An Analysis of LendingClub Data to Forecast Demand and Predict Risk of Loans

Thomas Munduchira **Sulli Vo** **Dannie Vo** **Ricardo Rendon Reynoso**

Sarah Rahman

1 Introduction

LendingClub is a peer to peer lending company in which their product allows consumers to both invest and borrow loans. They offer multiple kinds of loans like student loans, personal loans, auto refinancing loans and even business loans. The borrowers who are interested in obtaining loans will get a loan grade assigned to them which affect their interest rates and the amount of money they can borrow. A lot of the LendingClub data leads to insightful conclusions about the borrowing and investing patterns of all kinds of individuals. Through our investigation, we will explain patterns and similarities of the behaviors of borrowers and investors.

1.1 Questions of Interest

We intend to start off with exploratory data analysis of all the factors involved to find patterns and relationships. We will look at the data from multiple angles to get a sense of the intricacies that lie within the data. We will additionally match the trend we see in the data to external events to try to explain why such is happening.

We will also conduct time series decomposition in regards to the average loan amounts being requested. We will take a look at the trend and the seasonality so that we can better forecast spikes in demand.

After, we will try to fit prediction models in order to answer a couple questions: namely whether a loan request from a client should be funded or not from the perspective of the bank, and what interest rate a borrower would get for a loan from the perspective of a client. After finding good models, we will deconstruct them in order to get a deeper sense of the important aspects in such decisions.

1.2 Dataset

The dataset we are using is a compilation of data on loans issued by LendingClub from the period 2007 to 2015. The data includes information on the current loan status (how much has been funded so far, how much has been paid off, etc) as well as information about the borrower (occupation, income, credit score, etc). This data lends itself to a variety of interesting financial analysis, notably time series analysis since the data is date stamped.

We will touch on a number of variables present in the dataset throughout the course of this analysis. We will consolidate the meanings of all these variables here for future reference.

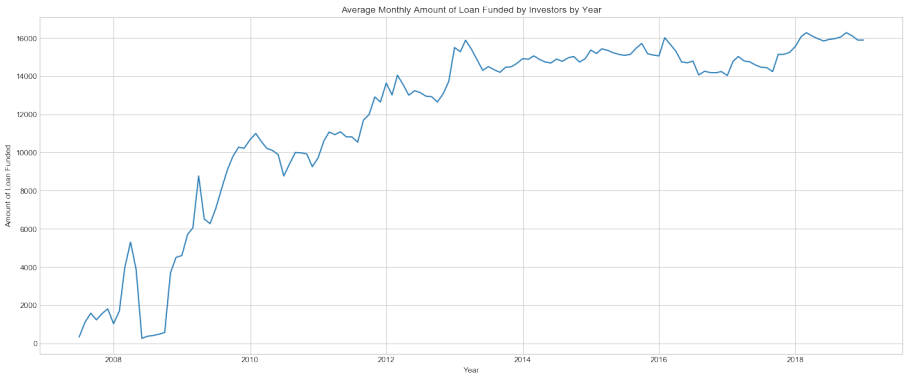
- loan_amnt: listed amount of the loan applied for by the borrower
- funded_amnt: total amount committed to that loan at that point in time
- funded_amnt_inv: total amount committed by investors for that loan at that point in time
- term: number of payments on the loan. Values are in months and can be either 36 or 60
- int_rate: interest rate on the loan

- installment: monthly payment owed by the borrower
- grade: loan grade that corresponds to the risk of the loan
- loan_status: current status of the loan
- total_bal_il: total current balance of all installment accounts
- emp_title: job title of the borrower
- next_pymnt_d: next scheduled payment date
- sec_app_mort_acc: number of mortgage accounts at time of application for the secondary applicant

More information about the dataset can be found here:

