Predicting Loan Defaults with Logistic Regression

Matt Allen

07/29/2018

## 1. Executive Summary

## 2. Introduction

The purpose of this project is to build a model to predict whether or not a borrower will pay back a loan. The description of the data set is [here](https://datascienceuwl.github.io/Project2018/loans50k.csv). The model will give a simple “Good” or “Bad” indicating the quality of the loan. The project will begin by preparing and cleaning the data. Variables will be selected by domain knowledge versus an automated method. Through exploration of the selected variables, the variables may be transformed to satisfy conditions of model fitting like being normally distributed for example. The data will be fitted to a logistic model with a training data set. Part of the data set will be held out as test data set to validate the model. The model is binomial and the mean also called the classification threshold can be adjusted to optimize for accuracy or profit.

## 3. Preparing and Cleaning the Data

The first step in preparing the data was to create a new column named response that has the values “Good” and “Bad”. It was based on the variable status. Only the rows with status of “Fully Paid”, “Charged off”, and “Default” were kept. “Fully Paid” maps to “Good”, and “Charged off” and “Default” were mapped to “Bad”.

The initial dataset consisted of 50,000 records. After reducing “Good” and “Bad”, the total number of records left were 34,655 records. The number of records with the variable response equal to “Good” was 27,074, and “Bad” was 7,581.

There were variables that could be removed, because they were not useful. For example, the status variable may be removed, because it has been transformed into the response variable. The totalPaid variable can be removed, because it is not knowable at the time of issue of a loan. The variable loanid was removed because it is just an identifier and has no predictive power.

The variable employment, which indicates job title, can be removed because there is so much variation. It would be possible to create a new variable with job title that puts titles in more general categories, but it probably is still not useful. A better indicator may be income from job.

The variable length indicates the length of employment. This could be a good indicator, but there are more than 1300 missing values. It should be removed, because removing the rows with missing income would significantly reduce the data set, and imputation methods may be challenging for this much missing data where there is not necessarily a good proxy.

The variable revolRatio indicates proportion of revolving credit in use. This can be removed, because it is captured in other variables like debtIncRatio. With similar resoning, bcRatio can be removed. The variable accOpen24 can be removed, because if an account is opened there will likely be a credit inquiry, which is captured in inq6mth.

The variable grade indicates the risk of the loan. I would consider it to be another response variable like status or the response variable derived from status, but with more levels. For this reason, it should be eliminated. The variable rate can be removed for similar reasons as grade. The loan rate is an indication of the borrower’s risk. The greater the risk the higher the rate.

The variable pubRec can be eliminated, because the variable delinq2yr should capture this information. If someone has pubRec against them, they would have also missed payments.

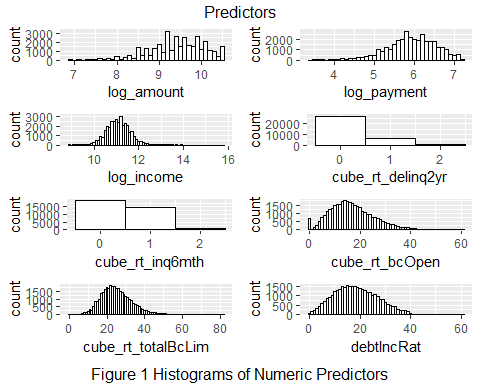
After inspection of the histograms of the variables totalBal and totalLim, it was discovered that their distributions are bimodal, which violates condition of normality, so debtIncRatio was used as a proxy. The variable openAcc was found not to be significant and was also removed.

After elimination of variables, the predictors that are left are payment, amount, verified, income, debtIncRat, delinq2yr and inq6mth. The variable verified is categorical, the rest are quantitative.

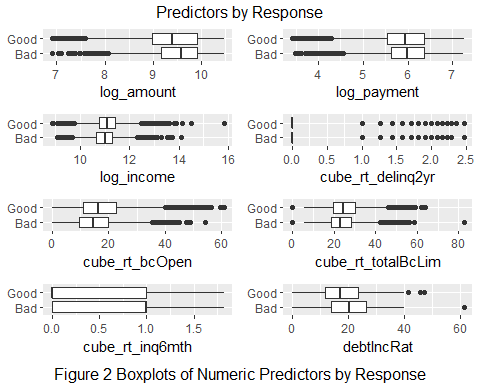
With the remaining variables, there is no imputation or record removal needed.

