**Freddie Mac Residential Mortgage Delinquencies**

DS 5110, Fall 2021

Thomas Butler (vra2cf), Andrej Erkelens (wsw3fa) , Matt Suozzi (mds5dd)

***Abstract***

The government sponsors home loans through the Federal Home Loan Mortgage Corporation or “Freddie Mac”. These allow potential home buyers to easily access mortgages. Freddie Mac has kept data on their loans since 1999. We have taken this data from January 1, 2019 to March 31, 2021 and tried to predict delinquency in these loans. We analyzed the data and built 4 different models: Simple Logistic Regression, Multivariable Logistic Regression, Random Forest, and Gradient Boosted Tree. Simple Logistic Regression used Interest Rate to model delinquency. The multivariable models used pipelines to transform the features - FICO Score, Loan Purpose, Number of Borrowers, Occupancy Status, Modification Flag, Combined Loan-to-Value CLTV, Original Debt-to-Income DTI Ratio, Interest Rate, and Loan Term - to predict delinquency. Simple Logistic Regression underperformed the other models. Multivariable Logistic Regression, Random Forest, and Gradient Boosted Tree all had similar AUROCs (0.683-0.7175). We concluded that Multivariable Logistic Regression, Random Forest, and Gradient Boosted Tree are all good models for predicting delinquency but more data could be collected to improve them.

***Introduction***

The Federal Home Loan Mortgage Corporation, also known as “Freddie Mac”, is a government-sponsored enterprise that provides a secondary market for residential mortgages. The organization effectively increases the money supply that mortgage lenders have to issue new loans, by buying mortgages, packaging them into mortgage-backed securities, and then selling them to private investors.

Since 1999, Freddie Mac has published quarterly reports on loan origination and performance data[[1]](#footnote-0). The origination reports contain data on new loans, including underwriting metrics, repayment terms, and servicing information. The performance reports include monthly payment activity or accrued interest and losses for each loan from origination through March 31, 2021.

For the project, the origination data spanned from January 1, 2019 through September 30, 2020, with loan payment history and delinquency status as of March 31, 2021. During this time period, the Covid-19 pandemic impacted the United States jobs and housing markets in different ways. While unemployment increased, home purchasing volume and housing prices skyrocketed.

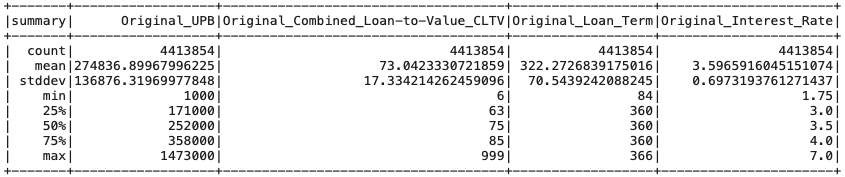
Using the combined origination and performance data sets, we will construct classification models to predict loan delinquencies. From the sample of 4.4 million unique loans with a total value of $1.2 trillion, improving portfolio quality and preventing losses can have meaningful economic impact.

***Data & Methods***

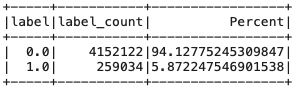
*Data Overview*

The data was sourced from the Freddie Mac website and downloaded as 14 separate text files (seven quarterly origination text files, and seven quarterly performance text files). The origination and performance files were each combined into two dataframes for data exploration, cleaning, and pre-processing, before merging into a final dataframe and saving as a parquet file.

The combined origination Dataframe included 4,413,854 rows and 31 columns, with each row identifying a unique Loan Sequence Number. The columns included numerical underwriting data (FICO score, Loan-to-Value, Debt-to-Income), categorical data (First Time Home Buyer, Occupancy Status, Loan Purpose, Location), and loan structure data (Interest Rate, Term Length, Amortization Type).

