Stat 652 Project

Blake Shurtz

February 6, 2019

The goal of the project is to predict the **loan_status** of a given loan. A background introduction to the data set is provided. An exploratory analysis of the target variable yields a highly accurate null model of a subset of the data, which also reduces the remaining levels of the target to two. The dimension of the data is reduced through a walkthrough of its 144 features. Statistical and machine learning models are executed on training data and evaluated on test data. Compared to the top model accuracy of 0.841, the models herein have predictive accuracies of between 96-98%.

Introduction

Data on loans comes from the Lending Club website at https://www.lendingclub.com/info/download-data.action. Data for the years 2012-2014 were downloaded, merged and cleaned. For a record of this process, see the rPub here.

Table 1: Dimension of Data

Dimension	Value
Rows	423812
Columns	145

Exploratory Analysis

Table 2: Target Levels

Level	Freq	Description
Charged Off	70613	*The original creditor has given up on being repaid according to the terms of the loan*
Current	14585	*The loan is open and is being re-paid*
Default	1	*The loan won't be paid*
Fully Paid	337704	*The load has been paid*
In Grace Period	362	*The loan is late, < 13 days*
Late (16-30 days)	126	*The next payment is past due by 13-30 days*
Late (31-120 days)	417	*The loan is late by 31-120 days*

The two main categories as represented by their frequency are *Fully Paid* and *Charged-Off*.

The third category by size is *Current*. We'll return to category this momentarily.

The last 3 categories, *In Grace Period*, *Late (13-30 days)*, and *Late (31-120 days)* are chronologically ordered levels between *Current* and *Charged-Off*. If a *Current* status loan is not repaid by the due date, it enters *In Grace Period*, then the two categories of lateness, then after 120 days the loan is categorized as *Charged-Off* then *Default*.

"loan_status = Current" Data Subset

Exploratory analysis indicates that loans classified as *Current* consistently have non-zero values for **out_prncp**, defined as "Remaining outstanding principal for total amount funded". In other words, current loans still have outstanding debt repayments.

As it turns out, almost all (14573/14585) *Current* loans have **out_princp** greater than 0.

Interestingly, only 903/409223 non-*Current* observations also have a non-zero value for this variable. It turns out that the remaining 903 observations are assigned to other categories of late-payers.

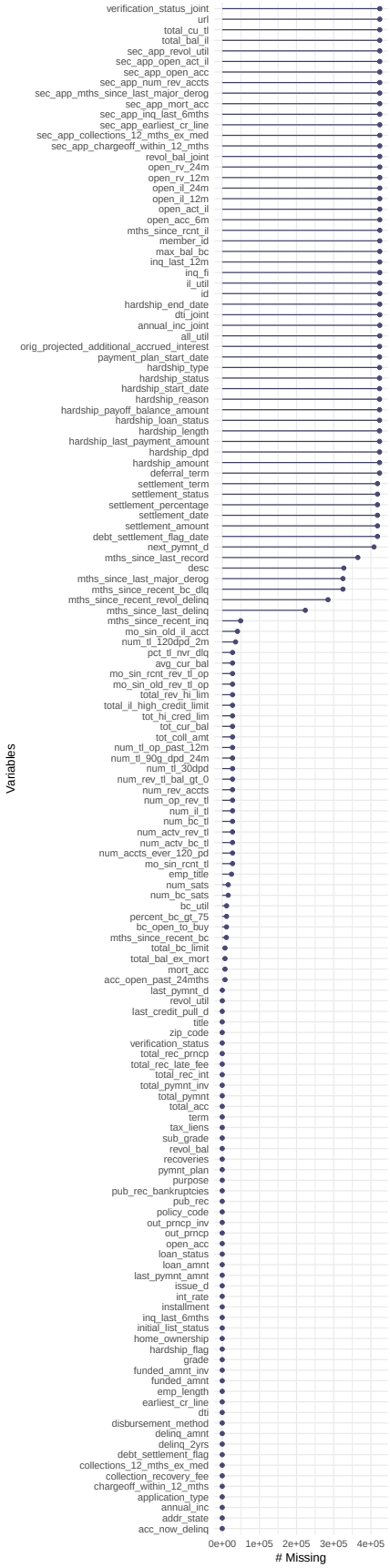
In other words, **out_princ > 0** belongs exclusively to 5 levels of the target. Furthermore, the levels are overwhelmingly represented by the loan class of *Current*.

