What Do I Invest In?

April 7, 2019

Executive Summary

Managing personal finances is a skill that not many college students are taught; finance classes usually teach students how to manage other people's money, but very rarely their own. Therefore, the motivation behind this analysis is to help individuals who are interested in adding loans to their personal portfolios, but don't really know how to choose between the options. Utilizing supervised learning techniques, I build a model that will guide investors in choosing investments that maximize their returns. The rule is to maximize portfolio expected return as defined as:

$$E(R_p) = \sum_{i=0}^{n} R_i P(\text{loan i is fully paid})$$

subject to the investor's capital constraints, where n is the number of loans in the portfolio. For the purpose of this analysis, I will use the interest rate as the rate of return.

Even if loans are paid off, some may be paid off early/late. Investors may want borrowers to pay off their loans late as they will accumulate more interest. Even accounting for the time value of money, an investor makes more money when borrowers take longer to pay because the lowest interest rate the Lending Club offers is 6.46% whereas the risk free rate used to discount investments is about 3%. So, assuming the investor does not need the money immediately, it is ideal for the borrowers to pay off their loans later. Therefore, I will also build a model that predicts how early or late a borrower will repay their loan and hence, the final rule is to maximize portfolio expected return as defined as:

$$E(R_p) = \sum_{i=0}^{n} R_i P(\text{loan i is fully paid}) + R_i T_i$$

where T: the amount of time difference between loan i's expiration date and the last date payment was received, on an annualized basis. To appeal to the investors who don't want to think in terms of probabilities, I also categorize the loans into those that will default and those that will not.

Based on the analysis, the probability that a loan will default can be best explained by the term, interest rate, annual income, purpose, number of inquiries in the last 6 months, number of derogatory public records, revolving balance and revolving utilization. Furthermore, I was able to identify a probability threshold of 0.14 for classifying the loans into those that will default and those that will not, which yielded a 0.14 misclassification rate. Lastly, the model that predicted how early/late a loan was fully paid back identified term, dti, revolving utilization, loan amount * term, term * interest rate, term * verification status, term * inquiries in the last 6 months, term * total account, dti * open account, open account * revolving utilization as the most important factors. However, although this model performed relatively well, the testing error is 391.5 days, which is more than a year!

One of the main issues that I faced throughout the process was that my RStudio kept crashing, so I could not conduct computationally-heavy processes. Furthermore, since there were more loans that were fully paid back than loans that defaulted, the model may be slightly biased because there was not a good proportion of both outcomes. There was also autocorrelation between the variables, so moving forward, analyses could account for that.

Analysis

Data Summary

The data obtained from the Lending Club contained 38,971 observations and 39 variables between the years 2007-2011. I only considered data from this period as loans made after 2011 may not be closed out yet. Even though the dataset had 39 variables, I could not use all of them as including post-loan data is unrealistic to predict the default rate before making the investment! The available variables were:

Pre-funded loan data

(a) loan_amnt: The listed amount of the loan applied for by the borrower
(b) int_rate: Interest rate on the loan
(c) grade: LC assigned loan grade
(d) sub_grade: LC assigned loan subgrade
(e) installment: The monthly payment owed by the borrower if the loan originates
(f) purpose: The monthly payment owed by the borrower if the loan originates.
(g) term: The number of payments on the loan. Values are in months and can be either 36 or 60.

Borrower basic information

(a) emp_title: The job title supplied by the Borrower when applying for the loan
(b) emp_length: Employment length in years. Possible values are between 0 and 10 where 0 means less than one year and 10 means ten or more years.
(c) home_ownership: The home ownership status provided by the borrower during registration or obtained from the credit report. Our values are: RENT, OWN, MORTGAGE, OTHER.
(d) annual_inc: The self-reported annual income provided by the borrower during registration
(e) zip_code: The first 3 numbers of the zip code provided by the borrower in the loan application
(f) addr_state: The state provided by the borrower in the loan application
(g) verification_status: Indicates if income was verified by LC, not verified, or if the income source was verified

