy and confusion matrix for each model is calculated and then compared to give a better view of the impact of different algorithms on the

particular dataset.

3.2. Algorithms used

- **Logistic Regression Algorithm:**

  One of the most common, successful and transparent ways to do the required binary classification to “good” and “bad” is via a logistic function. This is a function that takes as input the client characteristics and outputs the probability of default.

  The **logistic function**, also called the **sigmoid function** is an S-shaped curve that can take any real-valued number and map it into a value between 0 and 1, but never exactly at those limits.

  1 / (1 + e^-value)

  Input values (x) are combined linearly using weights or coefficient values to predict an output value (y). A key difference from linear regression is that the output value being modelled is a binary value (0 or 1) rather than a numeric value.

  y = e^ (b0 + b1*x) / (1 + e^ (b0 + b1*x))

  Where y is the predicted output, b0 is the intercept term and b1 is the coefficient for the single input value (x). Each column in the input data has an associated b coefficient that must be learned from your training data.

- **K-Nearest Neighbours Algorithm:**

  The k-nearest neighbours (KNN) algorithm is a simple, easy-to-implement supervised machine learning algorithm that can be used to solve both classification and regression problems. The KNN algorithm

assumes that similar things exist in close proximity. In other words, similar things are near to each other.

The KNN Algorithm:

1. Load the data
2. Initialize K to your chosen number of neighbors
3. For each example in the data: Calculate the distance between the query example and the current example from the data. Add the distance and the index of the example to an ordered collection.
4. Sort the ordered collection of distances and indices from smallest to largest (in ascending order) by the distances.
5. Pick the first K entries from the sorted collection.
6. Get the labels of the selected K entries.
7. Return the mode of the K labels

- **Random Forest Classification Algorithm:**

It is an ensemble tree-based learning algorithm. The Random Forest Classifier is a set of decision trees from randomly selected subset of training set. It **aggregates the votes from different decision trees** to decide the final class of the test object.

Random Forest is a supervised learning algorithm. It can be used both for classification and regression.

It works in four steps:

1. Select random samples from a given dataset.

2. Construct a decision tree for each sample and get a prediction result from each decision tree.

3. Perform a vote for each predicted result.

4. Select the prediction result with the most votes as the final prediction.

- **XGBoost Algorithm:**

  XGBoost is a decision-tree-based ensemble Machine Learning algorithm that uses a gradient boosting framework.

  As opposed to the bagging where trees are built parallely, in boosting, the trees are built sequentially such that each subsequent tree aims to reduce the errors of the previous tree.

  Steps:

  1. Fit a model to the data, F1(X)=Y

  2. Fit a model to the residuals, h1(X) = Y - F1(X)

  3. Create a new model, F2(X) = F1(X) + h1(X) [Note: F2 is boosted version of F1]

  and goes on… Fm(X) = F(m-1) (X) + h(m-1) (X)

  Notice, here hm is just a model and not just a tree based model. Therefore, Gradient boosting is just a framework where one can plug in any model although plugging in tree-based models gives better results.

  In XGBoost, we fit a model on the gradient of loss generated from the previous step. In XGBoost, we just modified our gradient boosting algorithm so that it works with any differentiable loss function.

The results of these algorithms would then be compared or combined to form a model with better accuracy.

4. Work Done

1. Data set obtained from kaggle.
2. Preparation of synopsis for the project.
3. Research paper analysis about the existing work of this field through Review paper from (1976 – 1996) and a research paper published in 2019.
4. Data Cleaning of obtained dataset in EXCEL.
5. Data Pre-processing.
6. Data Analysis and Visualization.
7. Model Formation and Result Comparison.

4.1. Results and Discussion

1. The original dataset contained multiple columns that were entirely empty or with most of their values as missing which was required to be removed.
2. Dataset also consisted of columns that carried redundant data which only took the model towards overfitting and hence, had to be eliminated.
3. "loan_status" is the target variable.

   It consists of the following unique values:
   Current ; Fully Paid ; Charged Off ; Late (31-120 days) ; Issued ; In Grace Period ; Late (16-30 days) ; Default ; Late
   Since our project consists of a binary classification approach, only two distinct values are considered i.e**. Fully Paid** and **Default** (consisting of Charged Off, Late (31-120 days), Default, Late).
4. The ratio **(Fully Paid: Default)** is approximately 170000: 50000 which makes the model biased. Hence approximately 80000 **'Fully Paid'** values are randomly removed to restore the model's balance.
5. Null values are removed using fillna() method.
6. **Relation between the average loan amount and loan status:** Loans of greater amount have a greater tendency to default.
7. Most people who apply for loans either have their **own home or a home on**

rent or mortgage.

8. Borrowers who have their source of **income verified** are more likely to fully pay back their loans.

9. The **purpose** for which a loan is applied for plays an important role in its default prediction. Loans taken up for a **small business** have the highest default rate.

10. **Initial list status** has approximately equal default rate for all its unique values and hence is discarded.

11. Most borrower file for **Individual loans** over **Joint loans** and hence this column is removed since it contains only one unique value, mostly.

12. By correlation matrix analysis, some of the **relevant variables** are:
    term(months)
    int_rate
    grade
    dti
    revol_util
    total_pymnt
    total_rec_prncp
    total_rec_late_fee
    recoveries
    collection_recovery_fee
    last_pymnt_amnt
    loan_status

13. Final dataset shape: **(142375, 20)**
    After creating dummy variables (to decrease redundancy): **(142375, 38)**

14. **Logistic Regression:**

    Accuracy: 96% approx.

