Predicting Loan Status with Logistic Regression

Jeremy McCormick

11/23/2021

## Executive Summary

### Problem Statement

Predicting whether a loan applicant will eventually default is important for managing the profitability of a bank’s lending operations. Profits would typically be negative if every loan were approved due to the high number of defaults. A typical loan application process will filter out only a portion of these bad loans, leading to lower profits than if some of these could be initially screened out. We explore the capability of a statistic model to identity which loans are likely to be good in order to maximize the overall profit.

### Proposed Solution

In this study, we have created a statistical model which rates loans based on their probability of being good. We started by cleaning the data and removing non-predictive variables, as well as those which were considered redundant, too difficult to work with, or poorly correlated with loan status. We explored the association of the remaining variables with loan status using detailed graphs. A logistic model was developed to predict the status, using an optimization technique to reduce the number of predictors. The final model was then tuned separately for accuracy and profit. The most profitable model is summarized in detail.

### Value

Though the overall predictive power of the model ends up being weak in identifying bad loans, it still provides considerable benefits compared with using no model at all, with an increased profit of over 300 perfect. Even though we did not achieve a high level of predictive accuracy for the bad loans, the model is still able to provide a considerable increase in profitability.

### Conclusion and Next Steps

This study provides a proof of principle that even a weakly predictive model for loan status can still provide benefits in terms of increased profits. Additionally, the study identifies which variables are most predictive in this area and could be used as the basis for additional analysis using more advanced modeling techniques to improve the results even further.

## Introduction

### Data

The uncleaned data contains 50,000 records and 32 data columns as described in the [documentation](https://datascienceuwl.github.io/Project2018/TheData.html). The response variable **status** contains the current status of the loan and will be transformed into a variable with two levels: “Good” or “Bad.” Data Columns will be dropped if they appear to be unuseful for prediction.

### Research Questions

Is it possible to accurately predict whether applicants are likely to default on their loans? Can the accuracy of the prediction be tuned in order to maximize profit?

This will help Eau Claire Bank decide which loan applications should be granted or rejected.

## Preparing and Cleaning the Data

### Load Data

The data was loaded from a CSV file.

loans <- read.csv("loans50k.csv")

### Prepare the Response Variable

There were 50000 rows in the initial dataset. The status of the loan was transformed into a binary variable depending on its value. Just over 15 thousand records were removed based on their status, with 34655 remaining for analysis. Of these, 7581 were bad and 27074 were good, so there were almost four times as many loans with good status compared to bad.

### Data Cleaning

Missing data in the form of blank strings was transformed into *NA* values. Then all the rows containing *NA* in any of their columns were dropped. This seemed reasonable compared with interpolation, since this operation results in only a small fraction of the total records being removed (less than 500).

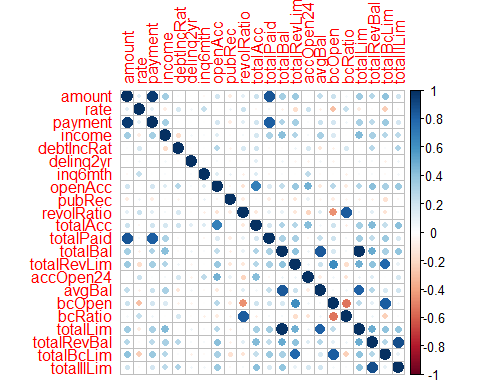
### Eliminate Variables

The following variables were initially dropped from the data:

* **loanID** - non-predictive ID number
* **employment** - contains far too many values for a reasonable categorical variable and re-encoding them seems impractical
* **reason** - slightly too many different values for a good categorical and there is already an “other” category so reducing the number of categories would be problematic (It also may be the case that rejecting a loan based on the reason could violate standards of fairness.)
* **state** - too many different values for a categorical (50) and unlikely to be very predictive compared with other columns (Loan applicants should also not be discriminated against based on their location.)

The **totalPaid** field will be dropped from the testing data and will not be used as a predictor, but it is kept in the main data frame, as it will be used for profit calculations later in the analysis.

A correlation matrix can be used to find collinearity between predictors and possibly eliminate some of them as redundant.