Borrower credit data

(a) dti: A ratio calculated using the borrower's total monthly debt payments on the total debt obligations, excluding mortgage and the requested LC loan, divided by the borrower's self-reported monthly income.
(b) delinq_2yrs: The number of 30+ days past-due incidences of delinquency in the borrower's credit file for the past 2 years
(c) earliest_cr_line: The month the borrower's earliest reported credit line was opened
(d) inq_last_6mths: The number of inquiries in past 6 months (excluding auto and mortgage inquiries)
(e) open_acc: The number of open credit lines in the borrower's credit file.
(f) pub_rec: Number of derogatory public records
(g) revol_bal: Total credit revolving balance
(h) revol_util: Revolving line utilization rate, or the amount of credit the borrower is using relative to all available revolving credit.
(i) total_acc: The total number of credit lines currently in the borrower's credit file
(j) pub_rec_bankruptcies: Number of public record bankruptcies

Post loan data

(a) issue_d: The month which the loan was funded monthly income.
(b) loan_status: Current status of the loan

(c) funded_amnt: The total amount committed to that loan at that point in time.
(d) funded_amnt_inv: The total amount committed by investors for that loan at that point in time
(e) total_pymnt: Payments received to date for total amount funded
(f) total_pymnt_inv: Payments received to date for portion of total amount funded by investors
(g) total_rec_prncp: Principal received to date
(h) total_rec_int: Interest received to date
(i) total_rec_late_fee: Late fees received to date
(j) recoveries: post charge off gross recovery
(k) collection_recovery_fee: post charge off collection fee
(l) last_pymnt_d: Last month payment was received
(m) last_pymnt_amnt: Last total payment amount received
(n) last_credit_pull_d: The most recent month LC pulled credit for this loan

Data cleaning: I converted all the dates into R's date format and utilized Regex to convert the terms to integers to ease calculations. I also encoded the status of the loans to 0: paid off and 1: charged off. The data only contained loans that were fully paid and charged off (defaulted and there is no longer a reasonable expectation of further payments). For the purpose of this analysis, I will use "default" and "charged off" interchangeably. Lastly, the employment length had a lot of n/a variables, so I removed employment length from the analysis.

Data manipulation: In order to determine if the loans were paid back early or late, I found the difference in time between the date the loan was issued and the last date payment was received, for loans that were fully paid.

Final data: The goals of this analysis were (1) predict the probability a loan will default, (2) predict if a loan will default and (3) predict when the fully paid loans will be paid off. The final dataframe used to answer (1) and (2) had 38,971 observations with 23 explanatory variables and an indicator variable to indicate if the loan defaulted. The final dataframe used to answer (3) had 33,503 observations with 20 explanatory variables and the number of days between when the loan was paid off and when the term ended.

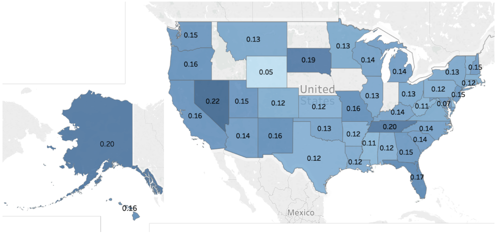


Figure 1: Proportion of charged off loans per state

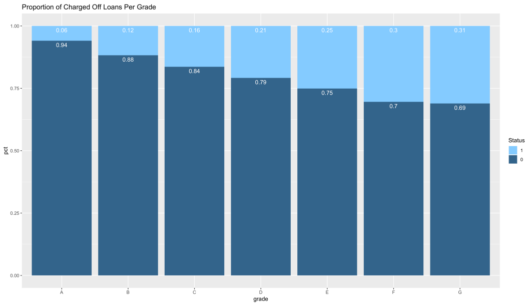


Figure 2: Proportion of charged off loans per grade

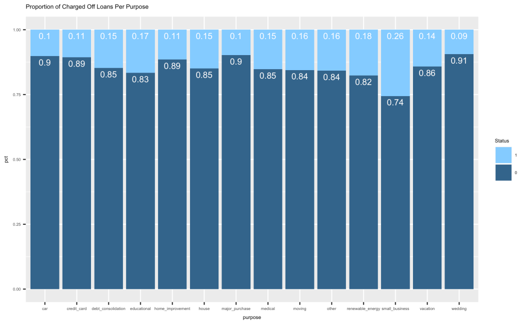


Figure 3: Proportion of charged off loans per purpose

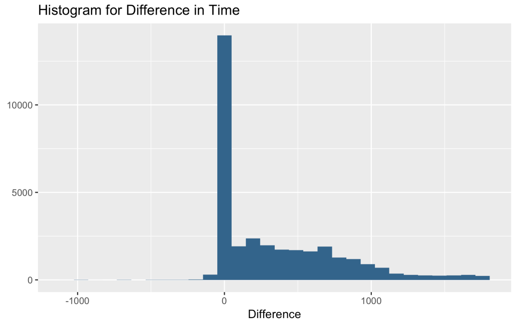


Figure 4: Histogram of how early/late loans were paid off

Models

I used a 70/30 train-test split for every model.

