Give Me Some Credit – Assessing a Potential Customer's Credit Risk

Nathan Kruse and Ibur Rahman

Math 8436

University of Nebraska at Omaha

Abstract

Banks play a crucial role in market economies. They decide who can get financing and on what terms and can make or break investment decisions. For markets and society to function, individuals and companies need access to credit. Credit scoring algorithms, which make a guess at the probability of default, are the method banks use to determine whether or not a loan should be granted. For this project, we try to improve on the state of the art in credit scoring, by predicting the probability that somebody will experience financial distress in the next two years. This will allow lenders to make the best possible financial decisions. The data are obtained from Kaggle on their website "Kaggle.com."

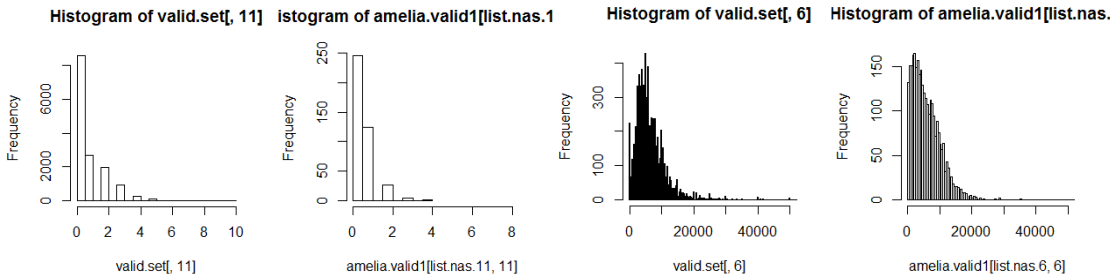
Banks decide who gets financing and on what terms. In order for them to make prudent decisions about who should receive loans and on what terms, they need a good credit scoring algorithm. In this paper we investigate a dataset with the goal of creating a superior credit scoring algorithm. All data manipulation and model fitting was done using R statistical software.

This paper will be structured as follows. In Section 1 we will discuss the dataset and the data cleaning process. In Section 2 we will discuss the selection of models and their individual matrix preparation. Section 3 will discuss how the models were assessed. Finally, in Section 4 we discuss our findings and conclusions.

1. The data and data cleaning process

This dataset contains data that are from approximately 150,000 people, and included are ten explanatory variables and one binary response variable. The explanatory variables are: percent of revolving credit used, age, number of times 30 to 59 days past due, number of times 60 to 89 days past due, number of times 90 days or more past due, debt ratio, monthly income, number of open credit lines and loans, number of real estate loans, and number of dependents. The binary response variable is whether or not the individual has had a "serious delinquency" in the past two years.

The data include a significant number of NA values for monthly income and number of dependents. While it is possible to replace the NA values with the column means, or even zero's if desired, we have chosen to use the Amelia package in R to deal with our missing data. Amelia imputes data under the assumption that the data are normally distributed with some mean μ, and covariate matrix Σ. Although these assumptions will not hold with our data, the authors of Amelia claim that their algorithm performs well even in the case of non-normal data, including categorical or count data (Blackwell, Honaker, & King, 2015). This seems to be the case with our data. We believe these imputations to be sufficient. The histograms show the distribution of a random sample of 15,000 monthly incomes and number of dependents next to a histogram of the imputed values from Amelia.



2. Model selection and matrix preparation

Datasets that include a binary response variable can be modeled by many different types of models including: logistic regression, linear discriminant analysis (LDA), quadratic discriminant analysis (QDA), gradient boosting, trees, random forests, neural networks, among others. We fit each of these models, some with many different parameterizations and variables, to the dataset and shall share the results of many of these. However, we shall only investigate in detail here the three models that we found to best fit this dataset; a logistic regression model, an LDA model, and a deep learning neural network model.

Logistic Regression Model

The logistic regression model is frequently used for modeling data with binary response variables. Given that our dataset has a binary response, logistic regression is, of course an appropriate model type for this dataset.

To fit a "good" logistic regression model, one must choose appropriate explanatory variables. This could mean removing some variables from the dataset, performing transformations on the data, creating new variables from information derived from the data, or including interaction terms. We chose the latter. For our best regression model, we used all 10 explanatory variables, 29 pairwise interactions, 11 triple interactions, and 1 quadruple interaction. While some of the original explanatory variables were not significant on their own, their interactions were very significant so we included them in the model.

Deep Learning Neural Network

As computer processing speed continues to increase, neural networks are becoming progressively more popular as a way to model all types of data, to include classification problems. We too, thought that we could train a neural network to make accurate predictions based on our data.

Training a neural network requires a number of different parameters to be tuned. One must supply the data to be used, the loss function(s), the learning rate, the activation function(s), the number of hidden layers, and the number of nodes in each layer. Depending on which R package is used, more may be required.

