Predicting Loan Level Mortgage Loss

An exploration of machine learning methods

Chris Walker

Table of contents

1 Introduction

This document describes the motivation, data, estimation, and validation of a credit loss model which compares non-parametric machine learning techniques to predict realized mortgage-level losses in basis points (Bps) of origination UPB. Moreover, this model is designed to estimate *gross loss* which represents risk to the *financial system* and not any one party (Freddie Mac, mortgage insurers, etc). The motivations for this project are twofold:

1. Study the relationship between common loan application details (credit score, debt-to-income ratio, loan-to-value ratio, etc) and mortgage losses.
2. Evaluate a range of machine learning methods which can be used to model complex regression problems, many of which contain non-linear relationships.

Obtaining good estimates of *expected gross loss in a stress scenario* is useful for healthy management of the financial system, particularly when losses are estimated early in the mortgage life cycle such as the time of application. It is worth noting that extreme mortgage losses are rare and are generally contained to stressed economic periods, including the great recession period. Thus, this model aims to predict loan-level loss for a mortgage *given* a stress scenario.

First, a few definitions:

Term	Abbreviation	Definition
debt-to-income	DTI	Ratio of debt to a person's income
combined loan-to-value	CLTV	Ratio of loan balance to property value
credit score	FICO	Representation of credit history risk
basis-points	Bps	0.01% of a percent
unpaid balance	UPB	Loan balance to be paid
mortgage insurance	MI	Insurance paid to offset losses
delinquent	DQ	Failure to make payments

2 Data

2.1 Target Variable

Data for this project comes courtesy of Freddie Mac's loan level data repository. This data repository contains anonymous mortgage data in two primary components; origination data and over time data. Origination data contains information about a loan *at its origin*. This contains fields like the borrowers' UPB, FICO, LTV, and DTI when the loan was created. Alternatively, the over time data contains monthly summaries of mortgage activity as a borrower pays down their mortgage. This over time data contains information about losses. A mortgage loss occurs when a borrower fails to make payments and enters delinquent status. The financial system incurs losses when borrowers become delinquent for an extended period of time.

We compute loss as *risk to the financial system*. This means that it is possible to compute the amount of losses incurred by Freddie Mac (for example, losses minus insurance payments) but instead we compute total (or gross) losses incurred by every involved party. Our loss computation follows this general form:

$$\text{Gross Loss} = \text{Delinquent Interest} + \text{Current UPB} + \text{Expenses} - \text{Net Sale}$$

Delinquent interest is loan interest which accumulates when a borrower fails to make payments. Similarly, Current UPB denotes the loan balance at the time of delinquency. Once Freddie Mac repossesses the property they will attempt to sell the property. When the property finally sells much of the original loss is recouped (and sometimes a profit is made). However, in most cases there is still a gross loss balance which is positive representing loss to the financial system. These losses can be offset by MI assistance, but they are not included in this calculation.

$$\text{Gross Loss Bps} = 10000 \times \frac{\text{Gross Loss}}{\text{Origination UPB}}$$

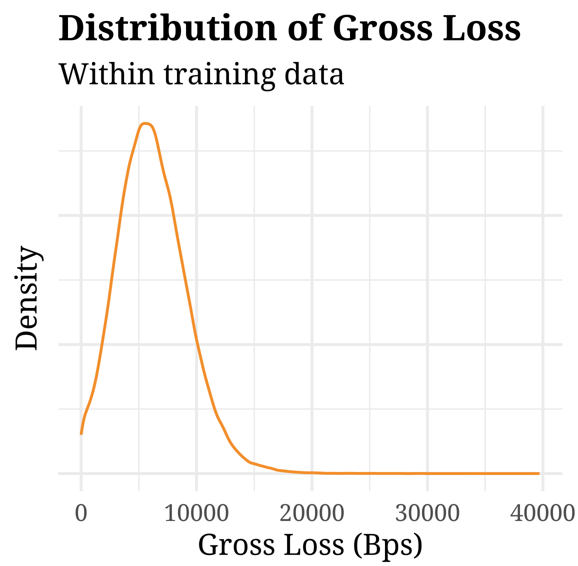
We divide gross loss values by the original loan balance (origination UPB) and convert them to Bps. This provides us a value representing the percentage of a loan's original value which we might *expect* to be lost given a delinquency in a stressed economic scenario. Because we wanted to simulate losses in a stressed environment, we utilized loans originated right before the financial crisis (2007 acquisitions). However, not all loans in this population experienced a loss. Using the user guide provided by Freddie Mac, we restrict to loans which satisfy these conditions:

