Research, Society and Development, v. 13, n. 10, e34131047092, 2024
(CC BY 4.0) | ISSN 2525-3409 | DOI: http://dx.doi.org/10.33448/rsd-v13i10.47092

Machine learning Bureau score for Home Lending in an American finance company

Pontuação do Machine Learning Bureau para empréstimos imobiliários em uma empresa financeira americana

Puntuación Bureau usando Aprendizaje Automático para Préstamos Hipotecarios en una empresa financiera estadounidense

Received: 10/01/2024 | Revised: 10/05/2024 | Accepted: 10/05/2024 | Published: 10/11/2024

Lucero Isabel Izquierdo Munoz
ORCID: https://orcid.org/0009-0009-8287-1064
Purdue University, United States
E-mail: lizquier@purdue.edu
Jose Manuel San Martin Galindo
ORCID: https://orcid.org/0009-0003-7855-7255
Purdue University, United States
E-mail: jsanmar@purdue.edu

Abstract
Our client is a leading provider of mortgage financing, originating loans and lines of credit to consumers in the US. Currently, they receive applications where applicants provide personal information and a soft pull of their FICO score is requested. That score is used to evaluate the applicant's credit worthiness and determine conditional approval and the type of product available for the customer, including conventional, FHA or other mortgage loans. After conditional approval, a formal application is initiated, and underwriters review the information to determine the final application decision. When evaluating applications below regulatory and business thresholds, the company has the intention to approve more applications and increase loan volume, and there is an expectation that through the enhanced credit assessment, our client will improve the percentage of Low to Moderate Income (LMI) population able to obtain mortgage loans. Both aspects have a direct impact on the reputation and economic profits of the firm, so they are of pressing importance to the company. This project aims to build an applicant-level bureau-only score based on upgraded bureau internal attributes. This score will eventually serve as the basis for evaluating a customer's credit risk before any loan structure or collateral information is considered. It will be used as a standalone score that can be used in the initial customer evaluation to identify better leads (mortgage inquiries for preapproval) and as input to a future application-level model.
Keywords: Machine learning; Predictive modeling; Home lending; Credit bureau; LMI consumers; Credit risk; Mortgage application.

Resumo
Nosso cliente é um fornecedor líder de financiamento hipotecário, originando empréstimos e linhas de crédito para consumidores nos EUA. Atualmente, eles recebem solicitações em que os candidatos fornecem informações pessoais e uma verificação suave de sua pontuação FICO é solicitada. Essa pontuação é usada para avaliar a capacidade de crédito do candidato e determina a aprovação condicional e o tipo de produto disponível para o cliente, incluindo empréstimos hipotecários convencionais, FHA ou outros. Após a aprovação condicional, uma solicitação formal é iniciada e os subscritores revisam as informações para determinar a decisão final da solicitação. Ao avaliar solicitações abaixo dos limites regulatórios e comerciais, a empresa tem a intenção de aprovar mais solicitações e aumentar o volume de empréstimos, e há uma expectativa de que, por meio da avaliação de crédito aprimorada, nosso cliente melhore a porcentagem da população de baixa a moderada renda (LMI) capaz de obter empréstimos hipotecários. Ambos os aspectos têm um impacto direto na reputação e nos lucros econômicos da empresa, portanto, são de importância premente para a empresa. Este projeto visa construir uma pontuação somente de agência em nível de candidato com base em atributos internos de agência atualizados. Essa pontuação eventualmente servirá como base para avaliar o risco de crédito de um cliente antes que qualquer estrutura de empréstimo ou informação de garantia seja considerada. Ela será usada como uma pontuação autônoma que pode ser usada na avaliação inicial do cliente para identificar melhores leads (consultas de hipoteca para pré-aprovação) e como entrada para um futuro modelo de nível de aplicativo.
Palavras-chave: Aprendizado de máquina; Modelagem preditiva; Empréstimos imobiliários; Agência de crédito; Consumidores LMI; Risco de crédito; Pedido de hipoteca.

Resumen

Nuestro cliente es un proveedor líder de financiación hipotecaria, originación de préstamos y líneas de crédito para consumidores en los EE. UU. Actualmente, reciben solicitudes en las que los solicitantes proporcionan información personal y se solicita una extracción suave de su puntaje FICO. Ese puntaje se utiliza para evaluar la solvencia crediticia del solicitante y determina la aprobación condicional y el tipo de producto disponible para el cliente, incluidos los préstamos hipotecarios convencionales, FHA u otros. Después de la aprobación condicional, se inicia una solicitud formal y los suscriptores revisan la información para determinar la decisión final sobre la solicitud. Al evaluar las solicitudes por debajo de los umbrales regulatorios y comerciales, la empresa tiene la intención de aprobar más solicitudes y aumentar el volumen de préstamos, y existe la expectativa de que, a través de la evaluación crediticia mejorada, nuestro cliente mejorará el porcentaje de la población de ingresos bajos a moderados (LMI) capaz de obtener préstamos hipotecarios. Ambos aspectos tienen un impacto directo en la reputación y las ganancias económicas de la empresa, por lo que son de gran importancia para la empresa. Este proyecto tiene como objetivo construir un puntaje exclusivo de la agencia a nivel de solicitante basado en atributos internos de la agencia mejorados. Esta puntuación servirá como base para evaluar el riesgo crediticio de un cliente antes de considerar cualquier estructura de préstamo o información sobre garantías. Se utilizará como una puntuación independiente que se puede utilizar en la evaluación inicial del cliente para identificar mejores oportunidades (consultas de hipotecas para aprobación previa) y como información para un futuro modelo a nivel de solicitud.