**

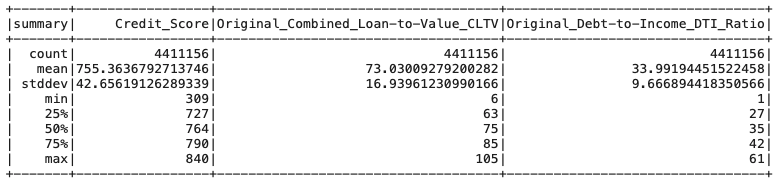
The combined performance data frame included 53,024,475 rows and 31 columns, with each row representing the monthly payment activity for a loan. The columns included Loan Delinquency Status, Loan Age, Remaining Months to Legal Maturity, Delinquent Accrued Interest, and Actual Loss Calculation. Each loan possibly had multiple rows to reflect the payment history, which could be classified as current or delinquent. Therefore, the data frame was filtered to include only the most recent payment and status for each loan, reducing the performance data frame to 4.4 million rows, to match the origination Dataframe.

**

While the origination data frame included the predictor features, the performance data frame contained the response variable - Loan Delinquency Status. To predict delinquent loans, the two dataframes were joined based on the Loan Sequence Number. Delinquent loans are a relatively rare event, at an approximate rate of 5.9%.

*Pre-processing*

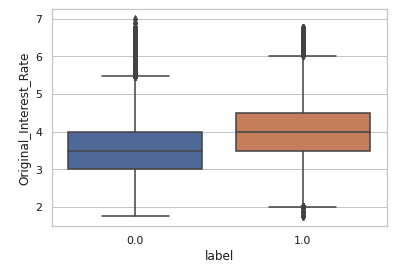
As shown in the summary table of the origination file, some of the predictor features contained erroneous values or missing data. For example, the maximum value for Combined Loan-to-Value of 999 represents an error. Similarly, records with errors in Debt-to-Income or Credit Score were eliminated from the final data, which reduced the available records by just 2,698, or <0.1%.



Furthermore, Credit Score was converted to a categorical feature, based on the traditional value ranges for ‘Very Poor’, ‘Fair’, ‘Good’, ‘Very Good’, and ‘Excellent’ credit. The ‘Bucketize’ transformation in the Pyspark ML library conveniently transformed Credit Score to dummy variables of 0-4: 0:300-579 (‘Very Poor’), 1:580-669 (‘Fair’), 2:670-739 (‘Good’), 3:740-799 (‘Very Good’), and 4:800-850 (‘Excellent’).

*Variable Selection*

The final merged data frame contained the following features: Credit Score, Loan-to-Value, Debt-to-Income, Interest Rate, Loan Term, Loan Purpose, Number of Borrowers, Occupancy Status, and Modification Flag. Interest rate was identified as one of the better features that delineated delinquency. Based on the boxplot below, loans that became delinquent (label 1.0) had a median interest rate that was 50 basis points higher than current loans.

**Current**

**Delinquent**

Interest rate effectively is an engineered feature, since the underwriter structured the loan based on the perceived risk of the available information, including Credit Score, Loan-to-Value, and Debt-to-Income. It is possible that the underwriter priced in other risk factors that are not presented in the dataset.

For future work, additional feature engineering may improve model performance. The current data could create ‘bucketized’ variables for loan size, or one-hot-encoded variables for loans originated before the pandemic, that might provide additional insight into loan delinquency.

Other variables did not provide meaningful insight due to the skewed distribution and high multicollinearity. For example, Occupancy Status and Modification Flag are questionable variables to include in the models, since they are heavily classified to one class. Occupancy Status is classified as a ‘Primary Residence (P)’ 91% of the time. Modification Flag is classified as ‘No’ 99.99% of the time.

***Model Training***

Four different models were considered: Simple Logistic Regression, Multivariable Logistic Regression, Random Forest, and Gradient Boosted Tree. Preprocessing cleaned and filtered the data, so there were no duplicate loans. Loans that were past due or delinquent were the positive class (1), while current loans were the negative class (0). Train and test data was split evenly; however, the training data was downsampled to balance the proportion of current loans (94% of the data) to delinquent loans (6%). Therefore, the models were trained on stratified data of 259K loans, with equal samples of current and delinquent loans. The test data set was not downsampled. Each model utilized pipelines to transform the features and cross-validation to tune the model parameters.

