Title
Analysis of Default in Peer to Peer Lending

Author
Ramirez, Arturo

Publication Date
2016

Peer reviewed|Thesis/dissertation

University of California
Los Angeles

Analysis of Default in Peer to Peer Lending

A thesis submitted in partial satisfaction
of the requirements for the degree
Master of Science in Statistics

by

Arturo Ramirez

2016

Abstract of the Thesis

Analysis of Default in Peer to Peer Lending

by

Arturo Ramirez

Master of Science in Statistics

University of California, Los Angeles, 2016

Professor Nicolas Christou, Co-chair

Professor Yingnian Wu, Co-chair

In the United States, stagnant interest rates, bank skepticism after the financial crisis and the rise of crowd sourcing has led to the development of Peer to Peer lending as an alternative to loans offered by banks and other traditional financial institutions. Typically, investing in these loans involves manually building a portfolio one note at a time or automating the procedure based on investor defined criteria. Investor returns are directly dependent on an individual borrower's repayment of the loan, so investing in a loan that defaults results in a direct loss.

This thesis will focus on two different approaches to analyze default in this new lending environment. First, we will explore loan default as a binary classification problem. We will use initial borrower data to build a decision tree classifier and evaluate performance based on binary classification metrics . We will then examine the construction of the classifier in an effort to gain insight on the indicators of default and develop possible investment strategies. Next, we will explore loan default as a survival analysis model. This will utilize payment history data, along with the initial borrower data, to build a proportional hazards model that evaluates time until default. This model will also be explored for potential insight on investment strategy.

The thesis of Arturo Ramirez is approved.

Qing Zhou

Yingnian Wu, Committee Co-chair

Nicolas Christou, Committee Co-chair

University of California, Los Angeles

2016

To my parents

Table of Contents

List of Figures

List of Tables

Acknowledgments

I'd like to thank the UCLA Statistics department faculty and staff for their unwavering support. I'd also like to thank my family and friends.

CHAPTER 1

Introduction

Traditionally, if you wanted to get a loan, you had to submit an application at a bank or similar financial institution. Peer to peer(P2P) lending is a rapidly growing industry that provides an alternative to banks. With P2P lending, applicants submit an application to the lending service. The lending service then decides if they will offer a loan and under what terms and conditions, which includes loan length, amount, interest rate and payment schedule. Individual investors can then pick which loans to invest in and how much money they will invest. Once the full amount for a loan is raised, the applicant receives the funds. Investors then receive monthly payments of principle plus interest based on the terms of the loan.

The emergence of P2P lending in the United States can be attributed to the current social, technological, and business environment. A recent Gallup poll [6] indicates 37% positive, 33% negative (+4 net rating) in public overall opinion of banks. Public opinion has rebounded since the height of the financial crisis, as between 2009 and 2012 opinions averaged a -24 net rating. Despite this slight level off of slightly positive opinion, it still remains quite far off the +29 net rating between 2001 and 2007. Public opinion on banks has rebounded since the financial crisis, but skepticism remains. Interest rates are also at historical lows [9], which gives the public less incentive to direct their capital through banks.

Technological and social conditions also contributed to the rise of P2P lending. The rise of smart phones and ever increasing ability to stay connected through social interactions has fostered an ideal environment for P2P lending to succeed. Ebay is the closest counterpart we can point to. Ebay's popularity as an internet based communal marketplace shows the general public is already comfortable with the P2P dynamic. Though we are dealing with a

different product being offered, loans, the framework is familiar.

In this thesis, we will use two different methods to analyze borrower default in this emerging market. The first method will model loan default as a binary outcome. The C5.0 classification algorithm is used, because it is easily interpreted and flexible. Classification and Regression Tree (CART) methods in general are user friendly, and C5.0 allows for both tree and rule models. We also make use of the integrated features such winnowing, boosting, and providing a cost matrix for weighting classification errors. This allows us to build models that takes in initial data from the applicant and predict whether they will default on their loan. We will compare the standard model with a model that weights errors using the cost matrix. We will then use standard classification model evaluation metrics such as the confusion matrix, accuracy and other error rates, Cohen's kappa, and the ROC curve to evaluate our models.

The Second method will use payment histories of the loans to create a Cox proportional hazard model. The classification method described previously restricts us to the use of loans that have reached an outcome, either pay off or default. The survival analysis approach allows us to make use of current loans as well as loans that have reached an outcome. First we will test the proportional hazards assumption in the input variables. We will then build a proportional hazards model using univariate and multivariate techniques. The model will be evaluated based on the likelihood ratio, Wald, and score chi-squared statistics. The output of this model will allow us to predict a survival curve for an individual loan.

CHAPTER 2

Data Selection

2.1 Lending Club

This thesis will analyze the US peer to peer lending environment by focusing on loans originated by Lending Club, the industry leader in volume. We will restrict the analysis to personal loans. Personal loans are offered for values ranging from $1000 to $35000 and on either 36 month or 60 month terms. Lending Club uses in house minimum borrower requirements and sets the terms for all loans offered. Loan data is restricted to loans issues between 6/14/07 and 9/30/2015. All data used are publicly available at Lending Club's website [2].

2.2 Classification Data Set

The data set for classification contains borrower data that is available during the open investment period of the loan. All variables used in the classification portion are either collected directly by Lending Club during the open investment period or derived from this information. For example, Lending Club collects a borrower's FICO score range while this thesis uses the median value.

A summary of the variables considered for use in constructing the classifier can be seen in Table 3.1. These variables include a variety of income and credit data on the borrower, as well as the terms and conditions of the loan. Current loans, or loans that have not reached an outcome, are excluded.

2.3 Survival Analysis Data Set

The data set used in the survival analysis portion contains the borrower data from classification data set, plus payment history throughout the lifetime of the loan. Borrowers are required to make principle plus interest payments monthly. Early full repayment of the loan is allowed. This data set includes payment timing, payment amounts, remaining principle owed, and an indicator for loan status. The status indicator allows us to identify loans as current or in default.

The language used when discussing loan status can be ambiguous, so for this analysis, we will consider any loan that has been charged off as in default. Lending Club allows for a delinquency period. In this analysis, loans within this delinquency period are considered current. Also, if a loan becomes delinquent but goes on to be current, or subsequently, fully paid, we will not mark that loan as having defaulted.

Table 2.1: Classification Variables

Variable Name	**Description**
annual_inc	The borrower's annual income
collections	# of collections in 12 months excluding medical
cr_time	# of years since first credit line was opened
delinq_2yrs	The number of 30+ days past-due incidences.
dti	Debt to income ratio
emp_length	Employment length in years
fico_start	The borrower's median FICO
grade	LC assigned loan grade
home_ownership	The home ownership status.
initial_list_status	The initial listing status of the loan.
inq_last_6mths	The # of creditor inquiries during the past 6 months.
installment	The monthly payment owed by the borrower.
int_rate	Interest Rate on the loan
loan_amnt	The listed amount of the loan.
loan_status	Current status of the loan
mths_delinq	The # of months since the borrower's last delinquency.
mths_derog	Months since most recent 90-day or worse rating
mths_record	The # of months since the last public record.
open_acc	The # of open credit lines in the borrower's file.
policy_code	publicly available policy_cod
pub_rec	# of derogatory public records
purpose	Borrower provided loan request purpose.
revol_bal	Total credit revolving balance
revol_util	Revolving line utilization rate
term	The # of payments on the loan
total_acc	The total # of credit lines in the borrower's credit file
verification_status	The status of income verification.