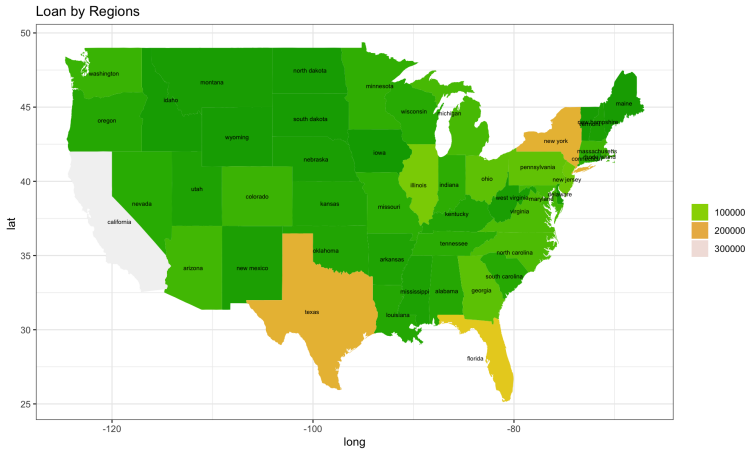
`https://www.kaggle.com/wendykan/lending-club-loan-data`

2 Data Exploration

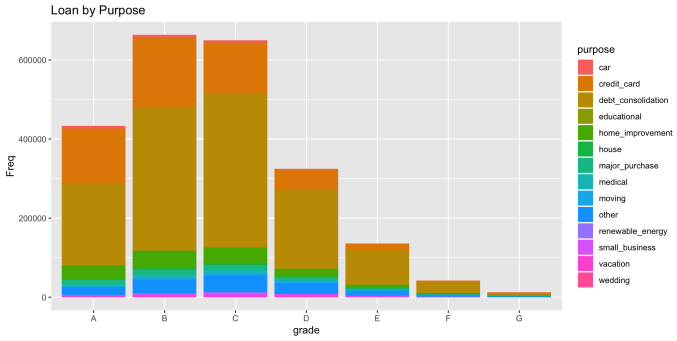


Here is the line graph of the amount of loans funded over the years. The year 2009 shows a stark dip in the loans funded. After that, the number of loans funded almost exponentially increases from 2009 to 2013, and then plateaus after that from 2013 to 2018.

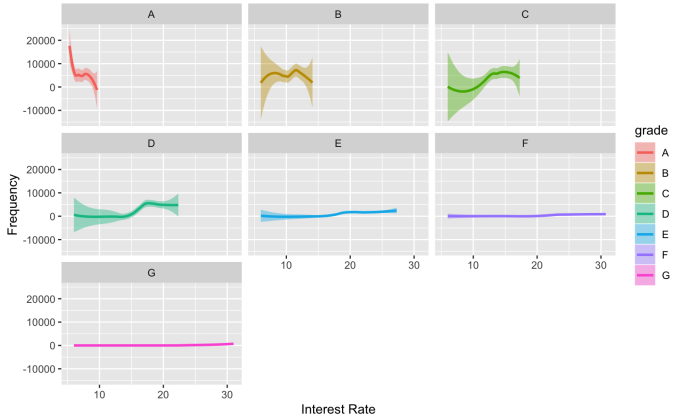
Based on the year-wise plot above, we can see that the average amount of loans kept rising from 2007 to 2015, but then fell from 2015 to 2016. This is most likely because the Federal Reserve raised its loan interest rate up by 0.25% in 2015-2016, causing people borrowing loans as well as the amount of loans requested to decrease. Moreover, the loan amounts decreased starting in March that year based on the first plot because the first interest rate increase happened in March.