The following additional variables were dropped based on high levels of correlation:

* **revolRatio** - highly correlated with **bcRatio**
* **avgBal** - highly correlated with **totalBal**
* **totalLim** - highly correlated with **totalBal** (assume the balance is more important than the credit limit)
* **totalBcLim** - highly correlated with **totalRevLim**
* **totalAcc** - highly correlated with **openAcc** (presumably do not care about the applicant’s number of closed accounts from their credit history)
* **bcOpen** highly correlated with **totalRevLim**

### Feature Engineering

The **verified** predictor was simplified into “yes” or “no,” assuming that “Verified” and “Source Verified” were equivalent.

To reduce the number of categories and combine ones with relatively small numbers of entries, the letter grade of the loan was set to “high,” “medium,” or “low.”

The employment length was converted to just a few categories including “short,” “medium” and “long.” The field was also renamed from **length** (an R built-in name that should be avoided) to **employed**.

The **term** field was changed to either “short” or “long” based on the loan period, corresponding to a 36 or 60 month period, for readability in model results.

After transforming the data appropriately, the categorical columns were transformed into factors.

## Exploring and Transforming the Data

### Data Scaling

The skewness was calculated for all numeric variables, and a list was made of those with high skew, i.e. greater than 1 or less than -1.

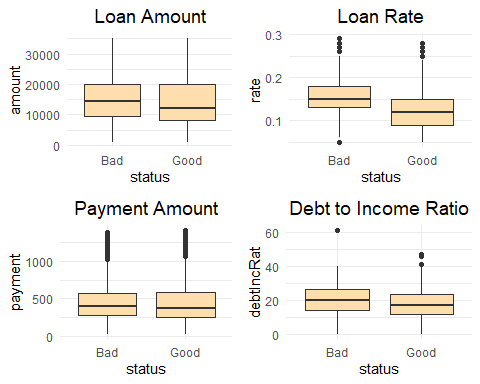
## [1] "High skew columns:"

## [1] "income" "delinq2yr" "inq6mth" "openAcc" "pubRec"   
## [6] "totalPaid" "totalBal" "totalRevLim" "accOpen24" "totalRevBal"  
## [11] "totalIlLim"

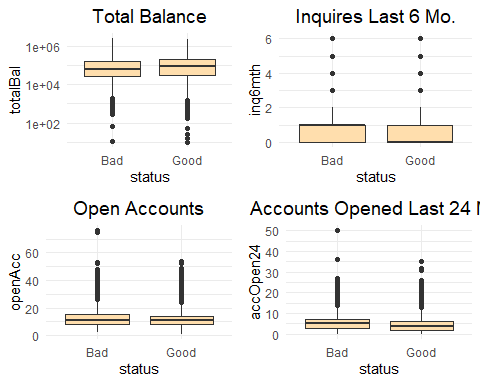
The columns with high skew were transformed to new ones with log scale. The original columns were kept in the data frame for accessibility and visualization.

### Data Exploration

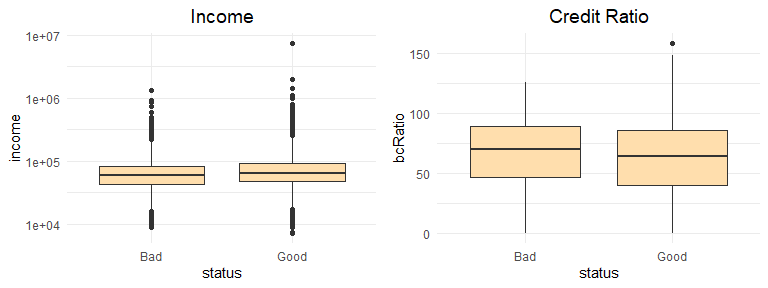
Boxplots were used to compare numeric values by loan status.



From these plots, the **amount** and **rate** showed a significant difference in median values based on loan status, whereas **payment** and **debtIncRat** did not. This suggests that there may be a correlation between the amount of the loan and its status, as well as the rate. The payment amount does not suggest any correlation, and the debt to income ratio suggests perhaps a small correlation.