CHAPTER 3

Models

3.1 C5.0 Algorithm

When analyzing default as a classification problem, the C5.0 algorithm is used to create a boosted decision tree classifier. Developed by Ross Quinlan [10], C5.0 is the next iteration of C4.5 which was popularized by the paper *Top 10 Algorithm's in Data Mining* [11]. In our case, we use C5.0 to construct a binary classifier, although it can also be used in multi-class problems. The pseudo code for constructing a C5.0 classifier is as follows:

Algorithm 1 C5.0 Pseudo Code

1: Check for base case:
2: For each attribute A, calculate the normalized information gain ratio from splitting on A
3: Select A_{best}, the attribute with the highest normalized information gain
4: Create a decision node that splits on A_{best}
5: Perform the same procedure on each subset obtained by splitting on A_{best}, and add these nodes as children of the node created by splitting on A_{best}

Bases cases are :

- All sampled observations belong to the same class → create a leaf node saying to choose that class
- No features provide information gain → create a decision node higher up using expected value of the class

- A previously unseen class is encountered $\rightarrow$ create a decision node higher up using expected value of the class.

The formulation of the decision tree here depends on the use of information gain and information gain ratio, which can be defined as follows: Let A be the set of attributes and X be the set of training examples. Let $val(x, a)$, where $x \in X$ and $a \in A$, be the value of specific observation x for attribute a. Let H be the entropy and $values(a)$ denote the set of all possible values of attribute a. Then the information gain ratio of attribute a is

$$IGR(X, a) = IG/IV$$

where the information gain, IG, is

$$IG(X, a) = H(X) - \sum_{v \in values(a)} (\frac{|x \in X | val(x, a) = v|}{|X|} \cdot H(x \in X | val(x, a) = v))$$

and the intrinsic value, IV is

$$IV(X, a) = - \sum_{v \in values(a)} (\frac{|x \in X | val(x, a) = v|}{|X|} \cdot \log_2(\frac{|x \in X | val(x, a) = v|}{|X|})$$

Attributes in A can be either nominal or numeric, which determines how the normalized information gain ratio is calculated. Nominal attributes are split by their possible values. Numeric attributes are split by some threshold L, where training set X is sorted on the values of a and L is selected as the value that maximizes information gain. Missing values are excluded from information calculations.

The initial decision tree is constructed using the described algorithm and formulas. The initial tree is then pruned using the upper limit of the binomial probability of observing E events in N trials. Pruning is carried out from the terminal nodes to the root, where for each sub tree, estimated errors of the branches are added and compared to the estimated error of replacing the sub tree with a terminal node. If creating a terminal node does not increase

the estimated error, the sub tree is pruned. This estimation method also checks to see if it is beneficial to replace a sub tree with one of its branches. This process removes extraneous attributes and then reconstructs the decision tree.

C5.0 also supports a variety of user defined functions that can help increase classification performance. These options include

- *AdaBoost* [5] can be used to produce boosted trees.
- Winnowing [8] can be used to remove irrelevant attributes.
- The minimum number of cases needed to create a terminal node can be adjusted.
- Cross validation can be implemented.
- Weighting can be applied to both individual cases and specific types of errors.
- Rule sets can be constructed in place of decision trees.

3.2 Cox Proportional Hazard Model

When we incorporate loan payment history and examine loan default from the survival analysis perspective, we construct a Cox proportional hazard model [3]. The model is used to examine the expected length of time until the occurrence of an event. In this case, the event is default.

In general, a survival model consists of two components, a baseline hazard function (denoted $\lambda_0(t)$ where t is any given time) and the effects parameters. The baseline hazard function denotes the instantaneous rate of event occurrence, and the effects parameters describe how the hazard,or failure rate, changes with variations in the explanatory covariates. When these covariates are multiplicatively related to the hazard, the proportional hazards condition is satisfied within the event process. When the proportional hazards assumption holds, the form of the baseline hazard function is not needed to estimate the effect parameters. This is the bases of the Cox proportional hazard model, which will be shown in the following:

Let Y_i represent the observed time of an observation i. Let C_i be the censoring indicator where $C_i = 1$ if the event occurred and $C_i = 0$ for a censored observation. Let X be the matrix of covariates. Then the Cox proportional hazard model has a hazard function of the form:

$$\lambda(t|X) = \lambda_0(t) \exp(X\beta')$$

Let $\theta_j = \exp(X_j\beta')$ and $X_1 \dots X_n$ be the covariate vectors for the n observations. We can then describe the partial likelihood and log partial likelihood as

$$L(\beta) = \prod_{i:C_i=1} \frac{\theta_i}{\sum_{j:Y_j \geq Y_i} \theta_j}, \ell(\beta) = \sum_{i:C_i=1} (X_i\beta' - \log \sum_{j:Y_j \geq Y_i} \theta_j)$$

Maximizing the log partial likelihood over β produces the model maximum likelihood parameter estimates. The accompanying partial score function and Hessian matrix of the partial log likelihood is

$$\ell'(\beta) = \sum_{i:C_i=1} (X_i - \frac{\sum_{j:Y_j \geq Y_i} \theta_j X_j}{\sum_{j:Y_j \geq Y_i} \theta_j})$$

$$\ell''(\beta) = -\sum_{i:C_i=1} (\frac{\sum_{j:Y_j \geq Y_i} \theta_j X_j X_j'}{\sum_{j:Y_j \geq Y_i} \theta_j}) - \frac{\sum_{j:Y_j \geq Y_i} \theta_j X_j \times \sum_{j:Y_j \geq Y_i} \theta_j X_j'}{(\sum_{j:Y_j \geq Y_i} \theta_j)^2})$$

These two components can then be used to maximize the partial likelihood. We can then evaluate the inverse of the Hessian matrix at the estimate of β to produce an approximate variance covariance matrix for the estimate, as is standard when producing approximate standard errors for the regression coefficients.

The case of tied times in Y must be addressed in order for this procedure to be applied. In our case, we will use Efron's method [4].

CHAPTER 4

Analysis of Default

4.1 Default as Binary Classification

We will now build a binary classifier using the variables listed in table 2.1 as potential model covariates. The first step in the model building procedure is to perform an initial data exploration. The first step is to examine the sample size and proportion of events. When constructing classifiers, we must always ensure the sample size is adequate and that no single class has a minuscule occurrence rate, or else we may need to explore corrective methods for modeling rare events. As figure 4.1 shows, we do not see any issues here. In the 36 month loan group we have a default rate of 25.35% and in the 60 month loan group we have a default rate of 60%, giving an overall rate of 31.85%.