Therefore, by subsetting the data on the basis of **out_princp > 0** and applying the null model that all observations are *Current*, I can achieve 94% accuracy on the subsetted data (comprising 3% of the total data). This also suggests that I can exclude **out_prncp** from subsequent models.

In addition, by eliminating 5 of the 7 response categories, I'm left with only two categories to focus on: *Charged-Off* and and *Fully Paid.* Thus, the target **loan_status** becomes a binary variable, which will allow me to apply binary classification methods.

Missing Data

Many variables are missing most or all of the data. See the figure on the next page. There is a pretty big drop-off between **mths_since_last_delinq** and **mths_since_recent_inq**, so I'll draw the line there. In all, 58 features were removed.


verification_status_joint
url
total_cu_tl
total_bal_il
sec_app_revol_util
sec_app_open_act_il
sec_app_open_acc
sec_app_num_rev_accts
sec_app_mths_since_last_major_derog
sec_app_mort_acc
sec_app_inq_last_6mths
sec_app_earliest_cr_line
sec_app_collections_12_mths_ex_med
sec_app_chargeoff_within_12_mths
revol_bal_joint
open_rv_24m
open_rv_12m
open_il_24m
open_il_12m
open_act_il
open_acc_6m
mths_since_rcnt_il
member_id
max_bal_bc
inq_last_12m
inq_fi
il_util
id
hardship_end_date
dti_joint
annual_inc_joint
all_util
orig_projected_additional_accrued_interest
payment_plan_start_date
hardship_type
hardship_status
hardship_start_date
hardship_reason
hardship_payoff_balance_amount
hardship_loan_status
hardship_length
hardship_last_payment_amount
hardship_dpd
hardship_amount
deferral_term
settlement_term
settlement_status
settlement_percentage
settlement_date
settlement_amount
debt_settlement_flag_date
next_pymnt_d
mths_since_last_record
desc
mths_since_last_major_derog
mths_since_recent_bc_dlq
mths_since_recent_revol_delinq
mths_since_last_delinq
mths_since_recent_inq
mo_sin_old_il_acct
num_tl_120dpd_2m
pct_tl_nvr_dlq
avg_cur_bal
mo_sin_rcnt_rev_tl_op
mo_sin_old_rev_tl_op
total_rev_hi_lim
total_il_high_credit_limit
tot_hi_cred_lim
tot_cur_bal
tot_coll_amt
num_tl_op_past_12m
num_tl_90g_dpd_24m
num_tl_30dpd
num_rev_tl_bal_gt_0
num_rev_accts
num_op_rev_tl
num_il_tl
num_bc_tl
num_actv_rev_tl
num_actv_bc_tl
num_accts_ever_120_pd
mo_sin_rcnt_tl
emp_title
num_sats
num_bc_sats
bc_util
percent_bc_gt_75
bc_open_to_buy
mths_since_recent_bc
total_bc_limit
total_bal_ex_mort
mort_acc
acc_open_past_24mths
last_pymnt_d
revol_util
last_credit_pull_d
title
zip_code
verification_status
total_rec_prncp
total_rec_late_fee
total_rec_int
total_pymnt_inv
total_pymnt
total_acc
term
tax_liens
sub_grade
revol_bal
recoveries
pymnt_plan
purpose
pub_rec_bankruptcies
pub_rec
policy_code
out_prncp_inv
out_prncp
open_acc
loan_status
loan_amnt
last_pymnt_amnt
issue_d
int_rate
installment
inq_last_6mths
initial_list_status
home_ownership
hardship_flag
grade
funded_amnt_inv
funded_amnt
emp_length
earliest_cr_line
dti
disbursement_method
delinq_amnt
delinq_2yrs
debt_settlement_flag
collections_12_mths_ex_med
collection_recovery_fee
chargeoff_within_12_mths
application_type
annual_inc
addr_state
acc_now_delinq
Variables
0e+00
1e+05
2e+05
3e+05
4e+05
Missing

Dimension Reduction

Before building the models, I have to reduce the number of features in my data. Otherwise, the ML models will take too long to compile.