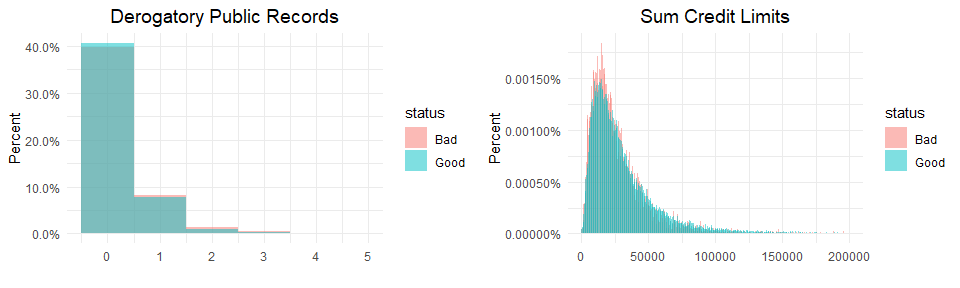
Based on these plots, the good loans seem to have a higher balance. There were more credit inquiries and number of accounts for bad loans, with approximately the same number of open accounts.



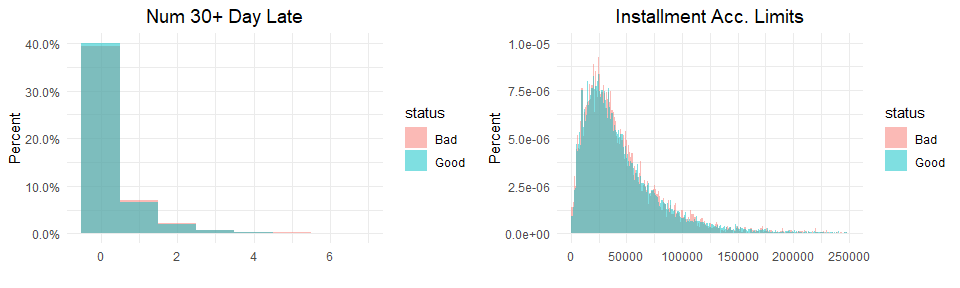
The income is heavily skewed, so it is displayed using a log scale. The plot shows a higher income for good borrowers compared with bad. The credit ratio is slightly higher for those with bad loans.

Judging from these plots, there do not appear to be any strong numeric predictors for bad loans among the variables examined.

Some of the numeric variables have skewed distributions, as well as many zero values, making them unsuitable for boxplots, so histograms can be used instead.



The above plots show that the derogatory public remarks field has a similar, but not identical, distribution between good and bad borrowers. About 40% of borrowers have no remarks, around 8% have a single remark for both statuses, with a much smaller number higher than this. There are a higher proportion of bad borrowers than good with more than one remark. The distributions of the credit limits show that the bad statuses peak earlier in amount.

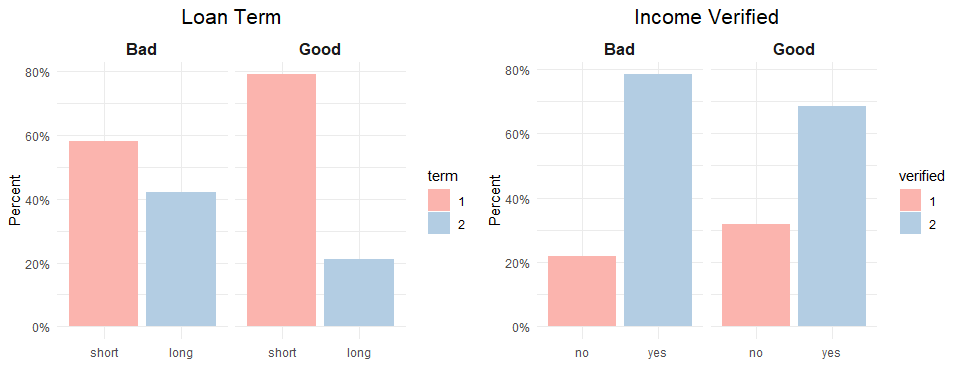


The number of 30+ day late payments is similar, with the bad borrowers having a higher proportion of missed payments for those with at least one. The installment account limits also look very similar.

