Loan Data Analysis

EE 660 Course Project

Project Type (1) Design a system based on real-world data

Number of student authors: 1

Weiding Huang

[weidingh@usc.edu](mailto:weidingh@usc.edu)

03/12/2020

1. Abstract
2. Introduction
   1. Problem Type, Statement and Goals

Whether to issue a loan to a person requires a lot of investigation and time, and it will cause losses to the loaner if a lender is behind the due and the debt must be charged off. Typically, banks spend lots of resources and time to review loan applications. In order to help banks to expedite the process, this project is aimed to build a model and classify whether a loan is going to be paid off or charged off. The data set used in the project contains 18 feature spaces and 10000 data points. Among these 18 features, 8 features are categorical data and 6 features contain missing data points. Due to the large data quantity, mixing of categorical and numeric data and many missing data points, a comprehensive preprocessing step needs to be conducted in order to get the best result.

* 1. Literature Review

The project was previously done by Victor Hugo Pereira on Kaggle (<https://www.kaggle.com/panamby/bank-loan-status-dataset>). He filled in missing data points using the mean values, and then removed collinear features. For classification models, he used logistic regression, k-nearest neighbors, support vector machine, naïve Bayes, gradient boosting, and random forest. In this project, the best model was gradient boosting, resulting in an 81.7% accuracy on test set.

* 1. Our Prior and Related Work

None

* 1. Overview of Our Approach

In the project, four different algorithms were experimented to optimize the result. They are logistic regression, random forest, k nearest neighbor, and EM estimation. In addition, linear regression is used to predict and fill in the missing data points according to other data that does not have missing features. All the models use the default parameters except for logistic regression. Because logistic regression sometimes would not converge under the default number of iterations, so the number of iterations was set to 1000.

1. Implementation
   1. Data Set

The bank loan dataset contains two sets: a training set and a test set. The training dataset contains 18 different features and 100514 data points (last 514 are NaN). The test dataset contains 17 features (missing loan status) and 10353 data points.

|  |  |  |  |
| --- | --- | --- | --- |
| Feature name | Categorical/  numerical | Number of missing data points | Description  (cardinality, range) |
| Loan ID | Numerical | 0 | Serial number (eg.14dd8831-6af5-400b-83ec-68e61888a048) |
| Customer ID | Numerical | 0 | Serial number (eg.981165ec-3274-42f5-a3b4-d104041a9ca9) |
| Loan Status | Categorical | 0 | Fully Paid, Charged Off |
| Current Loan Amount | Numerical | 0 | 10.8K – 100M |
| Term | Categorical | 0 | Short Term, Long Term |
| Credit Score | Numerical | 1950 | 585-7510 |
| Annual Income | Numerical | 1950 | 81.092K -17.8M |
| Years in current job | Categorical | 424 | <1 year, 2 years, 3 years, 4 years, 5 years, 6 years, 7 years, 8 years, 9 years, 10+years |
| Home Ownership | Categorical | 0 | Home Mortgage, Rent, Own Home, HaveMortgage |
| Purpose | Categorical | 0 | Debt Consolidation, Home Improvements, Business Loan, Buy a Car, Medical Bills, Buy House, major purchase, Take a Trip, small business, Educational Expenses, moving, wedding, vacation, renewable energy |
| Monthly Debt | Numerical | 0 | 0 - 436K |
| Years of Credit History | Numerical | 0 | 3.6 - 70.5 years |
| Months since last delinquent | Numerical | 5224 | 0 -131 Months |
| Number of Open Accounts | Numerical | 0 | 1 - 55 |
| Number of Credit Problems | Numerical | 0 | 0 - 10 |
| Current Credit Balance | Numerical | 0 | 0 – 16.2 M |
| Maximum Open Credit | Numerical | 0 | 1 – 14 M |
| Bankruptcies | Categorical | 21 | 0 (not bankrupt), 1 (bankrupt) |
| Tax Liens | Categorical | 0 | 0 (not a tax lien), 1 (tax lien) |

* 1. Data Methodology

Firstly, by just observing the data on Kaggle, there are 514 rows of all NaN data. These were dropped. Then 5% of the data (500 data points) was used as pre-training data in order to determine the algorithms needed for filling the missing data and prediction models. 10 % of the data (1000 data points) was used as test data and 85% of the data (8500 data points) was used as training data. The split was done before any preprocessing to prevent any data snooping.

Cross validation was used to compute average accuracy for each algorithm. After finish preprocessing step, a 5-fold cross validation was used to exam the results of training models. And the model with the best average accuracy was used in test dataset that was separated from the original training set. Also, the same model is used to predict result in the original test set. Both of the separated test set and original test set were used only once.

* 1. Preprocessing, Feature Extraction, Dimensionality Adjustment

The first step was to get rid of useless features such as loan ID and customer ID which has no affects to the prediction accuracy.

The second step of preprocessing was to know how many data points are missing in each respective feature. The graph below shows the missing data points.

Diagram

Description automatically generated

*Figure 1*: Visualization of missing data points

Feature ‘Months since last delinquent’ was mostly missing, so it was dropped.

Next step was to convert all the categorical data to numerical data.

Term: Short Term: -1, Long Term: 1

Years in current job: < 1 year: 0, 1 year: 1, 2 years: 2, 3 years: 3, 4 years:4, 5 years: 5, 6 years: 6, 7 years: 7, 8 years: 8, 9 years: 9, 10+ years: 10

Home Ownership: Rent: 1, Own Home: 2, Have Mortgage: 3, Home Mortgage:4

After all the data has been converted to numerical data, filling in the missing data points was the next step. The basic concept of how to fill in missing data is that a feature that has missing data are treated as unknowns and predicted by other features that do not have any missing data points. The algorithms used to predict the value was linear regression because of most of the features have linear patterns.

* 1. Training Process

Logistic regression: logistic regression is often used in binary classification. It gives a probability of a data point with certain features belonging to a class: Logistic regression is a suitable algorithm for this project since this project is to predict a two-class result, whether the loan is going to be paid off or charged off. The logistic regression algorithm provided in sklearn allows to choose regularization from none or l2, but the results were almost identical. It might be the optimal solution is already inside the regularized area, so the regularization did not do much.

Random Forest: random forest consists of multiple decision trees. It uses bagging to average the prediction from every decision tree and taking the Chart, radar chart

Description automatically generatedmajority vote as the class for each stump.

Figure simplified random forest

In random forest, different max depth affects the performance. If the tree is too deep, it might cause the data to overfit. Or the tree is too shallow, it has a lower accuracy. So multiple depths (none, 5, 10, 15, 20) were chosen to find out the best depth parameter.

K nearest neighbor: k nearest neighbor has the similar methodology like the random forest. A unknown point is classified by the majority vote of its k nearest neighbor

* 1. sadsadas

1. Final Results and Interpretation
2. Contributions of each team member
3. Summary and conclusion
4. References