What is the probability a loan will default?

Logistic Regression 1:

Based on prior knowledge, I decided to fit a logistic regression using variables that I thought would be best at explaining the probability that a loan would default. The variables were grade, purpose and inquiries within the last 6 months. This model yielded a testing root mean squared error of 0.3399.

Logistic Regression 2 (LASSO):

Deciding to take a more systematic approach, I utilized 10-fold cross-validation on the training data to build a logistic regression model that produced the minimum binomial deviance. Usually, the full dataset is used when cross-validation is utilized, but I wanted to compare every model's performance using the same testing dataset, and felt it would be erroneous to include the test data in the training process, even if cross-validation was used.

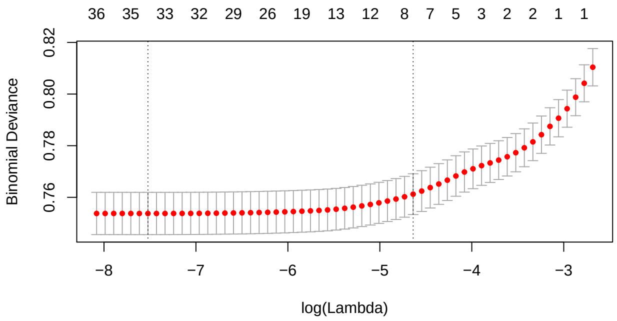


Figure 5: Plot of binomial deviance for first LASSO model

The final variables in this model were loan amount, term, interest rate, grade, home ownership, annual income, verification status, purpose, monthly debt payments to total debt ratio, number of delinquent records in the past 2 years, number of inquiries in the last 6 months, open accounts, public records, revolving balance, revolving utilization, total accounts, number of public record bankruptcies. Using an Anova() test, I found that not all the variables were significant at the α = 0.05 level, so I removed them in the next model. The testing error was 0.3373, an improvement from the previous model.

Logistic Regression 3 (LASSO):

Improving on the previous model, I removed loan amount, grade, home ownership, verification status, monthly debt payments to total debt ratio, number of delinquent records in the past 2 years, open accounts and number of public record bankruptcies as they were not significant at the $\alpha =$ 0.05 level. This resulted in a model with 9 predictor variables. Conducting another Anova() test, all the variables except for total account were significant. The testing error was 0.3373, with no improvement from the previous model.

Logistic Regression 4 (LASSO):

Building on the previous model, I removed total account. The final model's variables were significant at the $\alpha =$ 0.05 level and produced a test error of 0.3372.

Random Forest Regressor:

I decided to use the random forest regressor instead of classifier as I was more interested in the probability that a borrower would default. I built random forests with 300, 400, 500, 600, 700 trees and found that the forest with 300 trees yielded the lowest test misclassification error, 0.3386. The top 5 variables that had an effect on this model were annual income, revolving utilization, revolving balance, dti and interest rate.

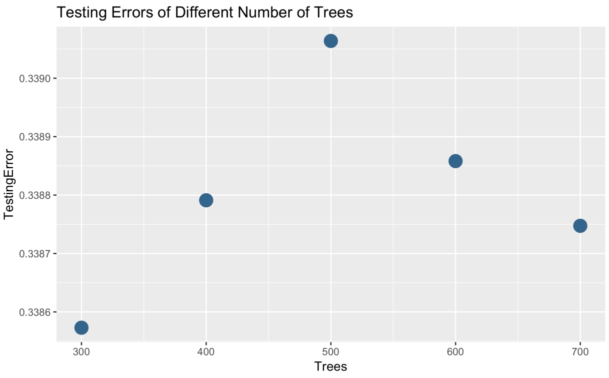


Figure 6: Test Misclassification Errors for Different ntrees

Final Model:

By comparing the test error of the models, the fourth logistic regression model produced the lowest rate. The probability that a loan defaults can be determined by the term, interest rate, annual income, purpose, inquiries in the last 6 months, number of public records, revolving balance and revolving utilization. Each of these variables has a positive impact on the log odds of defaulting, except for annual income and loans for weddings. Interest rate has the highest marginal effect.