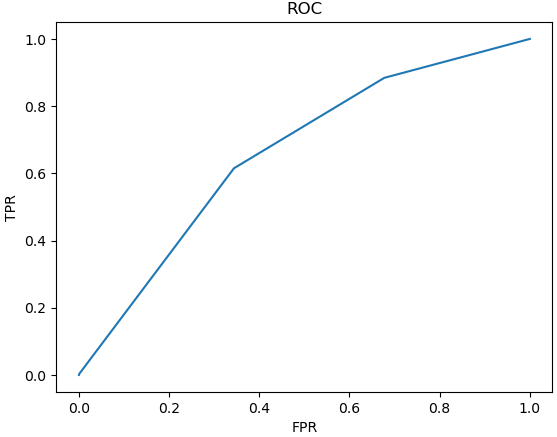
Models were evaluated based on the results of the confusion matrix and the intent of the model. In order to maximize loan interest revenue, the threshold could be set to increase the number of current loan predictions, while risking a higher number of loans predicted as current that actually are delinquent. In order to minimize loan delinquency, the threshold could be set to increase the number of delinquent loan predictions, while risking the opportunity cost of not underwriting current loans.

***Simple Logistic Regression Model***

For a single variable logistic regression model, Interest Rate was selected as the key feature, as it effectively is an engineered feature, since the underwriter would structure and price the loan based on the risk. The Interest Rate was scaled with the MaxAbsScaler function to preserve sparsity of the data. There was no missing data. This was put into a pipeline and fit to the downsampled training data.

A 5-fold logistic regression cross-validated (CV) model was made testing for 3 different parameters: regParam, elasticNetParam and maxIter. Please check the “Best Model’s Parameters” in the “Tabled Results” section below for the Simple Logistic Regression’s parameters. A threshold was chosen using the CV model by maximizing the F1 Score with the training data. This threshold was run on the test data. Please check the “Test data metric” in the “Tabled Results” section below for the metrics on Simple Logistic Regression. The confusion matrix is in the confusion matrix section.

The Receiver Operating Characteristic curve was generated for different thresholds to measure the trade-off of True Positive Rate and False Positive Rate.



ROC Curve for Simple Logistic Regression

The equation for this model is:

***Multivariable Logistic Regression Model***

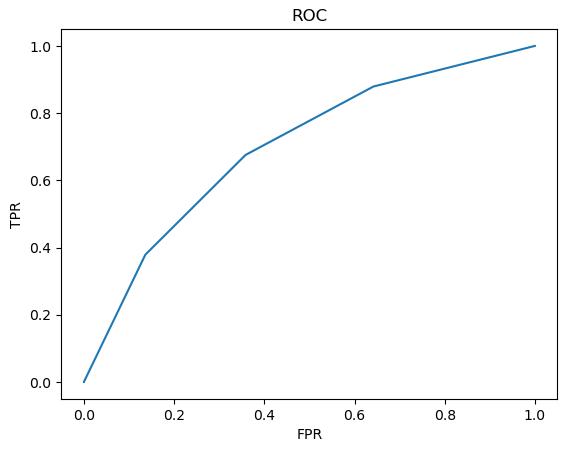
A multivariable logistic regression model on Delinquency incorporated the additional underwriting metrics and loan features, including: FICO Score, Loan Purpose, Number of Borrowers, Occupancy Status, Modification Flag, Combined Loan-to-Value CLTV, Original Debt-to-Income DTI Ratio, Interest Rate, and Loan Term.

Loan Purpose, Number of Borrowers and Occupancy Status are categorical data that was one hot encoded. Modification Flag was made into a binary variable where 1 = Yes and 0 = No. Combined Loan-to-Value CLTV, Original Debt-to-Income DTI Ratio, Interest Rate, and Loan Term were scaled with MaxAbsScaler to maintain sparsity. All data besides FICO Score didn’t have any missing data. These methods were put into a pipeline and run with the full data.

First Time Homebuyer Flag, Loan Age, Remaining Months to Legal Maturity, and Change of Interest Rate were not included in the model. First Time Homebuyer Flag wasn’t included because of a >65% missing data values. Loan Age and Remaining Months to Legal Maturity were a time metric based on Loan Term and were superfluous since we were trying to include only values known before the loan was given. Change of Interest Rate wasn’t included because values known before the loan was given were only included.