    Precision:
    0 (0.96)
    1 (0.98)

    Recall:
    0 (0.97)
    1 (0.98)

    F1 Score:
    0 (0.96)
    1 (0.97)

15. **K-nearest Neighbours:**

    Accuracy: 89% approx.

    Precision:
    0 (0.86)
    1 (0.89)

    Recall:
    0 (0.81)
    1 (0.93)

    F1 Score:
    0 (0.83)
    1 (0.91)

16. **Random Forest:**

    Accuracy: 96-97% approx.

    Precision:
    0 (0.90)
    1 (0.99)

    Recall:
    0 (0.98)
    1 (0.94)

    F1 Score:
    0 (0.94)
    1 (0.96)

17. **XGBoost:**

    Accuracy: 98% approx.

    Precision:
    0 (0.93)
    1 (0.96)

    Recall:
    0 (0.94)
    1 (0.96)

    F1 Score:
    0 (0.93)
    1 (0.96)

4.2. Individual Contribution of project members

Shruti Gupta:
Data cleaning
Encoding
Data visualization
Correlation matrix
KNN Model application
Random Forest Model application
Result comparison.

Vidhi Garg:
Data pre-processing
Feature scaling
Data visualization
Dummy variable creation
Logistic Regression Model application
XGBoost Model application
Result comparison.

5. Conclusion and Future

To conclude, the project **"Credit Risk Analysis"** aims at studying various components affecting the probability of whether a loan would be fully paid back or defaulted, select relevant components and then use them with multiple machine learning classification models. These models are then compared using metrics like accuracy, precision, recall and F1 score.

The project has a very vast scope in the future. Variations that can further be applied:

- Using different ensemble models.
- Using datasets from other sources and time-periods.
- Combining more than one dataset for predictions.
- Creating hybrid models.

6. References

- https://machinelearningmastery.com/precision-recall-and-f-measure-for-imbalanced-classification/#:~:text=Precision%20for%20Multi%2DClass%20Classification,false%20positives%20across%20all%20classes.
- https://medium.com/@pushkarmandot/how-exactly-xgboost-works-a320d9b8aeef
- https://xgboost.readthedocs.io/en/latest/
- https://towardsdatascience.com/https-medium-com-vishalmorde-xgboost-algorithm-long-she-may-rein-edd9f99be63d#:~:text=XGBoost%20is%20a%20decision%2Dtree,all%20other%20algorithms%20or%20frameworks.
- https://www.tutorialspoint.com/machine_learning_with_python/machine_learning_with_python_classification_algorithms_random_forest.htm
- https://www.datacamp.com/community/tutorials/random-forests-classifier-python
- https://towardsdatascience.com/random-forest-classification-and-its-implementation-d5d840dbead0
- https://www.gepsoft.com/GeneXproTools/AnalysesAndComputations/MeasuresOfFit/Classification.htm
- https://stackoverflow.com/questions/38077190/how-to-increase-the-model-accuracy-of-logistic-regression-in-scikit-python#:~:text=Hyperparameter%20Tuning%20%2D%20Grid%20Search%20%2D%20You,the%20hyperparameters%20of%20your%20model.&text=Also%2C%20you%20should%20avoid%20using,numbers%20for%20your%20final%20model.
- https://www.geeksforgeeks.org/python-difference-between-pandas-copy-and-copying-through-variables/
- https://towardsdatascience.com/two-is-better-than-one-ensembling-models-611ee4fa9bd8
- https://www.w3schools.com/python/ref_random_choices.asp
- https://www.datacamp.com/community/tutorials/joining-dataframes-pandas
- https://www.researchgate.net/post/How_do_I_reduce_number_of_Independent_Variables_before_running_Logistic_Regression
- https://www.statisticshowto.com/correlation-matrix/
- https://datatofish.com/correlation-matrix-pandas/
- https://cmdlinetips.com/2018/01/how-to-get-unique-values-from-a-column-in-pandas-data-frame/#:~:text=We%20can%20use%20Pandas%20unique,on%20the%20column%20of%20interest.
- https://www.geeksforgeeks.org/working-with-missing-data-in-pandas/
- https://stackoverflow.com/questions/48585947/in-fillna-what-is-the-difference-between-pad-and-ffill-method
- https://datascience.stackexchange.com/questions/39137/how-can-i-check-the-correlation-between-features-and-target-variable

- https://www.investopedia.com/terms/l/loan-grading.asp
- https://towardsdatascience.com/feature-selection-with-pandas-e3690ad8504b
- https://ml-cheatsheet.readthedocs.io/en/latest/logistic_regression.html
- https://machinelearningmastery.com/logistic-regression-for-machine-learning/
- https://www.statisticssolutions.com/what-is-logistic-regression/
- https://blog.exsilio.com/all/accuracy-precision-recall-f1-score-interpretation-of-performance-measures/#:~:text=Recall%20(Sensitivity)%20%2D%20Recall%20is,observations%20in%20actual%20class%20%2D%20yes.&text=F1%20score%20%2D%20F1%20Score%20is,and%20false%20negatives%20into%20account.
- https://scikit-learn.org/stable/modules/model_evaluation.html