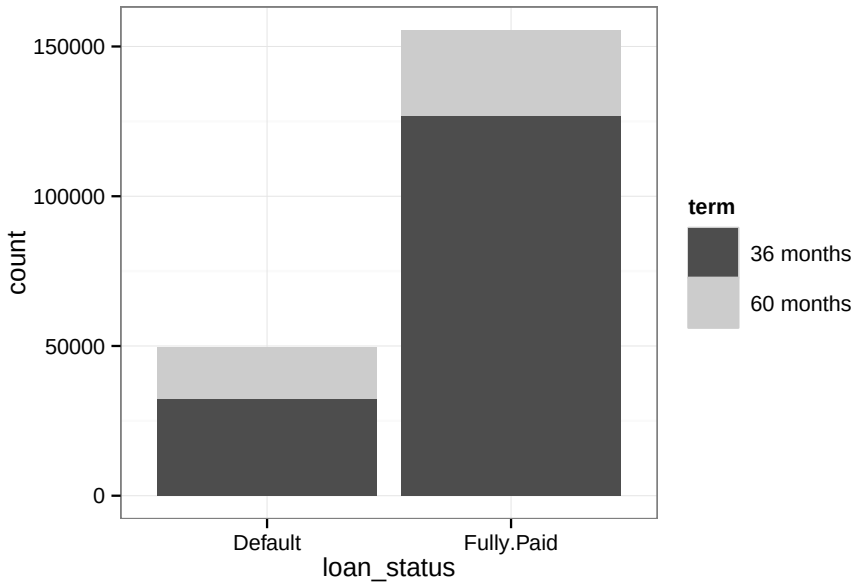

Figure 4.1: Default Count by Term Length

We should then explore missing values in our data set. The C5.0 algorithm is capable of handling missing values in that they are excluded from the information gain ratios used to make the tree splits. But if we encounter substantial or systematic missingness, we could explore imputation methods to replace them. In this case, we have 8,249 total missing values across 205,023 total observations, with 96.1% complete cases. Missing data does not appear to be an issue, so we will allow C5.0 to handle missing values in the standard method.

Now we should examine the loan standards that Lending Club imposes upon their loans. As stated previously, Lending Club determines the terms and conditions of the loans that they facilitate, individual investors have no influence other than deciding whether it will get funded or not. Without any prior knowledge, it would seem that the loan grade would be the most influential covariate we have at our disposal. Lending Club uses a proprietary model to assign loan grades. Grades are ordinal, with A being the "best"credit profile. We can see the breakdown of loan grades in more detail in figure 4.2.

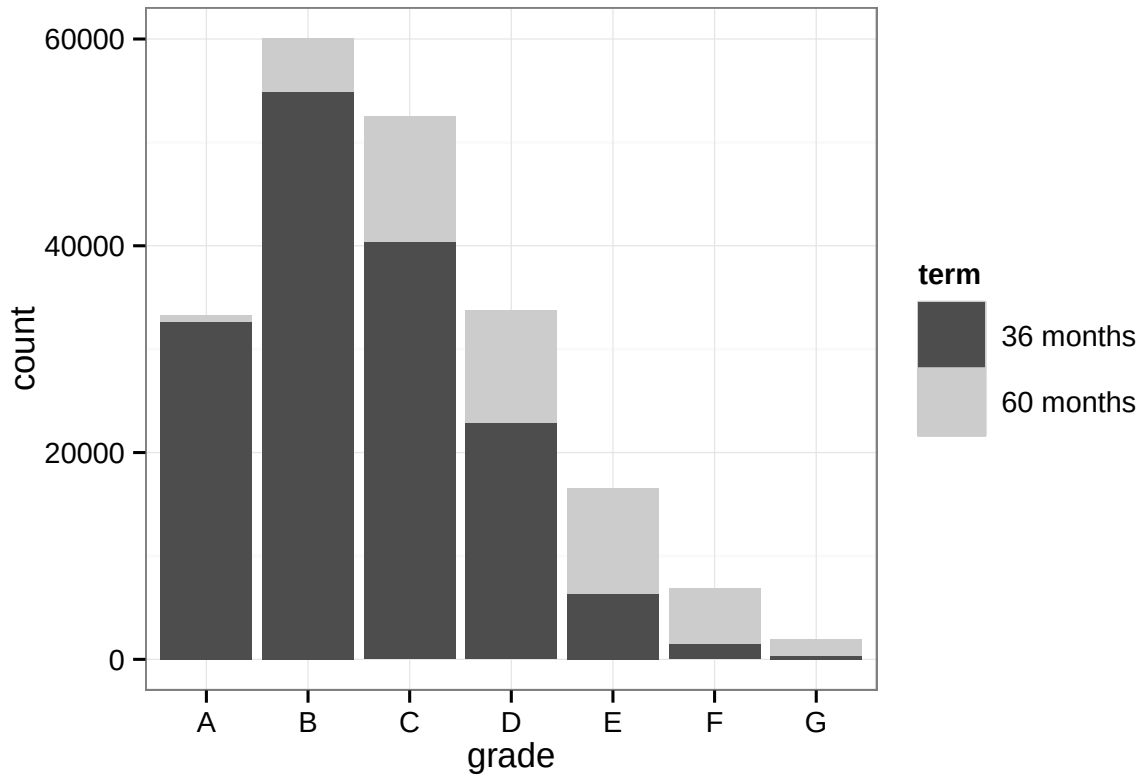

Figure 4.2: Loan Grade by Term Length

Figure 4.3 examines the dollar values of loans offered. By Lending Club standards, we see only personal loans with dollar values between $1,000 and $35,000. Lending Club also offers business loans which have longer term options and higher dollar value options, but we are restricting this analysis to personal loans. We can see that loan amounts are centered around $10,000. We have a median value of $12,000, mean value of $13,390, with a first quartile of $7,200 and third quartile of $18,000.