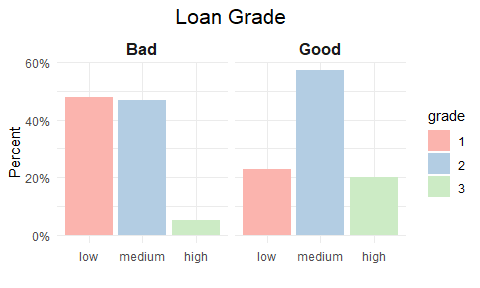
Though some differences are evident based on loan status, they are minor, and we again conclude that none of the numeric predictors appear to show strong correlation.

### Categorical Variables

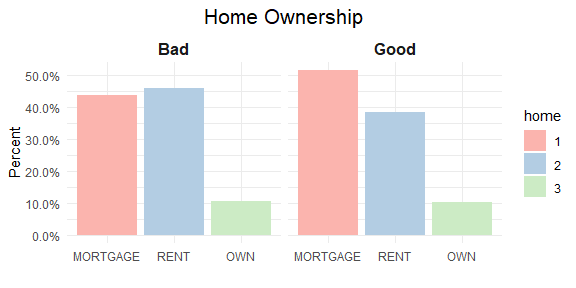
Bar charts were used to compare the categorical data by status, using a proportion rather than totals for ease of interpretation.



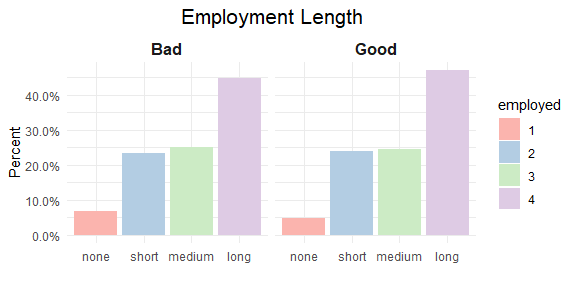
The above plots show that a higher proportion of bad loans have a long term compared to the good loans, and more bad loans have a verified income statement than the good loans.



Very few of the bad loans have a high grade and most are low grade, while most of the good loans have a medium grade.



A slightly higher proportion of the good borrowers have a mortgage as compared with the bad. More of the bad borrowers rent than the good ones.



The bar graph for employment length shows very little difference between the statuses, aside from there being more *NA* values in the bad loans, with the meaning of this being unknown to us, so this field will be dropped from the set of predictors.

The situation for the categorical variables is better than for the numeric ones, with several of them showing differentiation in values between the good and bad loans.

### Results of Data Exploration

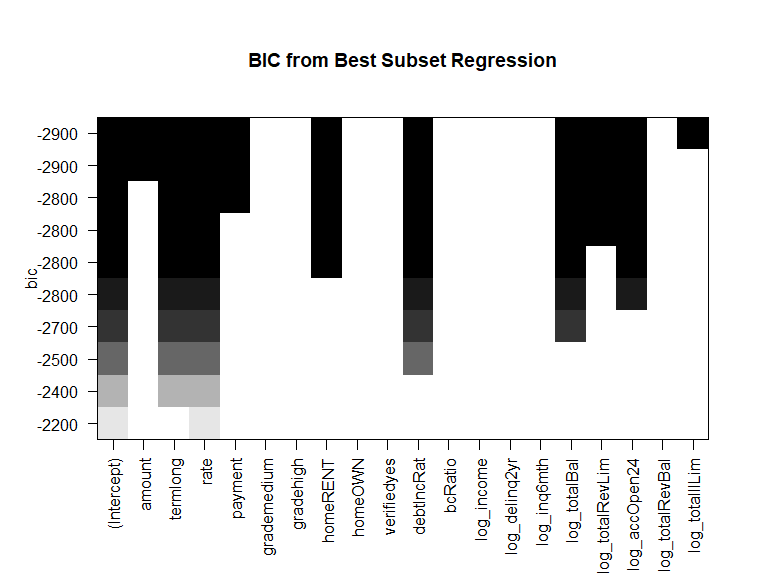
Based on data exploration, ten of the original thirty predictor were dropped, and eleven of the numeric variables were log scaled. One of the factors was dropped due to lack of difference in proportions between loan statuses. None of the numeric predictors appeared to have strong correlations with loan status, and some of the categorical variables showed weak to moderate correlation based on their graphs.