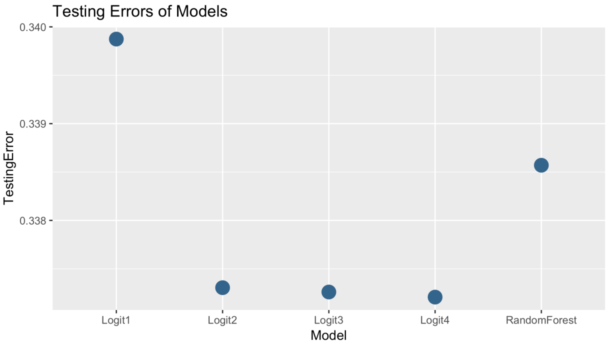
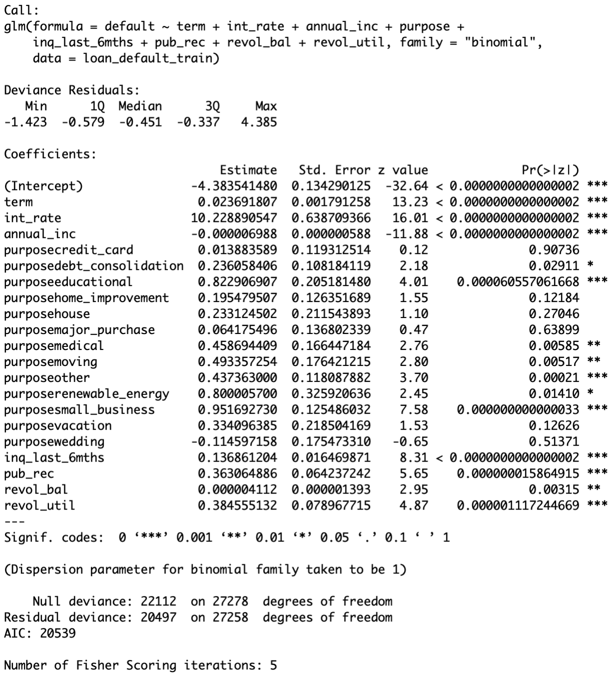


Figure 7: Test Misclassification Errors for the Models

The final model was:



```
Call:
glm(formula = default ~ term + int_rate + annual_inc + purpose +
    inq_last_6mths + pub_rec + revol_bal + revol_util, family = "binomial",
    data = loan_default_train)

Deviance Residuals:
   Min      1Q  Median      3Q     Max
-1.423  -0.579  -0.451  -0.337   4.385

Coefficients:
                             Estimate  Std. Error z value               Pr(>|z|)
(Intercept)              -4.383541480 0.134290125  -32.64 < 0.0000000000000002 ***
term                      0.023691807 0.001791258   13.23 < 0.0000000000000002 ***
int_rate                 10.228890547 0.638709366   16.01 < 0.0000000000000002 ***
annual_inc               -0.000006988 0.000000588  -11.88 < 0.0000000000000002 ***
purposecredit_card        0.013883589 0.119312514    0.12                0.90736
purposedebt_consolidation 0.236058406 0.108184119    2.18                0.02911 *
purposeeducational        0.822906907 0.205181480    4.01    0.000060557061668 ***
purposehome_improvement   0.195479507 0.126351689    1.55                0.12184
purposehouse              0.233124502 0.211543893    1.10                0.27046
purposemajor_purchase     0.064175496 0.136802339    0.47                0.63899
purposemedical            0.458694409 0.166447184    2.76                0.00585 **
purposemoving             0.493357254 0.176421215    2.80                0.00517 **
purposeother              0.437363000 0.118087882    3.70                0.00021 ***
purposerenewable_energy   0.800005700 0.325920636    2.45                0.01410 *
purposesmall_business     0.951692730 0.125486032    7.58    0.000000000000033 ***
purposevacation           0.334096385 0.218504169    1.53                0.12626
purposewedding           -0.114597158 0.175473310   -0.65                0.51371
inq_last_6mths            0.136861204 0.016469871    8.31 < 0.0000000000000002 ***
pub_rec                   0.363064886 0.064237242    5.65    0.000000015864915 ***
revol_bal                 0.000004112 0.000001393    2.95                0.00315 **
revol_util                0.384555132 0.078967715    4.87    0.000001117244669 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 22112  on 27278  degrees of freedom
Residual deviance: 20497  on 27258  degrees of freedom
AIC: 20539

Number of Fisher Scoring iterations: 5
```

Which loans will default?