Long Right-Tailed Variables

There are a number of long right-tailed variables in the data set, which I define as a variable where the vast majority of the observations are 0, with a right-tail distribution that comprises around 5%-10% of the remaining observations.

The correlation between each of these features and the target were measured. 11 features with correlations of < 0.05 were removed.

Highly Correlated Predictors

Just as there are features that are correlated with the target, there are also groups of features that are highly correlated with one another. The dimension of the data can be reduced by removing highly correlated predictors while maintaining the number of unique groups.

Four features can be removed: the **term** is reflected by the **int_rate**, so I'll remove **term**. **grade** and **sub_grade** are highly correlated, I'll remove **sub_grade**. **loan_amnt**/**funded_amnt**/**funded_amnt_inv** are similar, so I'll remove the latter two.

Date Variables

There are four time series variables in the data set: **issue_d**, **earliest_cr_line**, **last_credit_pull_d** and **last_pymnt_d**. While time series variables can also be useful in an exploratory analysis, none are correlated with the target.

Location Variables

There are two variables that have data on the location of the loan: **zip_code** and **addr_state**, both of which are factors that have 50+ levels. The high number of levels will greatly slowdown the ML algorithm, so they will be excluded.

What Remaining Features (Probably) Don't Matter?

The remaining features were checked against a data dictionary to decide a priori whether or not they are predictive of the target. While remaining open to surprises, the application of reasonable doubt to the features in question is a way to improve model performance with a low hypothesized Type 1 error. When in doubt, correlations were checked.

acc_open_past_24mths: Number of trades opened in past 24 months **Keep**
annual_inc: Self-reported income **Keep**
avg_cur_bal: How much is owed. Debt. **Keep**
bc_open_to_buy: Seems important to retail, not sure why it's here. **Drop**
bc_util: Current balance/credit limit. May be indicative of debt. **Keep**
collection_recovery_fee: Record of a small collection fee. Moderate correlation. **Keep**
debt_settlement_flag: Working with a debt-settlement company. Moderate correlation. **Keep**
dti: Debt-to-income **Keep**
initial_list_status: Whole or fractional loan. Weak correlation. **Drop**
installment: Monthly amount owed. Weak correlation. **Drop**
last_pymnt_amnt: Moderate correlation. **Keep**
mo_sin_old_xxx: Indicative of financial status at time of loan. NA correlation. **Drop**
mort_acc: Number of mortgage accounts. **Keep**
id and **member_id** are not predictors. **Drop emp_length** (employment length) has been tidied. **Keep*

Training and Test Data

I will divide the data up using the *createDataPartition* function in the **caret** package, "a series of test/training partitions are created using createDataPartition while createResample creates one or more bootstrap samples." (CRAN) The training data comprises 75% of the original data set.

```
set.seed(1234)
dindex <- createDataPartition(d$loan_status, p=.75, list=FALSE)
train <- d[dindex,]; test <- d[-dindex,]
```