## The Logistic Model

The loan status will be predicted using logistic regression with “1” denoting “Good” and “0” encoding “Bad.” We will initially categorize as “Good” those records with greater than 0.5 predicted probability of being in this category. We will use 80% of the data for training and the rest for testing, which results in 27416 records in the training set and 6855 for testing.

After constructing the initial model, two additional fields, the log-transformed versions of *openAcc* and *pubRec*, were dropped due to their high p-values. The final model was created without these variables included.

We then used the best subset regression technique to reduce the number of variables in the model. We will specify 10 variables as the max, which seems relatively parsimonious given that we began the analysis with 30 potential predictors. This shows the Bayesian information criterion (BIC) for those variables which were used in the best model, where a lower value is preferred.



The variables with no entries in their bins are the ones that were discarded by the selection.

The final model has the following coefficients.

## Estimate Std. Error z value Pr(>|z|)  
## (Intercept) 7.009583e-01 2.916184e-01 2.403683 1.623083e-02  
## amount 2.956243e-05 1.351704e-05 2.187049 2.873897e-02  
## termlong -8.504208e-01 8.779730e-02 -9.686184 3.451841e-22  
## rate -8.625828e+00 5.269407e-01 -16.369637 3.150866e-60  
## payment -1.246547e-03 4.276363e-04 -2.914971 3.557220e-03  
## homeRENT -2.726384e-01 4.115198e-02 -6.625159 3.468746e-11  
## homeOWN -8.806821e-02 5.615919e-02 -1.568189 1.168371e-01  
## debtIncRat -3.120312e-02 2.094250e-03 -14.899422 3.324441e-50  
## log\_totalBal 1.009440e-01 1.866489e-02 5.408230 6.365073e-08  
## log\_totalRevLim 1.997201e-01 2.520574e-02 7.923595 2.307410e-15  
## log\_accOpen24 -3.399265e-01 2.494691e-02 -13.625998 2.805501e-42  
## log\_totalIlLim 2.514997e-02 5.732741e-03 4.387075 1.148850e-05

All of the p-values are below alpha of 0.05, though only a few of the coefficient estimates seem particularly significant.

We used the best model to predict on the training data, which produces the following confusion matrix where “1” denotes a “Good” status and “0” is “Bad.”

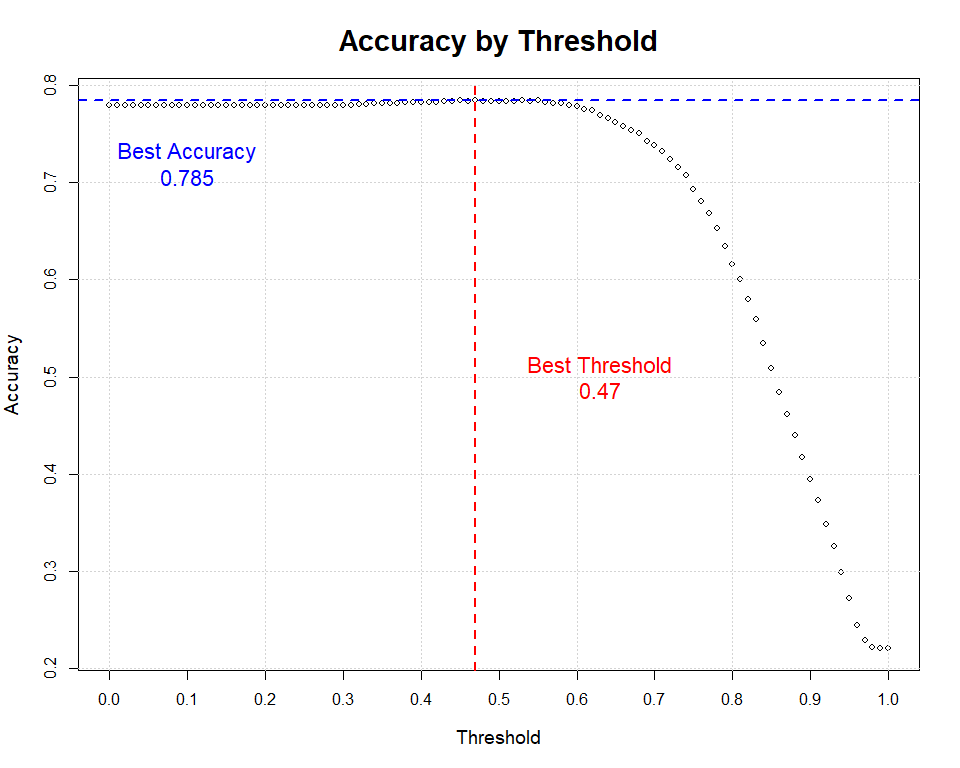
## Reference  
## Prediction 0 1  
## 0 181 149  
## 1 1331 5194