Training a neural network is also computationally expensive, and each time a parameter is tuned, the network must be retrained. Because of the expensive nature of neural networks, we were not able to train with as many variables as we did with the logistic regression model. For our best neural network we only used the original variables from the dataset to train the model. We set up 3 hidden layers with 28 nodes each, selected a cross entropy loss function, rectifier with dropout activation function, and fixed the learning rate at 0.5.

LDA

LDA is another popular model for classification problems. Unlike logistic regression where the response variable is estimated by conditioning on the explanatory variables, LDA estimates the distribution of each explanatory variable by conditioning on the response variable. Then, Bayes' Theorem is used to change back to conditioning the response on the explanatory variables. Since LDA is a common choice for this type of data, we too chose to use it.

Since training LDA models is not nearly as expensive as neural networks, we were able to use many more variables in our final model. In fact, we used almost as many as our logistic regression model. We used: all 10 explanatory variables, 29 pairwise interactions, and 8 triple interactions.

3. Model assessment

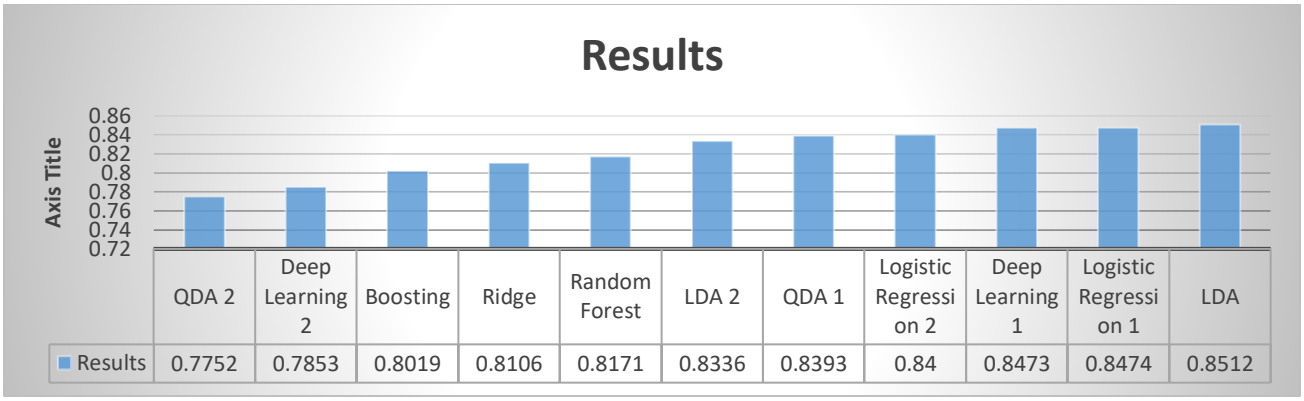
Model assessment was done completely through Kaggle submissions. Kaggle assessed the models using the Area Under the Curve (AUC) method instead of the simpler mean accuracy/error rate. This is because of the data itself. The response is very unbalanced, 93.3% of the training responses being "no." This means that simply guessing "no" for every row in the dataset will lead to a very high 93.3% accuracy rate. However, it is more interesting and useful to be able to predict which customers are likely to have "serious delinquencies" during the next two years rather than just predicting that 93.3% of them will not. AUC is a different way to measure how "well" a model predicts.

The AUC is calculated by using a combination of the "true positive rate" i.e. the proportion of the time the model accurately predicts a serious delinquency, and the "false positive rate" i.e. the proportion of time the model inaccurately predicts a serious delinquency. This curve is set inside of a unit square and the area under the curve is the AUC. An AUC of 1 is equivalent to perfect predictions and an AUC of .5 is equivalent to predicting all negative outcomes.

4. Findings

Using the AUC results returned from Kaggle as a measure, our best results were the three models described above. The neural network returned an AUC of .8474, logistic regression returned .8488, and LDA returned .8512. Our results were comparable to other submissions in the Kaggle competition, where the winning submission had an AUC of .8696.

The three submissions noted here were the best of the dozens that we submitted. Below is a graph of the AUC for many of those submission attempts, where 1 and 2 attached to model names represent our first and second best parameterizations.



We were not surprised to see that logistic regression models and LDA models performed nearly identically. We do believe however that a neural network could achieve better results than even our best LDA model, given that one has enough processing power. Our best neural network, for efficiency, was only trained on the 10 original variables, where as our logistic regression model for example, used 41 explanatory variables. These models achieving nearly identical results. The addition of interaction terms or new variables when training the neural network would likely lead to better results.

Based on our results, we believe that while traditional models like logistic regression and LDA do an adequate job modeling the data, in the future neural networks including deep learning methods may prove to be much more useful.

References

Blackwell, M., Honaker, J., & King, G. (2015, December 5). *Amelia II: A program for missing data*. Retrieved from https://cran.r-project.org/web/packages/Amelia/vignettes/amelia.pdf

Kaggle Inc. (2011). *Give me some credit*. Retrieved from https://www.kaggle.com/c/GiveMeSomeCredit