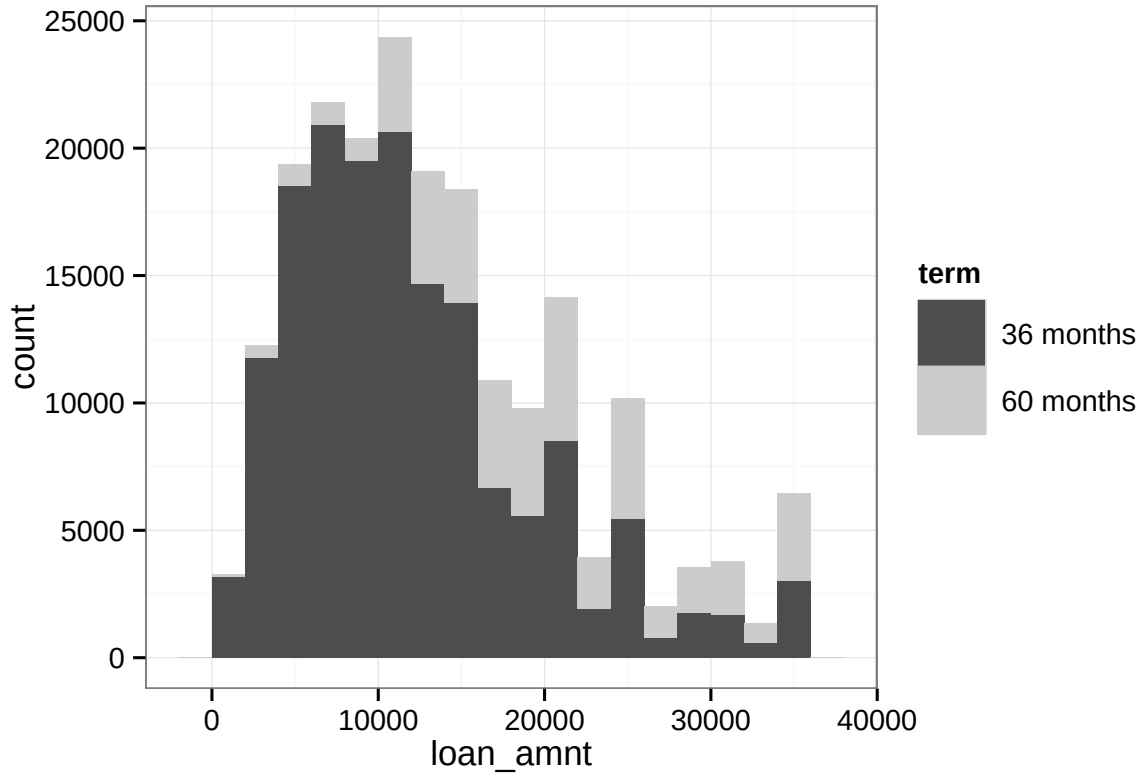

Figure 4.3: Loan Dollar Values by Term Length

Lending Club sets interest rates by taking their minimum baseline rate and adding a "risk and volatility" component. The baseline rate is stated to be 5.05%, with the additional component added based on the assigned loan grade. This adjustment included valuation for loan sub grades, which we are not including in our grade covariate. After taking in to account the loan grade adjustment, the lowest possible interest rate is 5.32%, with a maximum interest rate of 28.99%. Figure 4.4 shows that the interest rates correspond to loan grades being centered around B and C and also shows the loan sub grade differentiation.

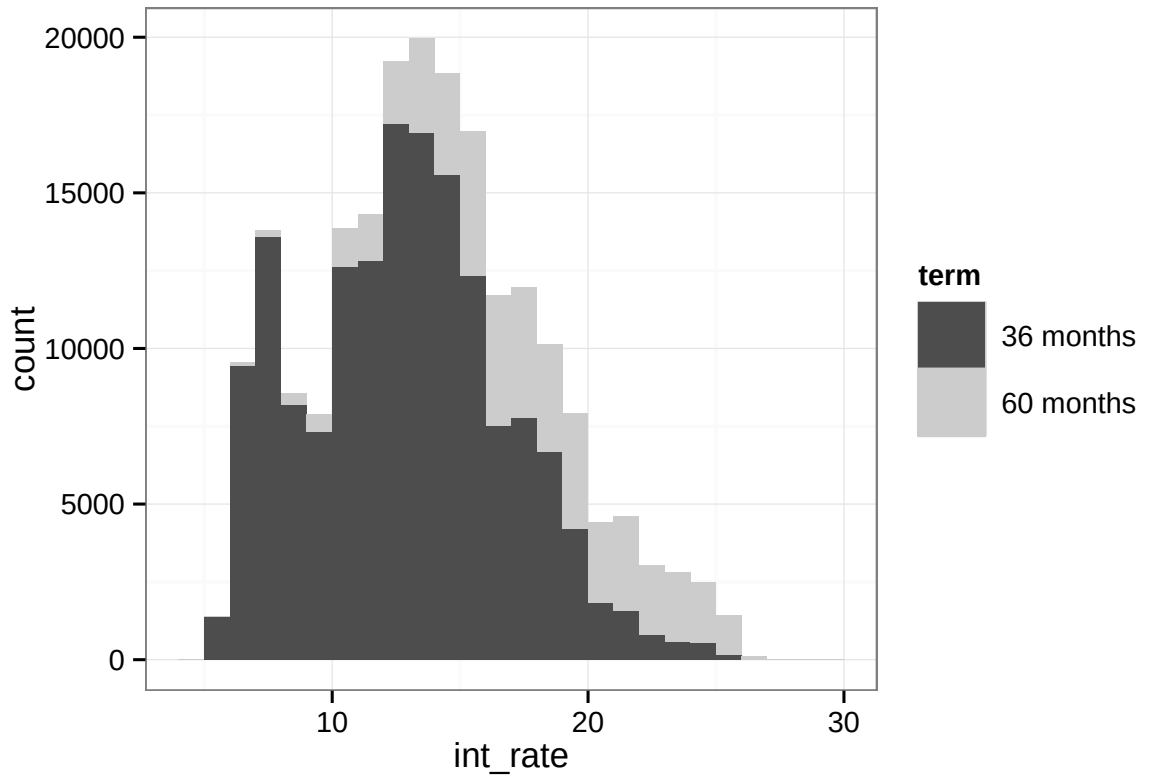

Figure 4.4: Interest Rates by Term Length

The last remaining Lending Club identified criteria is minimum starting FICO score. This threshold has been changed by lending club over the years as their policies have evolved. This data set includes loan from the introductory stages of the website. Because this includes legacy loans, the FICO score floor may not be consistent, but we should still see a clear floor value.

We've examined all of the self described terms of loan offerings from Lending Club, and they all appear to check out. We're now ready to begin building a C5.0 model. We now split the data between a training and testing set. The training set is used to develop the model, and the testing set it used to evaluate the model fit. The split the data on a 80% training, 20% testing split, using a form of random sampling that preserves the distribution of class proportions (Default vs Paid.Off) in both samples. After training the models, all performance metrics are relative to the training set.