- Is a third party sale, short sale or charge off, REO disposition, or whole loan sale
- Does not have a populated defect settlement date
- Has all components of the loss calculation present (not missing)

The Freddie Mac data does include a *net loss* calculation as part of the data set. We compute our target variable from scratch because we want to measure risk to the financial system. However, the included net loss value is set to zero if one of the last two conditions listed above is not present. Thus, the training data for this model applies the following two conditions:

```
zero_bal_code in ('02', '03', '09', '15')
and net_loss != 0
```

Our target variable has the following distribution:



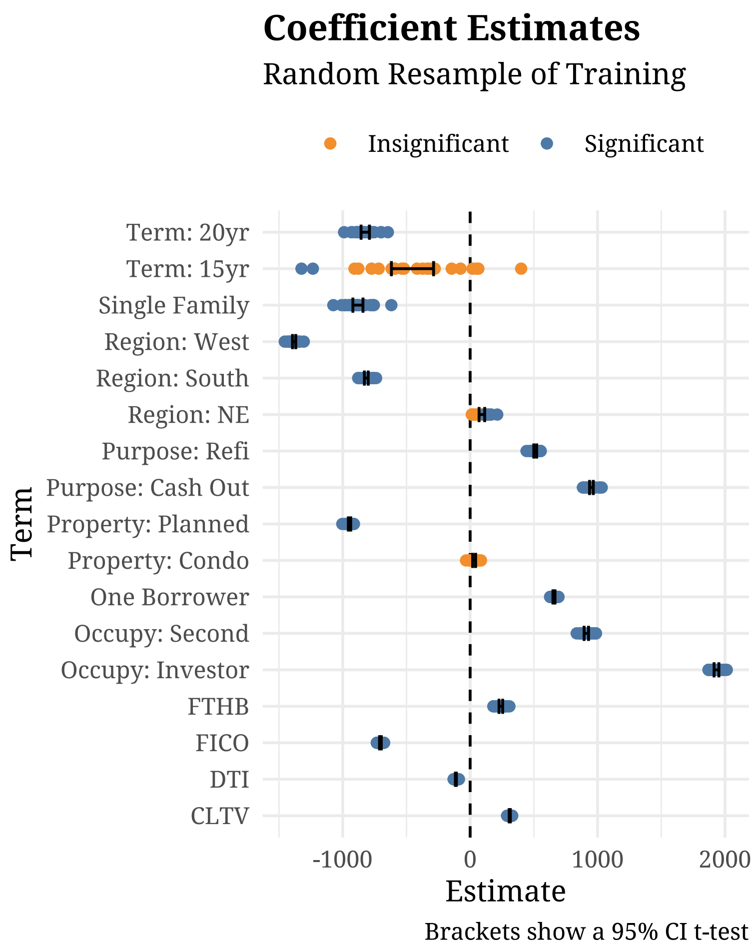
2.2 Independent Variables

This model makes use of loan level application characteristics to predict loss in basis points in origination UPB given a stressed economic scenario. While more sophisticated models can be estimated when more data is present (for example, early payment history data) we intentionally constrain this model to application level details to act as an *early warning loss model*. Fields which are present at the time of application include CLTV, DTI, FICO, occupancy status, property type, loan purpose, loan term, first time home buyer, and geographic region.

CLTV, DTI, and FICO are continuous values. This means that these values are a number which is intended to have a positive or negative relationship with gross loss. For example, as FICO scores increase, we would expect gross loss values to fall. All other predictors are categorical variables which means they indicate one of several options. For example, someone can live in one of four regions (but not more than one at once).