For those who don't want to think of probabilities, I categorized the loans. As an investor, I want to minimize my false negative rate because it's better to not invest in a loan that I think would not default but actually defaults. However, at the same time, I want a false positive rate that isn't too high because I don't want to miss out loans that will not default. Using the final model from the previous part, I determined the threshold that balances the false negative and false positive rates.

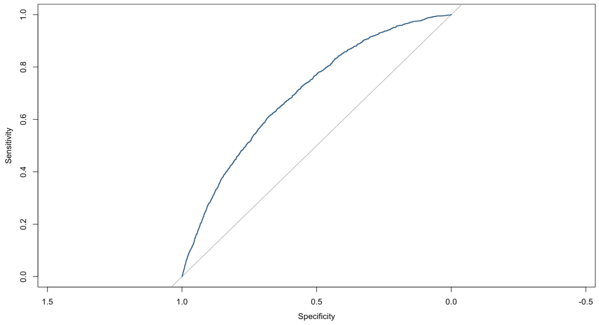


Figure 8: ROC plot of the model

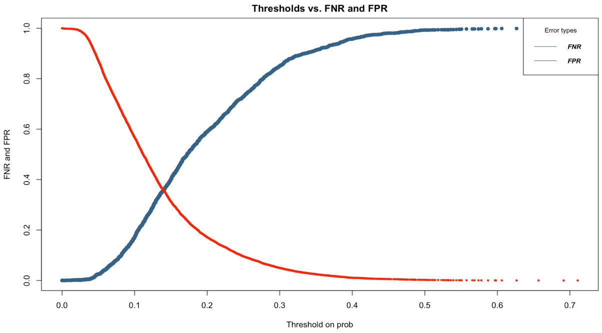


Figure 9: FNR and FPR for different probability thresholds

From here, I obtained the classification rule:

$$\begin{cases} 1 & P(default) \geq 0.14 \\ 0 & P(default) < 0.14 \end{cases}$$

This rule yielded a misclassification error of 0.141.

How early or late will the loans be fully paid off?

Exhaustive Search:

I first decided to do an exhaustive search by fitting all 2^p models. The optimal CP model had all the predictor variables in it, so I chose to include only 10 variables since that is when the CP starts to taper off.

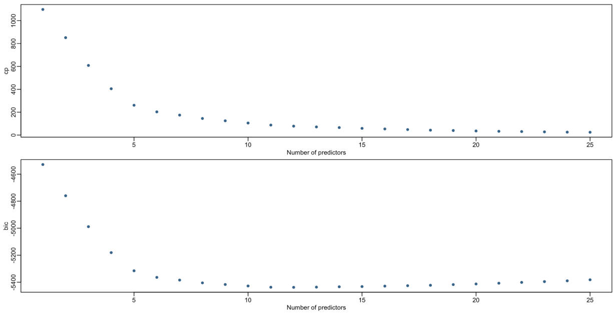


Figure 10: CP and BIC of models

This led to a model with term, interest rate, purpose, dti, delinquencies in the last 2 years, number of inquiries in the last 6 months, number of open accounts, revolving utilization and total accounts as independent variables to predict how early or late a loan would be paid back. Based on the Anova() test, all variables were significant and the test RMSE was 391.5.

LASSO Regression 1:

I chose to use the coefficients presented from lambda.1se instead of the lambda that produced the minimum mean squared error because that model included every variable. This model included the loan amount, term, interest rate, verification status, dti, number of inquiries in the last 6 months, open accounts, revolving utilization and total accounts. The test RMSE was 391.7.