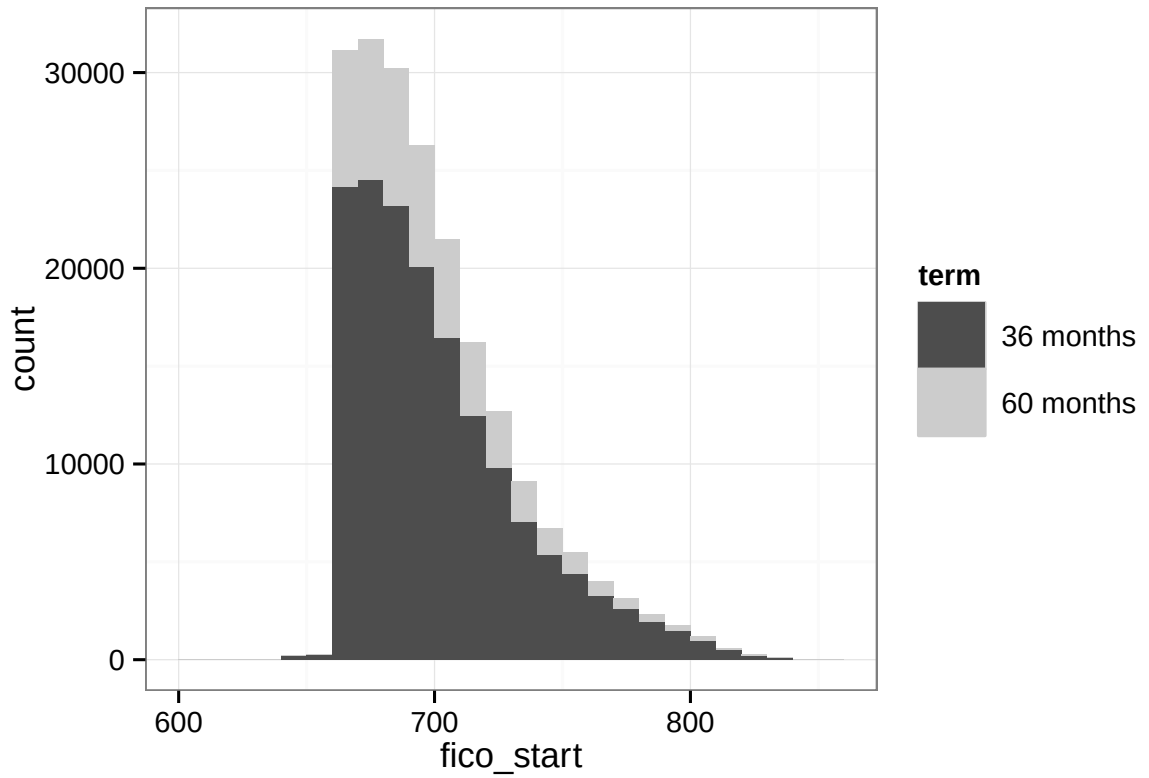

Figure 4.5: Starting FICO by Term Length

First we attempt to build a single decision tree, as a preliminary exploration of the process. This yields a classifier that performs no better than than random assignment in the test set, as evident in the 0.43 AUC . A single decision tree also has a large false negative, or full repayment is predicted when default is the true outcome, of 19.7%. We attempt to remedy this issue by constructing a boosted decision tree, which as described in 3.1, utilizes *AdaBoost* to create a boosted tree classifier. By implementing boosting, we are able to reduce false negative rate down to 9.4%. We also see a significant boost in AUC score, up to 0.65, which can be seen in figure 4.6.

We are singling out the false negative rate in this early stage of model comparison because it is logically related to the direct loss of investment. The reason being predicting a loan will be fully repaid and subsequently investing in that loan based on this prediction, only to have it fail, results in the loss of both initial investment and any forecasted profit off interest.

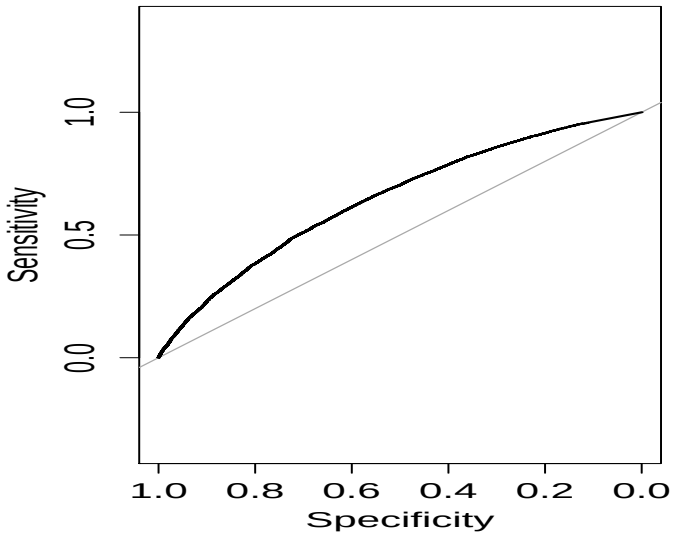

Figure 4.6: Boosted Tree AUC

We must remember that funding a loan is a non secured investment. This means if we are weighing model predictions false negatives are far more detrimental to investment strategy than false positives. A false positive will simply result in a missed opportunity to make a profit.

Because of this preference of one type of error over the other, we can explore the implementation of a weight matrix. Feeding our C5.0 classifier a weight matrix (error weighting, not individual case weighting), along with tuning other parameters helps us arrive at our final model. This final model also utilizes boosting, with the addition of penalizing false negatives five times more than false positives, and changes the minimum cases needed to create a terminal node from 2 to 6. A limitation of implementing a cost matrix is that we can no longer predict class probabilities. This is because the class probabilities derived from the class distribution in the terminal nodes may not be consistent with the final predicted class. Therefore, we can not use AUC for model comparison between the boosted and weighted classifiers (single tree is omitted because it is outperformed by the boosted tree in nearly all evaluation metrics). We will instead use the confusion matrix (outlined in table 4.1), and other metrics based on classification error rates, such as sensitivity, specificity, and others. The full list can be seen in table 4.2.

Table 4.1: Confusion Matrix Comparison

		Observed	Boost Tree		Final Model	
		Default	Fully.Paid	Default	Fully.Paid	
Predicted	Default	2.5%	3.3%	4.3%	5.5%	
	Fully.Paid	19.8%	74.4%	18.0%	72.2%	

There is no discernible difference in accuracy between the two models, as the 95% confidence intervals overlap. The trade off between sensitivity and specificity is expected, as this is a byproduct of introducing weighted classification errors. This is the sacrifice made in order to reduce false negatives. We also see that the final model shows an increase in Cohen's Kappa, which is generally thought of as a more robust measure of overall percent agreement(converse of error rate). This is because Kappa accounts for agreement between observed and predicted by chance. Therefore, if our only goal is to predict a loan that will be fully repaid, we should use the class prediction from the final, weighted model. But, from a practical standpoint, the boosted tree model results may be preferred if we choose to evaluate class probabilities. If we take probability predictions to represent the level of confidence of observing a loan that will be fully repaid, the boosted tree should guide investment decisions, rather than the weighted model.

Table 4.2: Error Based Model Comparison

	Boosted Tree	Final Model
Accuracy Lower	0.766	0.761
Accuracy Upper	0.773	0.768
Kappa	0.096	0.150
Sensitivity	0.958	0.929
Specificity	0.113	0.191
Detection Rate	0.744	0.722
Detection Prevalence	0.942	0.902
Balanced Accuracy	0.535	0.560

Up to this point, our model building process has focused exclusively on performance. We've used AUC, sensitivity, specificity, and other standard model fit evaluation metrics to guide us. But, perhaps we can also develop investment strategy based on the relationship between the covariates and the outcome, rather than focusing exclusively on model predicted outcome. To do this, we can examine the role our covariates play in predicting outcome. This is one of the advantages of using the C5.0 algorithm; we can easily interpret the influence of the covariates. The single tree, boosted tree, and final model all make significant use of initial winnowing of covariates. The pruning that follows also helps single out the most influential covariates. Table 4.3 provides an overview of how we whittle down the number of attributes needed to make a prediction. In all three cases, more than half the covariates are winnowed before any information gain calculations are made, and in each case not all remaining covariates are used in the decision tree(s). We also notice that the final model suffers a significant reduction in overall error. This is due to the weighting matrix allowing for far more false positives. We should also note the relatively small number of trees created when boosting is implemented. The C5.0 algorithm automatically halts boosting if no increase in accuracy is detected.