Here we perform a parameter stability exercise before leveraging more interesting machine learning methods. This method begins by randomly sampling the training data with replacement 100 times. From there, we estimate linear models and assess the stability of the coefficients. There are no additional transformations or splines applied to the continuous predictors, however the categorical variables are one hot encoded.

This means that the *default* values do not show up on the visual below. The coefficients measure the marginal increase or decrease over the default values for our categorical variables. For example, the default term is 30 years thus the two terms which show 15 and 20 years represent the marginal increase or decrease in expected loss *over* the 30 year default category.

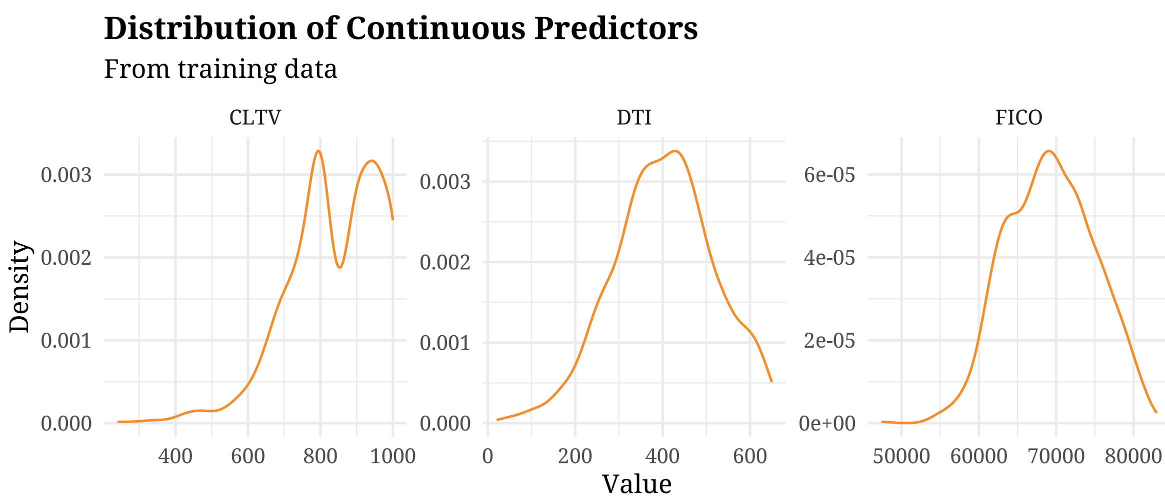


Most coefficients show relative stability with a tight confidence interval. Notable exceptions include 15 year mortgages, the northeast region, and condos. This means that the expected loss of these terms when compared to the defaults are not meaningfully different. Many of these estimations reveal sign flips and insignificant coefficients. The default categories for these terms are 30 years, the Midwest, and single family homes, respectively. This means we will treat these unstable categories as the defaults.

One other variable which might be unstable is DTI. This term tends to correspond with short term default over long term loss. While its confidence interval is small and it is always significant, the term has a negative coefficient which means we would expect losses to fall with increased DTI, which is counter intuitive.

All things considered, the primary terms for this model will be mortgage term, geographic region, loan purpose, property type, one unit, one borrower, occupancy status, first time home buyer, FICO, and CLTV. We will monitor the importance and direction of DTI across each learning method.

In the figure below we plot the distribution of each continuous predictor. Both DTI and FICO are *approximately* normal. The notable exception is CLTV which has large clusters at 80% and 100%. There is incentive for borrowers to put at least 20% down on their mortgage to avoid monthly mortgage insurance payments. Similarly, the cluster around 100% are borrowers who put very little down on their mortgage but still want to purchase a home.



Our complete data set contains 98489 records. We split the data into train and test. The train population is used for all model fitting procedures. We then use resampling approaches for interim steps (like hyperparameter tuning). Finally, we use the test population for final comparison between models.