A 5-fold logistic regression cross-validated (CV) model was made testing for 3 different parameters: regParam, elasticNetParam and maxIter. Please check the “Best Model’s Parameters” in the “Tabled Results” section below for the multivariable logistic regression’s parameters. A threshold was chosen using the CV model by maximizing the F1 Score with the training data. This threshold was run on the test data. Please check the “Test data metric” in the “Tabled Results” section below for the metrics on multivariable logistic regression. The confusion matrix is in the confusion matrix section.

The Receiver Operating Characteristic curve was generated for different thresholds to measure the trade-off of True Positive Rate and False Positive Rate.



ROC Curve for Multivariable Logistic Regression

The equation for this model is:

Where x = -0.9275 - 0.2011\*FICO\_Score - 0.0484\*Loan\_Purpose Category "N" + 0.0348\*Loan\_Purpose Category "P" + 0.1338\*Number\_of\_Borrowers Category "01" - 0.0971\*Occupancy\_Status Category "P" - 0.0837\*Occupancy\_Status Category "I" + 0.6089\*Modification\_Flag\_Bin + 0.1825\*Original\_Combined\_Loan-to-Value\_CLTV + 0.6049\*Original\_Debt-to-Income\_DTI\_Ratio + 1.6621\*Original\_Interest\_Rate + 0.1111\*Original\_Loan\_Term

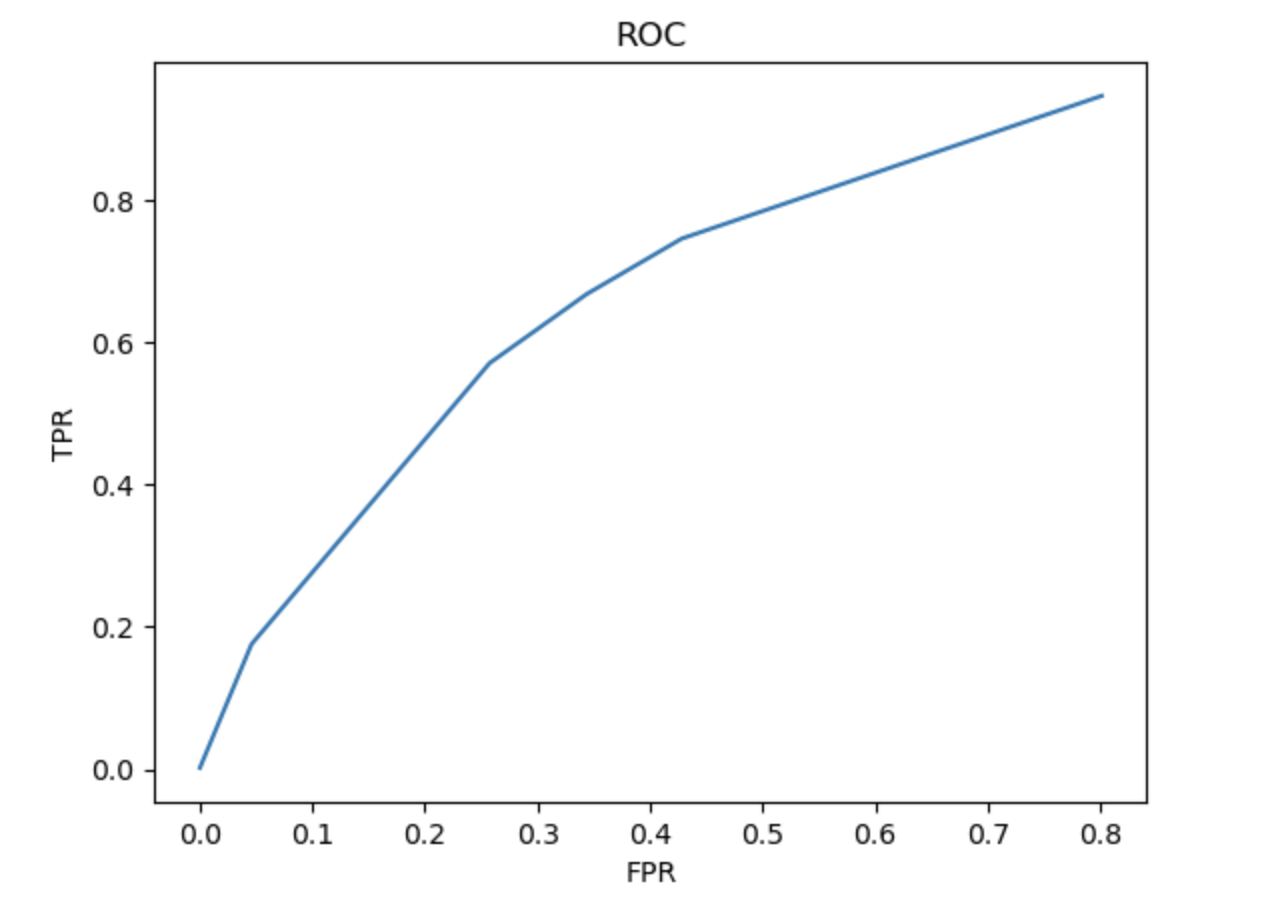
Note all variables besides FICO\_Score range between [-1,1]. FICO\_Score is a bucketed categorical variable of (0,1,2,3,4).