Table 4.3: Tree Construction Overview

	Single Tree	Boosted Tree	Final Model
# Trees	1	17	12
Attributes Winnowed	15	15	21
Attributes Used	9	11	5
Error %	22.2%	22.8%	34.0%

From here, we shift our focus to the attributes that survive winnowing and pruning and are used in the models. The C5.0 algorithm measures predictor importance by determining the percentage of training set samples that fall into all the terminal nodes after the split on that predictor. This is how the attribute usage percentage is calculated. We will use this attribute usage percentage as our predictor importance metric. Table 4.4 shows the breakdown of variable usage between the single tree, boosted tree, and final model.

Table 4.4: C5.0 Model Attribute Usage

	Single Tree	Boosted Tree	Final Model
Attribute	dti (100)	loan_amnt (100)	term (100)
(Usage %)	loan_amnt (11.37)	purpose (100)	int_rate (100)
	delinq_2yrs (8.26)	dti (100)	inq_last_6 (91.42)
	home_own (8.15)	ann_inc (100)	mths_last_rec (77.47)
	purpose (3.57)	home_own (99.98)	mths_last_derog (44.61)
	pub_rec (1.3)	delinq_2yrs (99.97)	
	mths_last_derog (1.22)	coll_12_mths (99.92)	
	revol_bal (0.34)	pub_rec (97.35)	
	ann_inc (0.20)	revol_bal (91.82)	
		mths_last_derog (80.56)	
		mths_last_delinq (12.96)	

We notice that with the boosted tree model, covariate influence follows what we would have expected without any type of exploration, with loan amount, annual income, loan purpose, and debt to income ratio dominating the usage percentages. The rest of the variables are related to time since last negative credit event. This suggests that recent credit history plays a pivotal role in default probability. Again, this is not unexpected. From this model we see that current earnings and most recent credit history are the most important covariates. But, the construction of the weighted final model starkly contrasts the boosted model. Here we see trees consisting of only 5 covariates. These include a couple of the recent credit history covariates seen in the boosted model, but the weighted model is dominated by loan term length and interest rate. This could be due to the fact that Lending Club sets the terms of the loans offered. Interest rate is a function of base rate plus volatility adjustment, where the adjustment is based on loan grade. This loan grade is proprietary, and is likely a function of many of the other covariates measured in the data set. So it would seem that interest rate serves as the best summary of a wide swathe of covariates.

This also is not unexpected, as higher interest rates will be tied to riskier loans and thus

a higher probability of default. Currently, this dimension reduction only serves as a way to boost prediction accuracy. But, because of this reduction, we can imagine that model training time can be significantly reduced. If a secondary market ever develops that trades at the same frequency of tradition stocks, this training time reduction could prove to be valuable.

4.2 Default as Survival Analysis

Using the data set that includes payment histories for each loan, we can approach loan default from the view of a survival analysis problem rather than a classification problem. In this sense, we are attempting to predict time until default.

This approach begins with examining the validity of the proportional hazards assumption, which states that the covariates are multiplicatively related to the hazard. We can do this by using the Kaplan Meier estimator to produce estimated default survival curves. In the case of categorical covariates, we can also perform a log rank test to determine if there is a difference between survival curves by category. Not only will this allow us to examine the proportional hazards assumption and identify candidate covariates for the proportional hazards model, but we can also gain an insight as to how these covariates are univariately related to survival.

We can start out by estimating the survival curve for 36 month and 60 month term loans, as shown in figure 4.7. From here we can see that survival rates for 60 month loans are lower than that of 36 month loans. There does not appear to be any issues with the proportional hazards assumption. We should note that we see a precipitous drop off in survival for the 36 month loans, but this takes place after the 36 month mark. This is due to collections efforts on late or past due loans that extend beyond the 36 month mark, while the loan is still classified under the 36 month category. Also, the log rank test indicates there is a significant difference ($\alpha = .05$) between the two survival curves.

Next, we examine the survival curve by loan grade in the same fashion, seen in figure 4.8. Here we see hazard by grade appears to be proportional as well. We do see slight

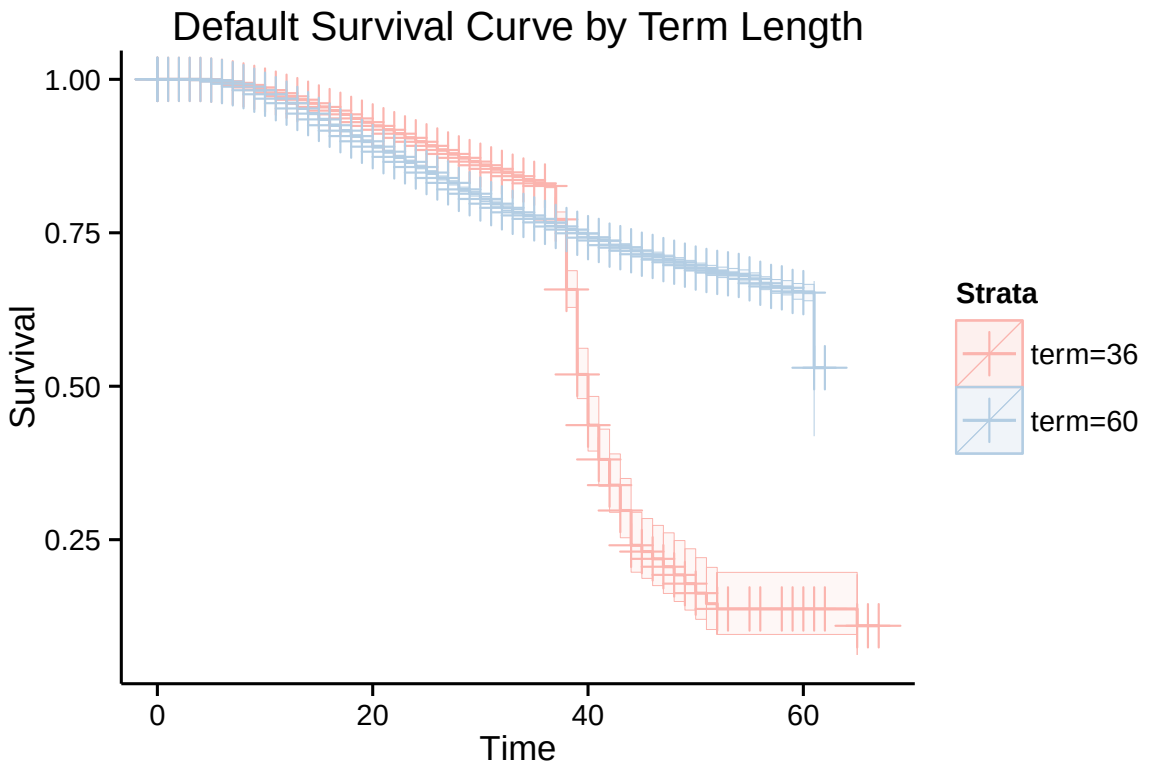

Figure 4.7: Default Survival Curve by Term Length

overlap near the end of loan lifetimes in the F and G grades, suggesting there may not be a difference in hazard rates among these two groups. But, because this cross over is only slight and occurs near the end of the loan lifetime, we will ignore it. As expected, the risk of default significantly increases both over time and as risk grade drops.

