Classifying Loan Defaulter using Machine Learning Techniques

Table of Contents

1. **Introduction**
2. **Motivation**
3. **Dataset**
   1. Data Collection
   2. Data Preparation
4. Methods
   1. Naïve Bayes
   2. KNN
5. Demo
6. Evaluation
7. Execution Instructions
8. Source code
9. References
10. **Introduction**

The idea behind this project is to build models for Loan Data provided by Lending Club that summaries a large amount of it’s customers’ data to give an overview of the trend of default status among customers who have taken a loan there. We are building a Naïve Bayes and a K-Nearest Neighbor Classification model to classify the customers into “Defaulted” and “Not Defaulted” status. The main task involved in this project are a follows:

1. Data Cleaning
2. Missing value imputations
3. Model building
4. Split the data into training and test sets.
5. Train the model based on the training set.
6. Predict loan status for the entire test set.
7. Calculate the model accuracy.
8. **Motivation**

**Lending Club** is the world’s largest online marketplace connecting borrowers and investors. Lending Club helps to make credit more affordable and investing more rewarding. They operate at a lower cost than traditional bank lending programs and pass the savings on to borrowers in the form of lower rates and to investors in the form of solid returns.

Any person who wants to provide loan for some interest can advertize to provide loan at a particular interest rate. Similarly, anyone who intends to borrow loan would advertize to get loan at his desired interest rate. Lending club tries to match these loan investors and loan borrowers.

The idea of being able to predict whether a customer would default on his loan or not, even before he is sanctioned a loan is an exciting one. This approach is also of utmost importance to any financial institution. Although, the models we build might not be perfect, it will help the institution and give them an idea of what to expect from a customer. The data we have collected might be from Lending Club, but this solution can be applied to any financial service, which makes this project even more useful and important.

The direct applicability of this model to real world problems in banking and financial firms makes this an interesting and cool project to work on. Also, so far we have only imported libraries to build models in R, Python and hence implementing from scratch will be a challenge.

1. **Dataset**

**Dataset Description:**

We used the loan data provided by Lending Club on Kaggle website. The files that we downloaded contain complete loan data for all loans issued to the customers through the 2007-2015, including the current loan status which is our predictor variable. The file is in the csv format and has a size of about 1 GB. It is a matrix of about 890 thousand observations and 75 variables. The data dictionary was provided along with the data which helped us understand the different variables involved. As every column is not necessary for the prediction, we chose only few of the most important numerical columns to predict the target variable. As mentioned above, our predictor variable is “loan\_status” which has the details of open, currently running and closed loans indicating whether a person has defaulted on the loan taken or not. And our purpose is to build models to accurately predict this classification.

**Features selected:**

|  |  |
| --- | --- |
| Features | Description |
| Int\_rate | Interest Rate on the loan (Numeric) |
| Loan\_amt | Amount of loan taken by the customer(Numeric) |
| Annual\_inc | Annual income of the customer(Numeric) |
| Open\_acc | Open credit lines in the customer’s account(Numeric) |
| Revol\_bal | Credit revoke balance |
| Total\_acc | Total number of credit accounts under the customer’s name |
| Revol\_util | Amount of credit the customer is using relative to all relative accounts |
| Delinq\_amnt | Past-due amount owned on the accounts |
| Dti | Ratio of the customer’s total monthly debt payments divided by his/her self-reported monthly income |