Population	Observations
Test	24623
Train	73866

3 Learning Methods

3.1 Linear Regression

3.1.1 Background

Our first modeling method is traditional ordinary-least-squares (OLS) regression, also known as linear regression. This method fits an additive model to the data by finding the coefficients or parameters which minimize the sum of squared residuals (or error) between actual and predicted values. However, for continuous predictors with non-linear relationships, we can improve the fit of the model via piece wise splines. This allows the linear plane to have bends or knots which form a jagged shape that can better fit the data.

Lets begin by discussing a brief theoretical background for linear regression or OLS. Arguably the most fundamental and widely used modeling technique, OLS generates a numeric value (called a coefficient) for each variable in the model. It also generates a constant value (also called the intercept) which is used to correct the overall level of the model predictions. Linear regression makes predictions using the following structure:

$$\hat{y}_i = \beta_0 + \sum_{j=1}^{p} X_{ij}\beta_j$$

Here we want to get the prediction for observation i denoted as $\hat{y}_i$. The prediction is the intercept (denoted as β_0) plus the sum of p terms. In this case p refers to the number of predictors. We multiply each predictor for observation i (denoted as X_{ij}) by the beta for that term (denoted as β_p).

OLS attempts to minimize the sum of squared residuals. A residual is simply the difference between the actual value for this observation and the predicted value which we can show as $\hat{y}_i - y_i$. The sum of squared residuals is:

$$\text{SSR} = \sum_{i=1}^{n} (\hat{y}_i - y_i)^2$$

The values for β_j can take the form of any real number. Generalized models make use of a search process called gradient descent to determine the best values for each parameter. However, for OLS we can actually *solve* for these values making linear models very quick to estimate! It takes this form:

$$\beta = (X^T X)^{-1} X^T y$$

Note that here we refer to the model estimates as β. In truth these should be denoted as $\hat{\beta}$ because the coefficients themselves are estimates but we adopt this convention for this report. With our model in hand the last step is to estimate p-values for each term, including the intercept. P-values come from estimating standard errors for each term. We can then use these values to conduct t-tests for each term and assess the statistical significance of each β. Ideally, p-values are very small because small p-values indicate that there is a good chance that the true coefficients (as opposed to the estimated coefficients) are something other than zero and are meaningful.

3.1.2 Estimation

To construct our splines we will leverage the `earth` package in R which relates to multivariate adaptive regression splines (MARS). Running MARS over our train population results in one knot recommendation per continuous predictor. They turn out to be 720 for FICO, 80 for CLTV, and 37 for DTI. We reject the DTI knot because of DTIs already small (and somewhat counter intuitive) coefficient. Our OLS model has the following specifications:

Term	Estimate	P.Value
Intercept	7021	<0.001
FTHB	265	<0.001
Single Family	-901	<0.001
DTI	-105	<0.001
One Borrower	659	<0.001
Occupy: Investor	1887	<0.001
Occupy: Second	869	<0.001
Property: Planned	-960	<0.001
Purpose: Cash Out	881	<0.001
Purpose: Refi	464	<0.001
Term: 20yr	-819	<0.001
Region: West	-1404	<0.001
Region: South	-836	<0.001
FICO < 720	-845	<0.001
FICO > 720	-412	<0.001
CLTV < 80	423	<0.001
CLTV > 80	171	<0.001

3.2 XGBoost

3.2.1 Background

3.2.2 Estimation

3.3 Generalized Additive Model

3.3.1 Background

3.3.2 Estimation

3.4 Neural Network

3.4.1 Background

3.4.2 Estimation

4 Performance Comparison

Population	Model	Gini	TMR 5%	RMSE
Train	XGBoost	0.622	26.3%	2868
Train	Torch	0.604	26.4%	2768
Train	Neural Network	0.596	26.2%	2771
Train	Generalized Additive Model	0.585	25.4%	2791
Train	Linear Regression	0.583	25.1%	2795
Test	XGBoost	0.598	26.5%	2872
Test	Torch	0.590	27.0%	2757
Test	Neural Network	0.582	27.3%	2761
Test	Generalized Additive Model	0.577	26.1%	2781
Test	Linear Regression	0.576	25.6%	2783

5 Recommendation

6 References