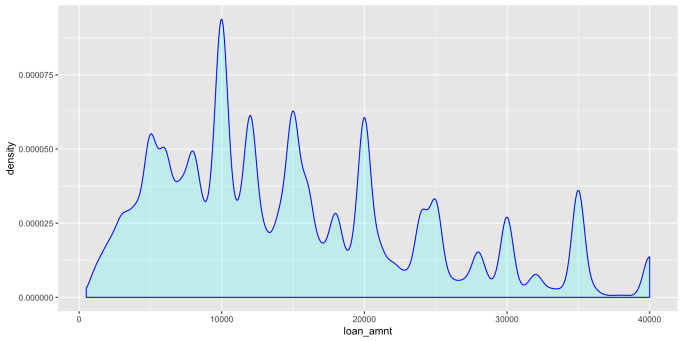
California has the most people asking for a loan. Texas and New York are the next biggest in terms of the number of people requesting loans. The 2019 consensus reveals that California is the most populated state, followed by Texas, Florida and New York. Since these regions are some of the most populous in the United States, it is understandable that they have the highest number of loan requests.



Debt consolidation is the most popular reason for a loan, followed by paying off credit cards. After that, it is home improvement/houses and the miscellaneous category. On the other hand, weddings, vacations and even cars are the least popular reasons for people to get loans. Loans with grades A, B, and C have a higher percentage of loans that are credit card related. For the loans that are grade D and below, credit card loans are not even 1/5th of the loans, but for grade A loans, credit card loans are almost a third of all loans. Loan grades are determined using credit and income data. It is interesting to see that better grade loans (hence better credit scores and income) are going towards paying off credit cards.



Here we see the frequencies of loans of different interest rates, separated into bins of different grades. The grade A loans have the highest frequency of low interest rate loans. For grades D or better, the frequency of loans with different interest rates vary a lot. But loans of E, F and G have almost a constant frequency of loans for different interest rates. In other words, there is almost an equal number of loans of all possible interest rate levels. This is expected since we expect better loan grades to be handed lower interest rates, while worse loan grades are expected to get interest rates that are across the spectrum.



This is the density distribution of the loan amounts. Based on the plot, loan requests range from $0 to $40,000. The shape of the graph is skewed to the right, signaling that most of the loans requested are relatively small and only a few requests come close to the upper limit of $40,000.

3 Time Series

We now look to model the data as a time series, with respect to the average loan amount requested per month over time. We do this to predict when demand will spike as well as when demand will fall in the future, as to be better prepared and have enough money in hand to fund these requests when needed. We want to avoid situations where a loan is fully qualified to be funded, but is not due to insufficiency in funds.

There are different components to a time series:

1. Trend: Increasing or decreasing value in the data
2. Seasonality: Repeated cycles in the data
3. Noise: Random variation in the data

3.1 Stationarity

A stationary time series is a time series in which the statistic properties remain constant over time. This includes the mean, variance, and autocorrelation. In other words, we should expect to see no trend or seasonality in a stationary time series. This is a useful assumption to make since if we can assume that the properties remain constant on past data, they will also remain constant on future data. This will allow us to do forecasting on the data, which is a key component of what we are trying to achieve with this analysis.

While the time series plot indicates a nontrivial trend, it is better to do a formal test to reach the desired conclusion. The Augmented Dickey Fuller Test (ADF) will allow us to conclusively check whether the data is stationary or not. It determines how strongly a time series is defined by a trend and/or seasonality.

Null Hypothesis: The time series is not stationary, has a unit root, and has time dependent structure.

Alternative Hypothesis: The time series is stationary, does not have a unit root, and does not have a time dependent structure.

Rejection Region: If the p-value is less than or equal to the significance level (with an alpha = 0.05), then we reject the null hypothesis.

We achieve these statistics after running ADF on our data:

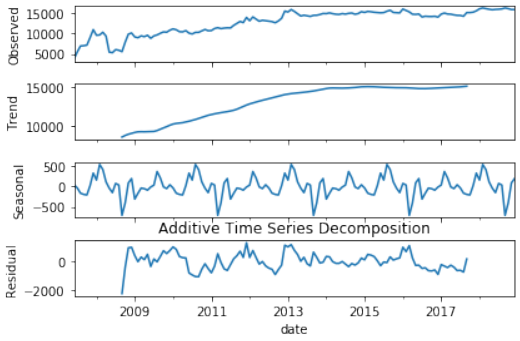
ADF Statistic = -1.235315

p-value = 0.658223

Since the p-value is greater than the 0.05 significance level, we fail to reject the null hypothesis. Therefore, the time series above is not stationary, and has a trend and/or seasonality associated with it.