The overall accuracy of our best model derived from this table is 78 percent, which includes all of the records in the test data. The model does a poor job of identifying bad loans, achieving an accuracy of only 12 percent. However, the accuracy for predicting good loans is 97, which is far better than for the bad ones.

We conclude that at the standard threshold of 0.5, the model is not accurate at predicting bad loans, as it misclassified most them as good, but it performs much better at correctly predicting good loans.

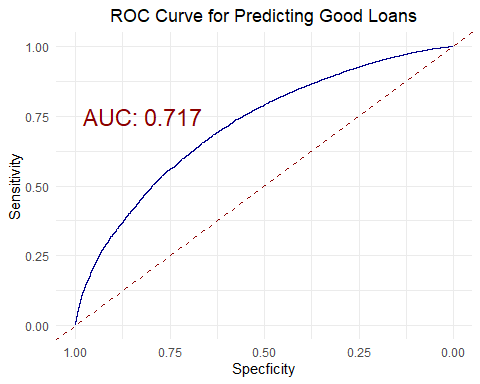
## Optimizing the Threshold for Accuracy

We benchmarked the model on the test data based on accuracy, scanning through probability thresholds from 0.0, in which all loans were classified as good, to 1.0, which essentially rejects all of them as bad. This results in an optimal threshold for accuracy of 0.47 with an overall accuracy of 0.7875 or approximately 79 percent.



Since most of the loans are good, the accuracy is initially high with an optimal threshold fairly close to 0.5. The accuracy falls off as the threshold increases to 1.0, which results in all of the loans being rejected as bad.

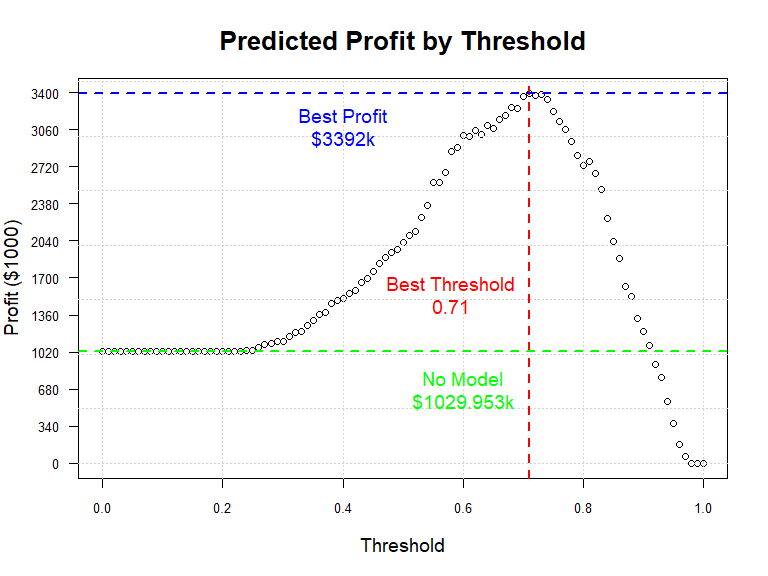
A ROC curve illustrates the trade-off between sensitivity (true positives) and specificity (true negatives) for our model.



This displays a smooth curve due to the nature of the linear model, and the AUC around 0.71 does not seem very impressive.

## Optimizing the Threshold for Profit

We also tuned the probability threshold based on profit, the results of which are shown below.



The profit starts out flat because the lowest predicted probability is around 0.23, so setting the threshold lower than this makes no difference. The profit then increases to the optimal threshold and finally falls to zero as more loans are rejected until none pass.