Step 3: Training a model on the data

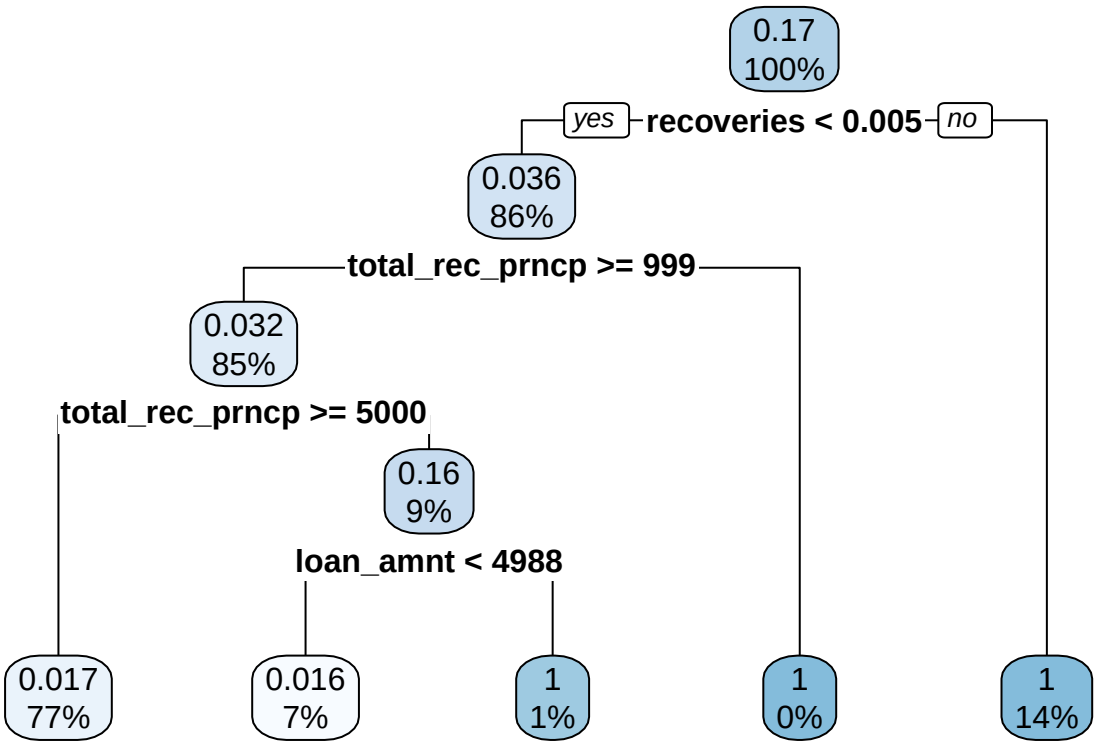
Model 0: Null Model

First, I'll impose the null model by calling 5/7 classifiers *Current*. Subsequent models are conditional on this subset of the data. The remaining subset is called below:

```
traincat <- train %>% filter(loan_status == "Charged Off" | loan_status == "Fully Paid")
traincat$loan_status <- as.numeric(traincat$loan_status) #recoding
traincat$loan_status <- ifelse(traincat$loan_status==1, 1, 0) #0 is paid, 1 is charged
```

Model 1: Classification Tree

The classification tree model using recursive partitioning executed properly with all of the features.



The model mistakenly reports *Fully Paid* predictions as slightly greater than 0. Other than than, **recoveries** (which has a correlation of 0.5 with the outcome) is the sole predictor of whether someone has charged off, that is, has not paid their loan.

recoveries is defined as "post charge off gross recovery," suggesting that is represents a certain amount of written-off debt. This makes a lot of sense. No loans that have been fully paid would need has a gross recovery, and loans that have been charged off would have some partial forgiveness.

The receipt of payments on the principal **total_rec_prncp** indicates that a loan is paid off. In short, paying principal is the best predictor that the loan will become *Fully Paid*.

Model 2: Logistic Regression

A logistic regression works as a method of classification. I tried to fit as many remaining predictors as possible, while still allowing the model to converge.

```
logit.mod <- glm(loan_status ~
            ###low correlation
            verification_status  + acc_open_past_24mths +
            avg_cur_bal + mo_sin_rcnt_rev_tl_op + int_rate + grade + emp_length +
            num_accts_ever_120_pd + num_actv_bc_tl + num_actv_rev_tl + num_bc_sats +
            num_bc_tl + pct_tl_nvr_dlq + loan_amnt +
            ###moderate correlation
            debt_settlement_flag #(r= 0.27)
            , data=traincat, family = binomial)
```

Four moderately correlated predictors lead the model to overfit: **recoveries** (0.53), **last_pymnt_amnt** (-.29), **collection_recovery_fee** (0.46), **total_rec_prncp**(-0.39). R gives the warning: "glm.fit: fitted probabilities numerically 0 or 1 occurred."

Attempts to add additional weaker predictors to the model caused the model to fail to converge and exclude observations: **home_ownership + annual_inc + purpose + dti + inq_last_6mths + open_acc + revol_bal + revol_util + total_acc + total_pymnt + total_pymnt_inv + total_rec_int + total_rec_late_fee + tot_coll_amt + tot_cur_bal + total_rev_hi_lim + total_bc_limit + total_il_high_credit_limit + num_il_tl + num_op_rev_tl + num_rev_accts + num_rev_tl_bal_gt_0 + mo_sin_rcnt_tl + mths_since_recent_bc + mths_since_recent_inq + percent_bc_gt_75 + tot_hi_cred_lim + num_sats + num_tl_120dpd_2m + num_tl_30dpd + num_tl_90g_dpd_24m + num_tl_op_past_12m + total_bal_ex_mort.**

Model 3: Logistic (Bayesian)

Given the failure of the logit model to include all moderately correlated predictors, a Bayesian logistic model with adaptive priors will be introduced.