3.2 Decomposition

With time series decomposition, we look to decompose a time series into the three different components: trend, seasonality, and stationary noise.



3.3 Fitting a Model to the Noise

An autocorrelation function (ACF) is a function that describes how well the current value of the series is related with the past value(s), called lags.

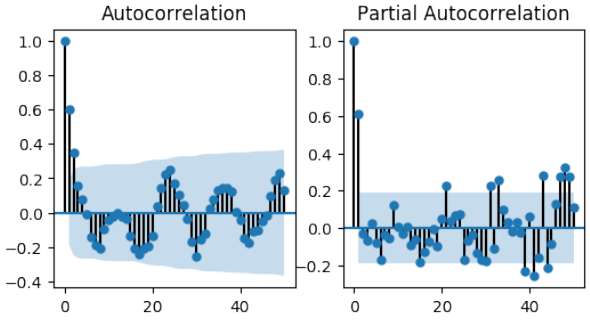
A partial autocorrelation function (PACF) is a function that describes how well the current value of the series is related with the past value(s), with the effects of earlier lags removed.

We can use the ACF and PACF plots on the stationary noise (the time series without the trend or seasonality) to find out what the optimal model would be for it.

There are in general two types of models:

The moving average (MA) model states that the current value depends on past residuals. Identification of an MA model is often done with an ACF plot, as such a time series will have non-zero autocorrelations only at lags involved in the MA model. In addition, the number of non-zero autocorrelations gives the order of the MA model.

The autoregressive (AR) model states that the current value depends on the past values. Identification of an AR model is often done with a PACF plot, as such a time series will have non-zero partial autocorrelations only at lags involved in the AR model. In addition, the number of non-zero partial autocorrelations gives the order of the AR model.



The PACF plot shows a high partial autocorrelation value for lag 1, and then it sharply drops off. This means that the model that would best fit the stationary noise would be AR(1).

With this analysis, we have broken down the different components of the time series and their overall contributions. We have determined that there is a significant trend and seasonality in the loan amounts being requested, and that the noise can be modeled by an AR(1) model. We can fit numerical models to the trend, seasonality, and noise and forecast future requests to better manage the money the bank has.

4 Data Wrangling

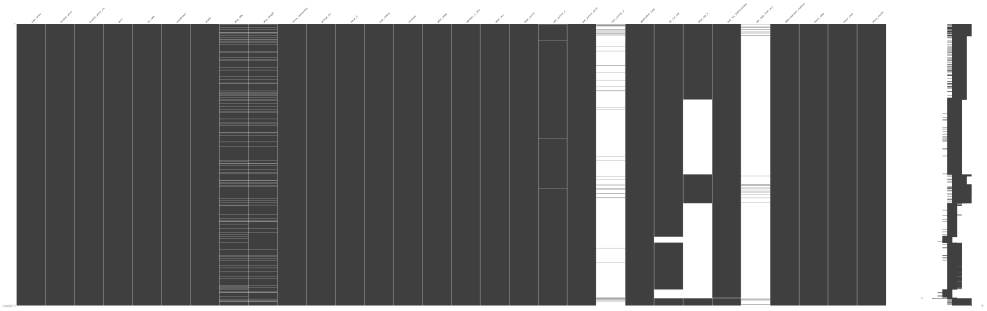
Before we jump into creating prediction models, we will have to first wrangle our data into a usable format.

We are starting off with 30 interesting and possibly impactful variables out of the 150 variables present in the dataset. Through wrangling, we will attempt to whittle that number down even more to reduce the dimensionality of the ensuing optimization problems.