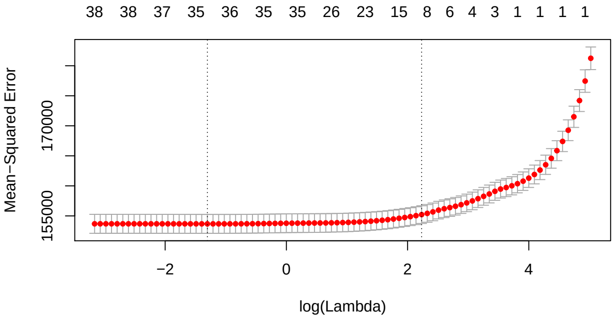


Figure 11: Plot of MSE for LASSO model

LASSO Regression 2:

Improving on the previous model, I removed loan amount as it was insignificant and verification status as it was barely significant, but this barely affected the model performance. The test RMSE increased marginally to 391.7.

LASSO Regression with Interaction Terms 1:

The performance of the previous models were not very good, so I decided to fit interaction terms to try to improve the performance and use LASSO regularization to choose a subset of variables. I chose to go with the variables that were in the lambda.1se model because the lambda.min model had too many variables and the aim was to develop a parsimonious model.

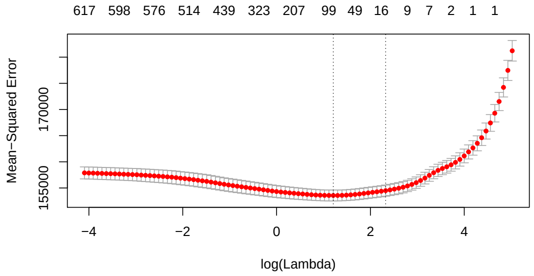


Figure 12: Plot of MSE for LASSO model

The variables in the model were term, dti, revolving utilization, loan amount * term, term * interest rate, term * verification status, term * inquiries in the last 6 months, term * total account, dti * open accounts, open accounts * revolving utilization. However, the test RMSE was 391.5, only decreasing by a bit.

LASSO Regression with Interaction Terms 2:

I removed insignificant variables from the previous model to build a model with term, dti, revolving utilization, term * interest rate, term * verification status, term * inquiries in the last 6 months and term * total accounts. This model performed the worst, with a test error of 393.1.

Final Model:

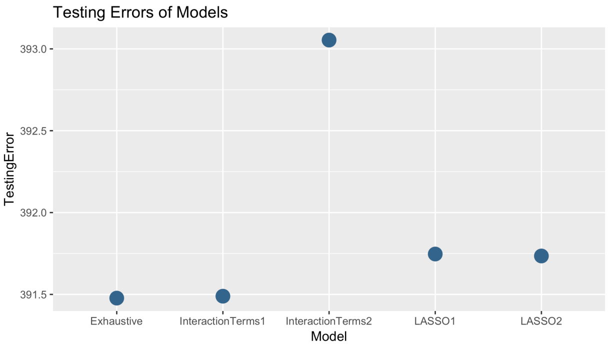


Figure 13: Test errors of models

All the test RMSEs were very high, which leads me to believe that the variables were not good predictors of how early or late a loan will be repaid. However, on a relative scale, the first LASSO regression with interaction terms performed the best, and therefore, I'll choose that as the final model. The model did not appear to violate any of the linear regression assumptions as the residuals are homoskedastistic and seem to be normally distributed. Based on the model, the variable with the highest marginal effect on how early a loan will be paid back is term * interest rate and the variable with the highest marginal effect on how late a loan will be paid back is the interest rate.

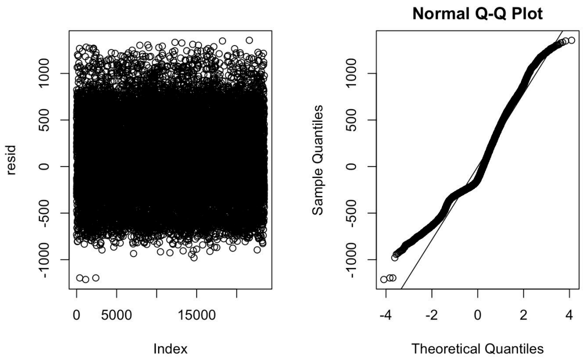
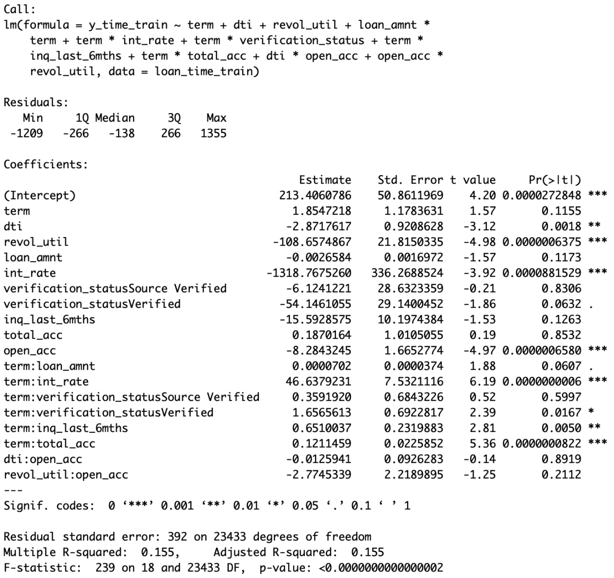