There were two executions of the model. The first execution ran 1000 iterations on the full training data set and the second ran 500 iterations on 25% of the training data (in order to decrease execution time).

```
#wrangle
traincatsmall$debt_settlement_flag <- as.numeric(traincatsmall$debt_settlement_flag)
set.seed(1234)
options(mc.cores = parallel::detectCores())
Sys.setenv(LOCAL_CPPFLAGS = '-march=native')
logit.bayes.mod <- map2stan(
    alist(
      loan_status ~ dbinom( 1 , p ) ,
      logit(p) <- a + b * debt_settlement_flag + c * recoveries +
                  d * last_pymnt_amnt + e * collection_recovery_fee +
                  f * total_rec_prncp,
      #adaptive priors
      a ~ dnorm( 0 , 1.5 ),
      b ~ dnorm(b_mu, b_sigma),
      b_mu ~ dnorm(0,1),
      b_sigma ~ dcauchy(0,2),
      c ~ dnorm(c_mu, c_sigma),
      c_mu ~ dnorm(0,1),
      c_sigma ~ dcauchy(0,2),
      d ~ dnorm(d_mu, d_sigma),
      d_mu ~ dnorm(0,1),
      d_sigma ~ dcauchy(0,2),
      e ~ dnorm(e_mu, e_sigma),
      e_mu ~ dnorm(0,1),
      e_sigma ~ dcauchy(0,2),
      f ~ dnorm(f_mu, f_sigma),
      f_mu ~ dnorm(0,1),
      f_sigma ~ dcauchy(0,2)
) , data=traincat,
 iter=1000, warmup=250, chains=1, cores=4)
```

Model convergence was poor with a large number of divergent samples. However, coefficients were calculated for all predictors. The regression model is below:

$log\frac{p}{1-p} = 0.4756 - 0.762 * debtsettlementflag + 2.434 * recoveries - 0.0008 * lastpaymentamount - 1.641 * collectionrecoveryfee - 0.0002 * totalrecprncp$

Step 4: Evaluating Model Performance

Firstly, I have to subset the data in order to apply the null model **loan_status =** *Current* to 5/7 levels. For each model evaluation, I will add this subset back into the final analysis after running the remaining data through the algorithm.

Given that there are more than two levels of the outcome, accuracy is the preferred method of evaluation, along with confusion matricies. Note that for confusion matricies, the level 3 *Default (n=1)* was excluded. As a result of the programming, the level *Fully Paid* may be represented by either a 0 or a 4.

Null Model Evaluation

After the data set is subsetted, a null model would assume that all loans are *Fully Paid*. This leads to an overall accuracy of 83.1%.

CART Model Evaluation

The CART model has an overall accuracy of 98%.

Table 3: Confusion Matrix- CART Model

	1	2	4	5	6	7
1	16179	0	0	0	0	0
2	0	3646	0	90	31	104
4	1474	0	84425	0	0	0

Logit Evaluation

The logit model has an overall accuracy of 79%.

Table 4: Confusion Matrix- Logit Model

	0	1	2	5	6	7
0	78532	14987	0	0	0	0
1	43	1528	0	0	0	0
2	0	0	3646	90	31	104

Logit Bayes Evaluation

The logit Bayes model had an accuracy of 84% and 96%. The model that executed on a larger proportion of the training data with more iterations was more accurate.

Table 5: Confusion Matrix- Bayes Logit

	0	1	2	5	6	7
0	84426	3544	0	0	0	0
1	0	2917	0	0	0	0
2	0	0	3646	90	31	104

Conclusion

On the entryway to Bruce Lee's dojo, there was a sign that read: "Reject what is useless. Accept what is useful." Bruce Lee is remembered as a martial artist who judged his ability as a fighter against all possibilities, ie. randomness. He synthesized what worked and what didn't into his own system of classification.

There are many models that I could have used in this project. But, because I have achieved 95%+ accuracy with both a statistical learning and machine learning algorithms, I will consider the assignment a success. I also put a lot of credit on the model success on a thourough exploratory analysis.