4.1 NA Values

We will first look at NA values and see if there are any variables that are primarily sparse. These variables will not be useful to us due to the lack of data, and as such, we can get rid of them.

Deleting rows that have any NA values results in 2753 rows in the end, from 1,340,973 to begin with. This is a major loss of data, and as such, we should find another way to weed out these NA values.



The nullity matrix, which highlights missing data, indicates that there are three variables with most of its data missing: next_pymnt_d (next payment date), total_bal_il (total current balance of all installment accounts), and sec_app_mort_acc (number of mortgage accounts at time of application for the secondary applicant). After getting rid of these three and then repeating the step of deleting any resulting rows with NA values, we see that 1,186,550 rows persisted a lot more than before.

4.2 Variable Selection

We will elect to drop employment title altogether, since it takes on 512,698 different values Total payment is something we will not know from new customers, so we will remove this as well.

Other variables we will remove include datetime data points since they are hard to work with: last payment date, earliest credit line, and issue date.

4.3 One Hot Encoding

Our next step will be to one hot encode all remaining nominal variables since prediction models usually only work with strictly numerical data.

4.4 Normalization

We will also normalize the scales of the variables to be the same, so that the weights in the incoming prediction models could be more aptly compared with each other.

5 Predicting to Fund a Request or Not

When a loan request comes in, the bank has to decide whether it should fund the loan or not based on the data available. We intend to build a model to leverage this data to decide this automatically - a model that will minimize the risk involved while still turning over a profit.

First, let us look at what the possible different statuses are in regards to a loan:

1. Charge off: the original creditor has given up on being repaid according to the original terms of the loan. It considers the remaining balance to be bad debt, but that does not mean that they no longer owe the amount that has not been repaid.
2. In grace period: still in time to pay but late
3. Late: have not paid the full amount on time
4. Current: in process
5. Fully paid: paid on time

For the purposes of creating a model, we will remove current loans from our data since we do not know if they will be successfully paid off or not.

To be able to categorize this model, we will treat the loan statuses of the remaining rows as such: if the loan is fully paid, we will assign it 1, and if the loan has any other status, we will assign it 0. This category will be the data point that we are trying to predict.

We will be fitting a few different models to this problem - the one that performs the best will be our final model. We will be attempting to use Logistic Regression, K-Nearest Neighbors, Random Forests, and Multi-layer Perceptron.

To evaluate these models, we will be utilizing k-fold cross-validation (with a k of 10). Cross-validation is a resampling technique in which we split the dataset into k groups in order to see how the model would react to new data that it has not been trained on. Each group is in turn used as the test set, with all the other groups serving as the training set, and the model is evaluated on how well it does on the test set. A good cross-validation accuracy indicates that the model is not biased and did not overfit on the training data.

5.1 Logistic Regression

Logistic regression is quite similar to linear regression in which one attempts to fit a line through all the data points while reducing the sum of squared errors, except this is done using categorical variable(s) as output. In the binary case, we have one sigmoid-activated output variable that models the probability that it is one of the two classes. If the probability is greater than a threshold, it will classify the data point as one class, otherwise it will classify it as the other.

Logistic Regression Cross-Validation Accuracy: 92.25%

5.2 K-Nearest Neighbors

K-nearest neighbors utilizes the closest neighbors of a data point in order to classify a new sample. When a new sample needs to be classified, the k nearest data points in parameter space are fetched; distance metrics that can be used include euclidean distance, cosine distance, etc. We will then enumerate the output classes of those k neighbors, and the new data sample will be given the label with the highest number of "votes."

K-Nearest Neighbors Cross-Validation Accuracy: 76.47%

5.3 Random Forests

Random forests is an ensemble method that leverages multiple weak learners in order to form a robust classifier. To be more specific, this method builds out multiple decision trees with different random subsets of the variables and allows them to each vote on the label for a new data sample.