## 4. Exploring and Transforming the Data

The variables amount, income, delinq2yr, inq6mth showed varying degrees of right skewness. From inspection of histograms, the variable amount, payment and income showed the most right skewness and were transformed by the log function. The variables delinq2yr and inq6mth were transformed by taking the cubed root. Histograms of the transformed variables are displayed in Figure 1.



To develop a picture of the relationships between the predictor variables, and the response, two boxplots for each response “Good” and “Bad” were created for a predictor variable. The boxplots were created to explore the differences in distributions between “Good” and “Bad” loans for each predictor variable. The plots can be seen in Figure 2. Looking at the log\_amount variable box plot, the mean of amount is lower for Good than Bad, which makes sense. A smaller loan would be easier to pay back. Similarly with the variable log\_payment, a loan that has a smaller payment makes it easier to make the payment each month. Also as indicated by log\_income, those with greater incomes have more money to pay back a loan.



After inspection of the boxplot for cube\_rt\_delinq2yr, the variable delinq2yr was removed from the model, because even after transformation it is extremely right skewed. Taking it out of the model did not effect the accuracy or profit predictions significantly.

After plotting and exploring of several other predictors, the shapes of bcOpen and totalBcLim after cube root transformation were found to be close to normal. Adding them to the model decreased AIC, and increased predictive power. The terms are significant. The variables bcOpen and totalBcLim which indicate credit card utilization may be proxies for totalBal and totalLim, which were removed because of their bimodal distributions. The variable bcOpen had 360 values with NA. These were removed from the data set.

## 5. The Logistic Model

From the summary of the model all the variables are very significant as indicated by p-values that are close to zero. Based on the VIF, the variables log\_payment and log\_amount show correlation. However, they are still very significant in the model, and will be kept.

The model with the coefficients is below.

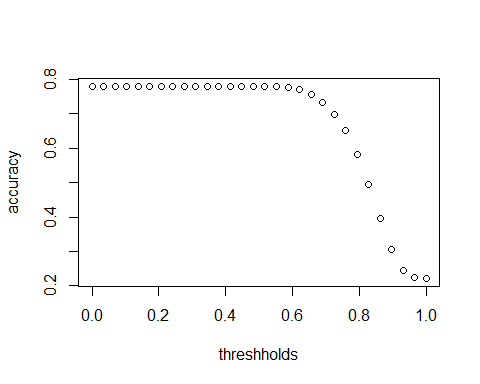
The contingency table along with the overall accuracy is below. The accuracy is at 79%. That is 79% of the time truly good or bad loans are classified correctly. Based on the accuracy and that the terms are all significant, this is a decent model. From the contingency table, there are relatively few loans predicted as bad compared to good loans. The model may be good at predicting good loans, and bad at predicting bad ones.

## predLoan  
## Bad Good Sum  
## Bad 27 1494 1521  
## Good 20 5318 5338  
## Sum 47 6812 6859

## [1] "Proportion correctly predicted = 0.779268114885552"

## 6. Optimizing the Threshold for Accuracy

The maxmimum accuracy occurs near the threshold of .55. As you can see from the plot above of accuracy vs threshold, the accuracy is mostly flat from 0 to 0.6, but from inspection of the data peaks at .55. It then declines between 0.6 and 1.0.

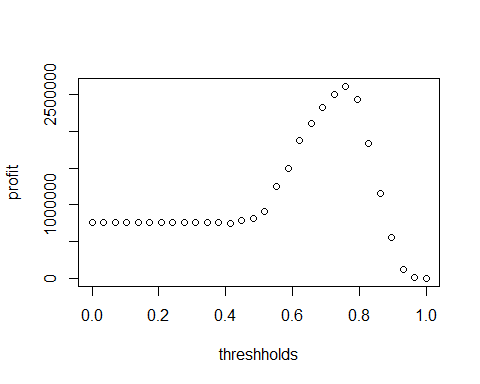


## 7. Optimizing the Threshold for Profit

The maximum profit of $2,609,607 occurs at a threshold of .76. The best profit threshold corresponds to an accuracy of 65%. The profit where the model is the most accurate at a threshold of 55% is $1,247,186. The maximum accuracy and profit thresholds do not coincide.

If no loans were denied the total profit would be $758,390. The increase from using the model is 244%.

If the model had predicted all the good loans perfectly, the total profit would be $12,020,435, which represents an increase of 1485%. Perfect prediction is at least 6 times better, so may be the model could be improved.



## 8. Results Summary

The classification threshold that produces the maximum profit is .76, which gives an accuracy of 65%.

Given a bad loan, the model will correctly predict it 54% of the time.

Given a good loan, the model will correctly predict it 68% of the time.