The profit assuming all loans are accepted is $1,029,953 for the test data. The calculated profit from our best model is $3,391,945, an increase of 329 percent compared with using no model. A perfect identification of loan status would result in a profit of $12,334,047, or 364 percent greater profits compared to using our model. While a perfect model would provide by far the best results, the model we created results in significantly greater profitability compared to none at all.

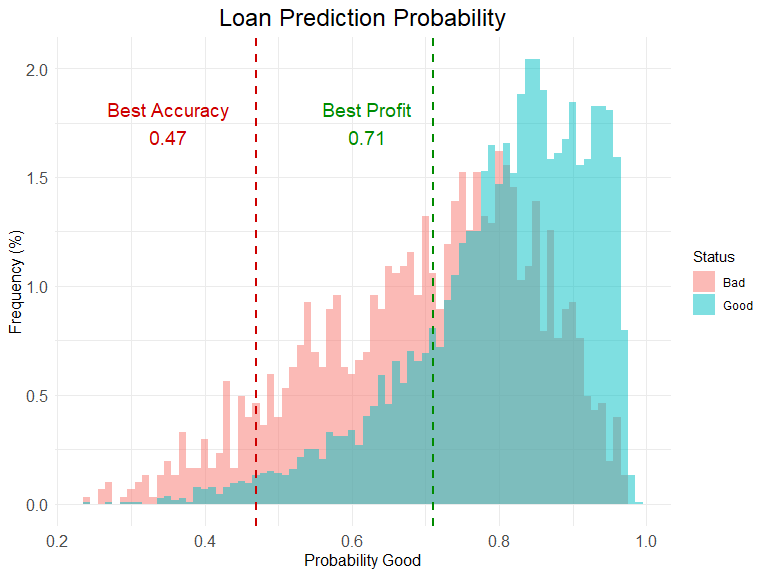
The maximum profit threshold does not coincide with the the maximum accuracy threshold. Rather, it is much higher. This is probably because the threshold that is most accurate accepts too many bad loans, which considerably affects the profit.

## Results Summary

A threshold of 0.71 for classifying a loan as good yields the highest profit. We summarize the results of this model as follows.

| Rate | Value | Description |
| --- | --- | --- |
| Sensitivity | 0.9828 | When good, how often predicts good? |
| Specificity | 0.0853 | When bad, how often predicts bad? |
| Pos Pred Value | 0.7915 | Rate of good amongst good predicted |
| Neg Pred Value | 0.5837 | Rate of bad amongst bad predicted |
| Precision | 0.7915 | When predicts good, how often correct? |
| Recall | 0.9828 | True positive rate |
| F1 | 0.8768 | Weighted avg of recall and precision |
| Prevalence | 0.7794 | How often does good condition occur? |
| Detection Rate | 0.7660 | Proportion of tot sample where outcome was detected correctly |
| Detection Prevalence | 0.9678 | Correct predictions of good as a proportion of tot |
| Balanced Accuracy | 0.5340 | Balanced accuracy measure using sensitivity and specificity |

Since the prediction accuracy for bad loans is quite poor, we will explore this further in order to understand why this might be the case. Even at the probability threshold with the best profit, which is fairly high, many of the bad loans are predicted as being good. This can be illustrated by the following histogram, which shows considerable overlap between the two categories.



The most profitable threshold rejects a large number of bad loans, many of which occur in the band between the most accurate and most profitable thresholds. As the threshold increases above this, the profit falls as too many good loans are dropped, even though additional bad ones are rejected. We observe that there are a large number of bad loans which were assigned a high probability of being good, with this distribution peaking around 0.8, well above the typical cutoff of 0.5. The characteristics of the probability distributions suggest that the predictors in the model do not have much power to identify the bad loans.

In conclusion, the model has high accuracy in predicting good loans, but it is poor at identifying bad ones. This means that while it is better than no model at all, it is considerably worse compared with a theoretically perfect model, due to its low rejection rate of bad loans. But even this model yields a considerable increase in profits, based on the fact that many of the worst performing loans can still be accurately identified and rejected. We conclude that even though the model has issues with flagging bad loans, it is still useful compared with using no model at all. We were able to build a model which identifies bad loan status accurately enough to achieve a significant increase in profits, and so the goal based on our initial research question was achieved.