Random Forests Cross-Validation Accuracy: 74.33%

5.4 Multi-layer Perceptrons

Multi-layer perceptron is a type of neural network with at least three layers: the input layer, one or more hidden layers, and the output layer. It uses gradient descent in a training method called backpropagation to modify the weights in order to classify the data with minimum loss. An activation function is applied on the output of each neuron to achieve nonlinearity and add flexibility to the model.

Multi-layer Perceptron Accuracy: 74.31%

5.5 Analysis

We can see that the model with the best performance is given through logistic regression which gives us 92.25% accuracy.

We have built out a successful prediction model to determine whether a loan request should be funded or not. But now, we want to see which variables proved useful in achieving this task.

An upside with random forests is that it makes looking at feature importances relatively easy. As stated before, this method utilizes multiple decision trees internally for prediction. Each decision tree can tell us the encountered importance of each feature, and it does this by computing the normalized reductions of the loss with the addition of each variable. This is commonly known as the Gini importance. The feature importances of all the trees are averaged over to compute the overall feature importances.

	importance
int_rate	0.120967
grade_A	0.108259
installment	0.061107
grade_D	0.059611
loan_amnt	0.057280
funded_amnt_inv	0.050823
home_ownership_RENT	0.047127
funded_amnt	0.042651
home_ownership_MORTGAGE	0.027318
tot_cur_bal	0.026442
term_ 36 months	0.022965
grade_E	0.022278
grade_B	0.018466
term_ 60 months	0.015541
grade_F	0.011611
application_type_Individual	0.010548
grade_C	0.008942
annual_inc	0.008709
application_type_Joint App	0.007111

It is important to note that this only tells us the most important features for our model. This does not mean that these features are the most important in general. Our model utilizes these features to

make accurate predictions, but other models could get similar or even better accuracy while using a different set of parameters.

In predicting whether a client's request should be funded or not, the most important variable utilized is the interest rate of the loan. After that comes the loan grade, the amount that needs to be paid every installment, the loan amount requested, and the amount already funded. These factors are understandably important in determining whether a loan request should be fulfilled or not.

6 Predicting Interest Rate

We also intend to build a model to predict the interest rate that a borrower will expect to receive for a loan request. It is often the case that a bank will assign an interest rate without specifying how they determined the data point. We will analyze the most important variables in our prediction model to hopefully get a sense of what data the bank uses to determine what the interest rate for a loan should be.

By using a Multi-layer Perceptron regressor, we were able to obtain the following accuracies:

Train Mean Absolute Error: 0.892

Test Mean Absolute Error: 1.024

While this looks to be a good model, the ultimate goal of this analysis is not prediction, but analysis of the underlying factors that go into determining the interest rate.

Permutation importance is a useful method that can show us the weights of the features that are most important in the model. On every column, we will randomly shuffle it while leaving all the other columns in place, and we will observe how much the accuracy of the predictions decrease in the shuffled data.

Weight	Feature
1.8439 ± 0.0027	grade_A
0.8837 ± 0.0091	grade_B
0.2538 ± 0.0022	grade_C
0.0886 ± 0.0004	grade_E
0.0773 ± 0.0006	disbursement_method_Cash
0.0616 ± 0.0005	disbursement_method_DirectPay
0.0605 ± 0.0010	installment
0.0573 ± 0.0004	grade_F
0.0490 ± 0.0002	term_ 60 months
0.0383 ± 0.0003	application_type_Joint App
0.0375 ± 0.0001	application_type_Individual
0.0333 ± 0.0003	home_ownership_RENT
0.0286 ± 0.0004	home_ownership_MORTGAGE
0.0137 ± 0.0003	home_ownership_OWN
0.0123 ± 0.0003	term_ 36 months
0.0110 ± 0.0002	loan_amnt
0.0107 ± 0.0001	grade_G
0.0085 ± 0.0002	funded_amnt
0.0049 ± 0.0002	purpose_debt_consolidation
0.0047 ± 0.0001	funded_amnt_inv
	… 85 more …

As stated previously, it is important to keep in mind that this only tells us one set of important features, and that there could be many more subsets of the features that could give us good results.