***Gradient Boosted Trees***

Our gradient boosted tree model using the same features and pipeline as the multivariate logistic regression. (For consistency in experiments and comparison)

We performed a 10 fold cross validation on the data using a parameter grid to tune the subsampling rate, max depth (of tree), feature subset strategy (using square root of feature in each selection vs. all), and step size (learning rate). The top parameters from the CV were 1.0 subsampling rate, .1 step size, all features used, and max depth of 5.

This set of hyperparameters needed to be approached with caution, as using all of the features and the full sub sample could be prone to overfitting. When testing these parameters on the test set we had very similar results to the other models and sets of parameters. This further emphasized the plateau we achieved in modeling the data set.

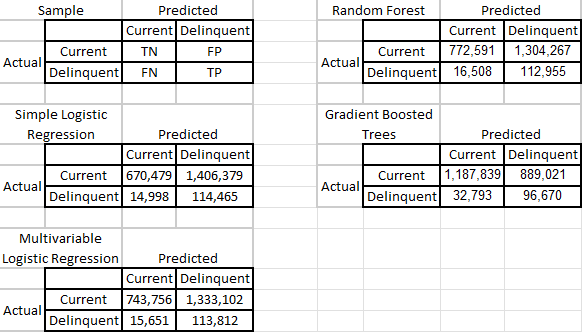


***Results***

***Confusion Matrix***

The model could be used to maximize loan revenue or minimize losses:

* True Negative (0,0): loans predicted as current that are actually current -> made money
* False Positive (0,1): loans predicted as delinquent that are actually current -> did not make money.
* False Negative (1,0): loans predicted as current that are actually delinquent -> lost money
* True Positive (1,1): loans predicted as delinquent that are actually delinquent -> did not lose money



***Tabled Results***

| **Best Model's Parameters** | | | | |
| --- | --- | --- | --- | --- |
| Model | **Simple Logistic Regression** | **Multivariable Logistic Regression** | **Random Forest** | **Gradient Boosted Trees** |
| regParam | 0.5 | 0.5 | N/A | N/A |
| elasticNetParam | 0 | 0 | N/A | N/A |
| maxIter | 10 | 10 | 10 | 20 |
| maxDepth | N/A | N/A | 6 | 5 |
| numTrees | N/A | N/A | 100 | 20 |
| subsamplingRate | N/A | N/A | N/A | 1 |
| featureSubsetStrategy | N/A | N/A | N/A | all |
| stepSize | N/A | N/A | N/A | 0.1 |
| maxBins | N/A | N/A | N/A | 10 |

Any other hyperparameter not included is the default parameter.