Figure 14: Residual plots



```
Call:
lm(formula = y_time_train ~ term + dti + revol_util + loan_amnt *
    term + term * int_rate + term * verification_status + term *
    inq_last_6mths + term * total_acc + dti * open_acc + open_acc *
    revol_util, data = loan_time_train)

Residuals:
   Min     1Q Median     3Q    Max
 -1209   -266   -138    266   1355

Coefficients:
                                        Estimate    Std. Error t value     Pr(>|t|)
(Intercept)                          213.4060786    50.8611969    4.20 0.0000272848 ***
term                                   1.8547218     1.1783631    1.57       0.1155
dti                                   -2.8717617     0.9208628   -3.12       0.0018 **
revol_util                          -108.6574867    21.8150335   -4.98 0.0000006375 ***
loan_amnt                             -0.0026584     0.0016972   -1.57       0.1173
int_rate                           -1318.7675260   336.2688524   -3.92 0.0000881529 ***
verification_statusSource Verified    -6.1241221    28.6323359   -0.21       0.8306
verification_statusVerified          -54.1461055    29.1400452   -1.86       0.0632 .
inq_last_6mths                       -15.5928575    10.1974384   -1.53       0.1263
total_acc                              0.1870164     1.0105055    0.19       0.8532
open_acc                              -8.2843245     1.6652774   -4.97 0.0000006580 ***
term:loan_amnt                         0.0000702     0.0000374    1.88       0.0607 .
term:int_rate                         46.6379231     7.5321116    6.19 0.0000000006 ***
term:verification_statusSource Verified 0.3591920     0.6843226    0.52       0.5997
term:verification_statusVerified       1.6565613     0.6922817    2.39       0.0167 *
term:inq_last_6mths                    0.6510037     0.2319883    2.81       0.0050 **
term:total_acc                         0.1211459     0.0225852    5.36 0.0000000822 ***
dti:open_acc                          -0.0125941     0.0926283   -0.14       0.8919
revol_util:open_acc                   -2.7745339     2.2189895   -1.25       0.2112
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 392 on 23433 degrees of freedom
Multiple R-squared:  0.155,     Adjusted R-squared:  0.155
F-statistic:  239 on 18 and 23433 DF,  p-value: <0.0000000000000002
```

Conclusion

Using an optimizing program such as Solver, investors can now use these models to maximize $E(R_p) = \sum_{i=0}^{n} R_i P($loan i is fully paid$) + R_i T_i$ as this analysis provides the necessary pieces to the puzzle.

The variable with the highest marginal effect to the probability that a loan will default is the interest rate. This is interesting because the interest rate reflects the grade of the loan, which is determined using the available credit information on the borrower. This illustrates the effects of autocorrelation between the variables as 'grade' was removed from the model due to insignificance very early on.

The classification threshold of 0.14 is relatively low, but I feel that a better unbiased-minimum variance analysis could have been done if there was a more balanced proportion of default vs. paid off loans; the higher proportion of paid off loans in the dataset may have affected the analysis.

The model to predict how early or late loans will be repaid did not perform very well as the testing error was 391 days, which is more than a year. As an investor, having my predictions off by a whole year may be detrimental to my portfolio as returns will compound over time.

Generally, machine learning tools perform better in categorizing than in point prediction. An interesting direction for analysis moving forward would be to incorporate Bayesian analysis in which we develop priors on when we think borrowers will default or pay back.