We can see that in our model, the loan grade is the most important feature in predicting the interest rate. It is most likely the case, however, that the interest rate is a direct function of the grade in the internal calculation used by the bank. While the loan grade is a useful predictor, predicting interest rate while accounting for the grade is a trivial task due to the understandably strong relationship present between them. In addition, the loan grade is not something that a borrower has access to.

When we drop the loan grades from our model, the accuracy decreases to:

Train Mean Absolute Error: 2.772

Test Mean Absolute Error: 2.854

The most useful weights in this model are:

Weight	Feature
14.9085 ± 0.0647	installment
2.4221 ± 0.0263	funded_amnt_inv
2.0564 ± 0.0038	loan_amnt
1.9045 ± 0.0122	funded_amnt
0.9511 ± 0.0060	term_ 60 months
0.4839 ± 0.0048	term_ 36 months
0.1980 ± 0.0027	application_type_Joint App
0.1497 ± 0.0023	application_type_Individual
0.0186 ± 0.0006	home_ownership_MORTGAGE
0.0166 ± 0.0005	purpose_credit_card
0.0133 ± 0.0005	disbursement_method_Cash
0.0084 ± 0.0008	last_pymnt_amnt
0.0050 ± 0.0003	purpose_debt_consolidation
0.0048 ± 0.0002	home_ownership_OWN
0.0023 ± 0.0002	purpose_home_improvement
0.0021 ± 0.0006	home_ownership_RENT
0.0020 ± 0.0006	addr_state_TX
0.0018 ± 0.0004	addr_state_NY
0.0017 ± 0.0003	purpose_car
0.0015 ± 0.0003	emp_length_2 years
	… 78 more …

The amount one will have to pay every installment would also be heavily influenced by the interest rate, which is likely why we see such high relative importance. This is not something we will have at the time of prediction for new data samples, so we will drop this factor as well.

When we drop installment from our model, the accuracy decreases to:

Train Mean Absolute Error: 2.645

Test Mean Absolute Error: 3.758

The most useful weights in this model are:

Weight	Feature
0.1821 ± 0.0025	term_ 60 months
0.0840 ± 0.0021	term_ 36 months
0.0349 ± 0.0018	disbursement_method_Cash
0.0203 ± 0.0006	purpose_credit_card
0.0187 ± 0.0011	home_ownership_RENT
0.0092 ± 0.0006	application_type_Joint App
0.0064 ± 0.0006	disbursement_method_DirectPay
0.0040 ± 0.0002	last_pymnt_amnt
0.0023 ± 0.0001	purpose_home_improvement
0.0016 ± 0.0003	home_ownership_OWN
0.0010 ± 0.0001	purpose_car
0.0008 ± 0.0002	emp_length_1 year
0.0008 ± 0.0003	home_ownership_MORTGAGE
0.0007 ± 0.0005	emp_length_2 years
0.0006 ± 0.0003	emp_length_< 1 year
0.0005 ± 0.0004	emp_length_10+ years
0.0004 ± 0.0001	purpose_major_purchase
0.0004 ± 0.0001	pub_rec_bankruptcies
0.0003 ± 0.0001	emp_length_6 years
0.0003 ± 0.0001	emp_length_3 years
	… 77 more …

By removing the loan grades and installment factors from our model, we are able to focus on predicting the interest rates using data points that the borrower has access to. In this whittled down model, the most important factor is whether the loan is being paid off over 36 or 60 months. This makes sense, as we would expect the interest rate to be higher with a 60 month payment plan. After that, the most important variable is whether the borrower will receive their loan in cash - it might be the case that the bank charges a higher interest rate in return for getting paid in cash.