| **Test Data Metric** | | | | |
| --- | --- | --- | --- | --- |
| Model | **Simple Logistic Regression** | **Multivariable Logistic Regression** | **Random Forest** | **Gradient Boosted Trees** |
| Best Threshold | 0.462 | 0.44 | 0.33 | 0.5 |
| Accuracy | 0.356 | 0.3887 | 0.401 | 0.6569 |
| Precision | 0.075 | 0.0787 | 0.0797 | 0.108 |
| F-Score | 0.485 | 0.524 | 0.5392 | 0.782 |
| Negative Predictive Value | 0.978 | 0.979 | 0.979 | 0.969 |
| AUROC | 0.683 | 0.714 | 0.7095 | 0.7175 |
| Time to run |  |  | 2674.76 |  |

***Future Work***

Unfortunately, our group ran out of time to select the best threshold for Gradient Boosted Trees so we selected the default value. Future work should run the model with multiple thresholds to determine which model has the best F1 score on the training data.

The classification model can be expanded from binary to multiclass to capture added risk and reward. For example, a loan with a 60% probability of becoming delinquent can be given a “Speculative” rating to suggest that an underwriter can make the loan for additional fees or collateral protection.

Gather additional data on loans that defaulted rather than late on payments. There were 44 loans that went to acquisition in our dataset and getting more data and making that a separate class will greatly increase the worst case scenario of a loan defaulting, especially if we choose to take the more risky loans that have late payments.

Additionally, future enhancements to the model could include other functions to predict overall interest income and loan losses. The Threshold was chosen with the training set by maximizing the F1-score. This is actually not the best method especially since our actual data is heavily skewed. It would actually be better to know how much a loan would make or how much a delinquent loan would lose, on average, then model for maximized profit.

Where Safety Factor is a positive number chosen based on how much safety on profit we want.

We would run this equation for every threshold and choose the threshold where Profit is largest. If the largest Profit is negative either the Safety Factor needs to be looked at and possibly lowered or the model is not good and needs to be examined.

This method would be even more effective if another model was created which predicted how much money was expected to be made on a particular loan and how much would be lost if the loan went delinquent. These values would be put into the equation below with the prediction values from another model.

Alternatively we could create a model which predicts likelihood of a loan default at each payment date and expected profit/loss at each time, then a summation can determine expected total profit of loan.

*t* = payment date

Data on exact loan $ amounts and expected profits on completion of loan and losses at different times out from the loan would help predict loan profits and losses at different time periods.

Since both of these methods require the work of at least 2 different models and more data that we don’t have access to, this would be a good candidate for future work.

For future work, cross-validation could be expanded to further check our hyperparameters. However, checking a larger amount of parameters, particularly larger value parameters, greatly increases the computation cost of all the CV models which is why we only check a few from each model as our time was better spent on improving other aspects of the project.

From a modeling perspective an interesting observation was that each of the models which we trained, from simple to complex, produced nearly identical AUROC. To explore this we cycled some features in and out of the pipeline to see if there was a confounding variable we were missing. This sort of plateau could indicate something variable missing from the data or simply an unexplainable result in the context of the mortgage loans. Regardless, the plateau was an interesting observation and indicated the key hurdle to achieving further improvements. It would be of interest to the lenders. This result could provide a basis for assisting mortgage lenders in providing an extra evaluation criterion. Future work should look into additional features or data which could improve the performance and explainability of our models.

***Conclusions***

As the information presented in the data sets have not changed since 1999, it does not appear that mortgage lenders have updated the underwriting approach since machine learning algorithms became available. Freddie Mac does not appear to have taken advantage of the technology and the data available to improve underwriting models. While Credit Score, Loan-to-Value, and Debt-to-Income have been reliable metrics, there may be important features that were not included in the data set. It should be noted that there are regulations in place that prevent mortgage lenders from discriminating based on race or gender, better known as redlining, which can correlated to geographic region, job history, or expected income growth. This does make implementing these models more difficult as opposed to proven older methods.

In this project we used Pyspark’s MLlib package to build several different machine learning pipelines aiming to predict mortgage loan delinquency. After a number of feature engineering and selection techniques we ended up with a series of models with similar, but solid performance. This may show a need for more data to improve performance and explainability of our models. Regardless, these models are a tool and guideline that allow lenders to help determine which loans to accept or reject.

1. [Freddie Mac Data](http://www.freddiemac.com/research/datasets/sf_loanlevel_dataset.page) [↑](#footnote-ref-0)