We then perform this univariate analysis on the remaining covariates and determine that we are comfortable with the proportional hazards assumption. In our case, we can safely move on to the model fitting procedure, but in the event of a violation, there are possible remedies. First, we could stratify by categories of that variable and allow for a different baseline hazard function for each stratum, while still using all data in parameter estimation. Second, we could cut the time intervals and fit a different Cox model for each time interval. Lastly, if we would still like to use a single model, the extended Cox model could be used. In this case, we would include the covariate in violation as a time dependent covariate.

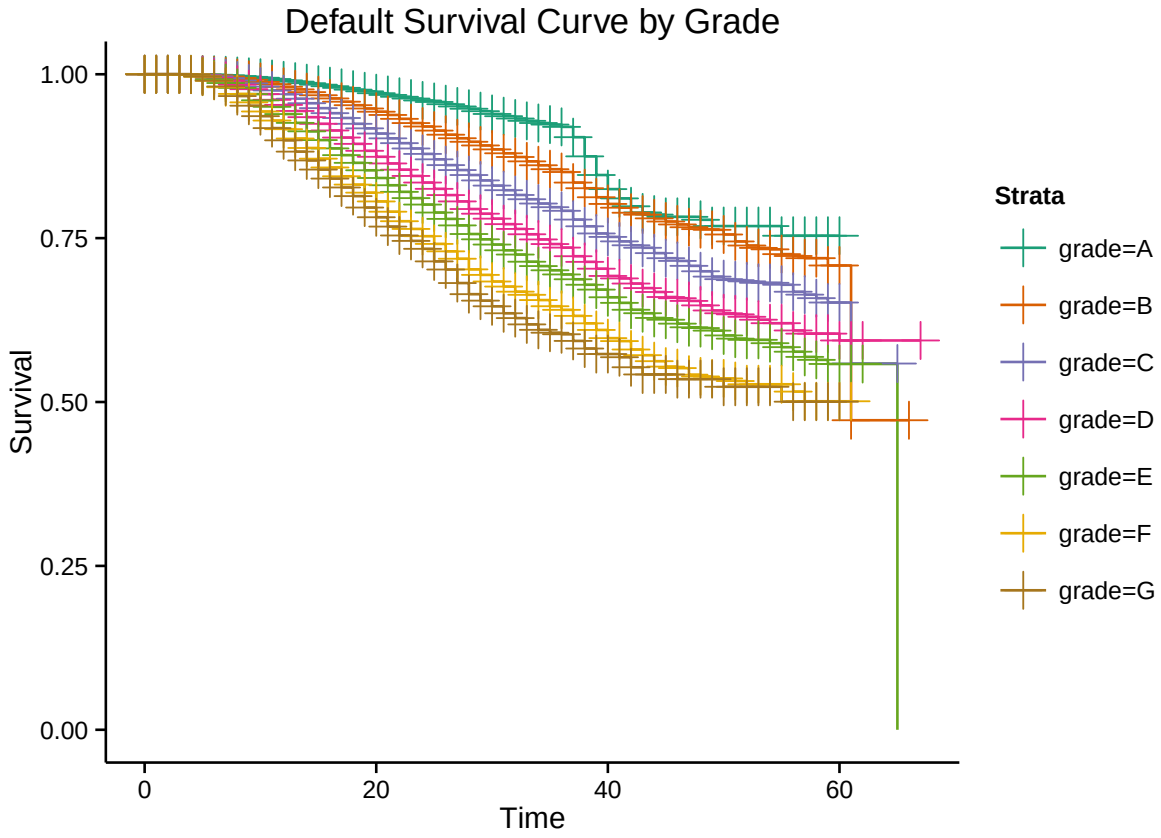

Figure 4.8: Default Survival Curve by Grade

Here, we do not encounter any violations, so we'll continue with the model building procedure normally. The issues with step wise regression techniques are well documented [7], so we will explore other options in variable selction. Because of the relatively large number of candidate covariates, we can use lasso regression [1] to help in the variable selection process. This shrinkage method reduces extraneous parameters' estimates to zero, and functions similar to the winnowing and pruning process used in the C5.0 algorithm. A summary of the model fit produced by the lasso regression can be seen in table 4.5. We should note that debt to income ratio is represented as a decimal value, and interest rate as the integer value of percentage. Scaling will aid in interpretability, but we don't see any relationships that are out of the ordinary.

In the case of the Cox proportional hazard model, the exponential value of the coefficient

Table 4.5: Lasso Cox Model Summary

Covariates	coef	exp(coef)	se(coef)	z	Pr(>—z—)
term60	-0.180	0.835	0.013	<0.001	0.000
gradeB	0.305	1.357	0.029	10.635	0.000
gradeC	0.450	1.569	0.036	12.456	0.000
gradeD	0.532	1.703	0.045	11.802	0.000
gradeE	0.552	1.737	0.055	10.065	0.000
gradeF	0.582	1.789	0.065	8.920	0.000
gradeG	0.568	1.765	0.076	7.435	0.000
dti	0.011	1.012	0.001	16.330	0.000
InterestRate	8.575	5297	0.374	22.919	0.000
EmploymentLength1-4 years	-0.092	0.912	0.018	<0.001	0.000
EmploymentLength10+ years	-0.228	0.796	0.019	<0.001	0.000
EmploymentLength5-9 years	-0.099	0.906	0.019	<0.001	0.000
Inquiries6M	0.096	1.101	0.004	26.687	0.000

can be used to examine the odds ratios of the covariates. Here, grade A and employment length of less than 1 year are the reference categories. Examining the odds ratios is analog to examining the role that each covariate plays in the decision tree method. It allows us to see how each covariate influence odds of experiencing loan default. Becuase of the lack of interaction effects, interpretation in straight forward. Typically, interaction effects can be explored if there is underlying theory behind the inclusion, but here we have a large data set and utlize lasso regression, which depends on cross validation. In less computationally intensive environments, this may be worth persuing.

After fitting the model, we can examine model fit by evaluating the Schoenfield residuals. This metric is useful for assessing time trend or lack or proportionality in the selected covariates. They are constructed based on plotting the residual versus event time. In large samples and when the assumptions hold, they should sum to zero, have expected value of zero, and be uncorrelated. In this visual check, we are looking for systematic deviation from

the mean of zero. A sample of this plot figure 4.9.

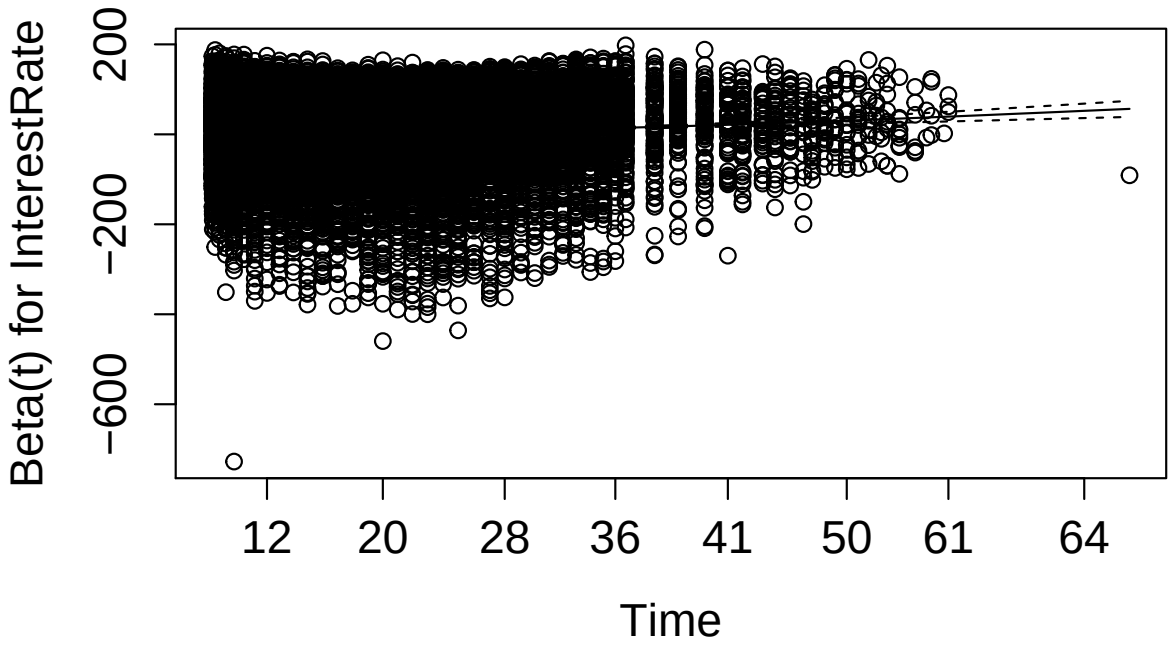

Figure 4.9: Schoenfield Residuals for DTI

Now that we have our final survival model, we can use it to predict survival curves. This can be done on both existing loans and new loans that are offered, as long as we have the values of the covariates in the model. It's important to note that when estimating survival curves for potential investments, one must consider the term length also. Evaluating a 36 month loan on the predicted 60 month survival rate could be detrimental to investment strategy and result in a loss.

CHAPTER 5

Concluding Remarks

5.1 Conclusion

This thesis has outlined two potential methods of evaluating default rates for loans in the Peer to Peer Lending market. The first method being classification based on initial loan application data. This process begins with initial data cleaning, missing value checks, and exploratory statistics. Once prepped, the data set is split between training and testing sets. The user then builds, evaluates, and compares models until a suitable classifier is constructed. The user then deconstructs the classifier and attempts to gain insight as to the relationship between covariates and the outcome.

The C5.0 algorithm is ideal for this situation because it is user friendly and flexible. A user with no previous knowledge of statistics or machine learning can construct a classifier with results that can be easily interpreted, in the form of either a decision tree or rule set. But, it also offers more complex customization options for a user with more experience constructing classifiers. The practical use of classifying loan default in this environment is that the end user can use the model to make decisions on which loans to invest in. As such, we can presume that individuals from a variety of backgrounds can take advantage of this approach.

The second potential method outlined is approaching default as a survival analysis problem and predicting time until default. For this portion, we use the Cox proportional hazard model, which can incorporate censored data. In our case, a censored observation is a current loan, which the classification method can not use.

This procedure involves initial data cleaning and setup, followed by examination of the

proportional hazards assumption in potential covariates. This is performed by examining the Kaplan-Meier estimator, which provides a survival curve estimate. Potential covariates are then transformed as needed, and a proportional hazards model is built using univariate and multivariate regression techniques. Model fit is evaluated using hypothesis testing on both the covariates and model fits, using the Wald and likelihood ratio tests, respectively. After establishing a suitable model, estimated survival curves for existing loans can be used to guide investment decisions. The model can also be used to estimate survival curves for newly funded loans by running the parameter values of the new loan through the constructed survival function.

The methods highlighted by this thesis provide potential investors with loan default evaluation tools that can be used under different scenarios, depending on the available data. The approaches are shown through the lens of Peer to Peer lending, and although issues specific to this environment are discussed, the highlighted procedures can be generalized to fit any class of similar problems.

5.2 Future Studies

This thesis focuses on methods for measuring default. This can serve as the foundation of an extension into expected return on investment. It may be beneficial to incorporate expected profits when constructing a model, and by extension, an investment strategy. This would also include the evaluation of collections on loans before they are charged off. Accounting for expected recovery amount could give a more accurate valuation of loans that do default. In the same line, early repayment reduces profit since the remaining balance accrues less interest, or none at all in the case of a fully repaid loan. The entire scope of expected profit can be incorporated in to this kind of analysis, which would fall in line with more traditional portfolio valuation techniques.

If one were to focus solely on classification error rates, we could explore the option of ensemble learning. Here we used the C5.0 algorithm for flexibility and interpretability, but there are a host of other classification techniques that can be utilized. When combined in

an ensemble of learners, it's possible we can obtain far more accurate predictions.

A limitation of this analysis is that it depends on whatever data that Lending Club has made publicly available. The survival analysis approach could benefit from the inclusion of a time varying component. It seems natural that changes in credit score and other potential covariates would have an effect on survival rates. Additional survival analysis methods could also be explored, as they could outperform the cox proportional hazard model.

References

[1] The lasso method for variable selection in the cox model. *Statist. Med.*, 16(4):385–395, 1997.

[2] Lending Club. Dowload data. `https://www.lendingclub.com/info/download-data.action`. Accessed: 2015-09-01.

[3] D. R. Cox. Regression models and life-tables. *Journal of the Royal Statistical Society. Series B (Methodological)*, 34(2):187–220, 1972.

[4] Bradley Efron. The efficiency of cox's likelihood function for censored data. *Journal of the American Statistical Association*, 72(359):557–565, 1977.

[5] Yoav Freund and Robert E Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119 – 139, 1997.

[6] Jeffery M. Jones. Ratings of us banking industry level off. `http://www.gallup.com/poll/184916/ratings-banking-industry-level-off.aspx`. Accessed: 2015-09-01.

[7] Michael S. Lewis-Beck. Stepwise regression: A caution. *Political Methodology*, 5(2):213–240, 1978.

[8] Nick Littlestone. Learning quickly when irrelevant attributes abound: A new linear-threshold algorithm. *Machine Learning*, 2(4):285–318, 1988.

[9] Board of Governors of the Federal Reserve System (US). Effective federal funds rate. `https://research.stlouisfed.org/fred2/series/FEDFUNDS`. Accessed: 2015-09-01.

[10] Ross Quinlan. See5 and c5.0. `https://www.rulequest.com/see5-info.html`. Accessed: 2015-09-01.

[11] Xindong Wu, Vipin Kumar, J. Ross Quinlan, Joydeep Ghosh, Qiang Yang, Hiroshi Motoda, GeoffreyJ. McLachlan, Angus Ng, Bing Liu, PhilipS. Yu, Zhi-Hua Zhou, Michael Steinbach, DavidJ. Hand, and Dan Steinberg. Top 10 algorithms in data mining. *Knowledge and Information Systems*, 14(1):1–37, 2008.