Palabras clave: Aprendizaje automático; Modelado predictivo; Préstamos hipotecarios; Agencia de crédito; Consumidores LMI; Riesgo crediticio; Solicitud de hipoteca.

1. Introduction

In an increasingly competitive home lending industry, the need for efficient and streamlined applicant processing has never been more critical. Financial institutions face mounting pressure to enhance customer experience while adhering to stringent regulatory standards. As a response to these challenges, the establishment of an in-house applicant bureau has emerged as a strategic solution. This initiative aims to centralize the evaluation and management of loan applications, significantly reducing turnaround times and improving decision-making processes.

The benefits of an in-house applicant bureau extend beyond mere efficiency. By leveraging advanced data analytics, lenders can gain deeper insights into customer needs and behaviors, allowing for more personalized service offerings. This targeted approach not only accelerates the application process but also fosters stronger relationships with potential borrowers, ultimately leading to higher customer satisfaction and retention rates. Additionally, centralizing applicant processing helps maintain compliance by ensuring consistent application of underwriting standards and policies. Bureau model suite will be an important criterion in the application decisioning process. The score will also be used to set new risk tolerance thresholds, to improve sales volume and to better assess applicants at or below current regulatory thresholds.

However, the implementation of an in-house applicant bureau is not without its challenges. Our client will navigate potential obstacles such as integrating new technologies, training users or business owners, and managing the transition from using only FICO to using this new proposed model. Moreover, ensuring data security and maintaining regulatory compliance remain paramount concerns. This applicant-level score needs to be converted into application-level score for final application decisioning. A higher Bureau score indicates a lower risk of going bad. Therefore, applicants with higher scores will be more likely to receive loan approval, wider choice of mortgage product selection and better loan terms and pricing. The model allows for consistent scoring and decision strategies across the entire HL product spectrum.

The Bureau model will be used in two major ways: First as a complement to FICO assessing applicant risk. Second, as an input to a future Application-level score. Then, Bureau model will act as a tool for:

- Lead development - Assess individual applicant risk to determine the type of product they qualify for
- Enhance assessment of applicants below FICO thresholds
- Underwriting of mortgage loans
- Swap in of applications below the FICO thresholds to increase volume

This research aims to build an applicant-level bureau-only score based on upgraded bureau internal attributes. This score will eventually serve as the basis for evaluating a customer's credit risk before any loan structure or collateral information is considered. It will be used as a standalone score that can be employed in the initial customer evaluation to identify better marketing leads and as input to a future application-level model. We will explore in this paper the rationale behind the development of an in-house applicant bureau, the potential benefits it offers, and the challenges that may arise during its implementation in the dynamic landscape of home lending.

2. Methodology

In this project, we used Industry Archive data that was purchased by our client from one of the three major United States credit bureaus to include mortgages opened in 2018 with their corresponding 24-month performance information for individuals. It was a 7 million applicant sample representing a full credit spectrum population. The data includes credit bureau attributes about Strategic Attribute Redevelopment (STARs) at an approximate time of application, three months prior Open date, and the Automated Response Format Specifications files (ARF) which includes the complete credit bureau information for the individual at 24 months from their open date to determine their performance. The other two credit bureaus´ archive files were also purchased for model validation. Alternative data was also obtained and shared with us to evaluate non-traditional or alternative bureau information, such as rental data or public records.

This Bureau score was developed using a combination of Industry standard attribute exploration and analysis used in traditional origination models as well as incorporating Machine learning techniques to examine attribute predictive potential, distribution values, coverage, and interactions.

Our Bureau will be an estimate of a customer's credit worthiness and the likelihood a mortgage loan will go "bad" with in the first 24 months after booking. A loan is considered "bad" if the status ever changes to foreclosure or 60+ days past due (DPD). Unlike other lines of business, Home lending (HL) does not have Through the Door population available for modeling analytics, pre-approvals or "Leads" are not consider applications and therefore the data is not kept in the system as such, a home lending application is a time sensitive legal document which often requires a fee from the applicant, once the applicant decides to apply with our client it is very likely to book/fund the loan, as booked to approve percentage is between 75 and 80%, where most "Declines" happen because the applicant terminates the process or because policy and/or documentation requirements are not met not because of a customer's credit risk.

Developing a model using only booked population may result in sample selection bias, especially for potential applicants that do not meet the current strategy and regulation standards. To observe a full spectrum population, we used an industry level archive from the credit bureau, where the modeling team can evaluate a national population pool of people that opened a mortgage in the year 2018.

The Bureau model will be developed using only STAR attributes as the predictive variables. STARs are traditional bureau attributes developed in-house. We also leveraged the access to additional data time periods by using a database that contains anonymized data that can be used for model validation and backtesting.

Bureau inference method was used to define the performance of the industry archive population. The bureau ARF includes tradeline level information at 24 months for each applicant in the archive. Using the credit bureau user's guide, the modeling team defined the target as follow:

- Identified mortgage trade opened in 2018.
- Excludes Disputed tradelines and trades where applicants are identified as Deceased.
- Used the tradeline Status, Payment Profile and Account Condition codes to identify Ever 60+ DPD and

foreclosure mortgage trades to define the target bad=1 else target bad=0.

Variable reduction techniques were used to discard attributes which show little promise for inclusion in the scorecard. This consists of several modules of analysis such as univariate, bivariate, clustering, correlation and identifying feature importance through default ML models. Before the variable selection step, data treatment techniques were used to deal with missings, special values and outliers. Final model attribute list was used to train multiple Machine Learning models like XGBoosting (XGB) to identify the best model based on statistical significance, model complexity, processing time and business insight.

The key criteria for success of this model are measures of rank-ordering power, which indicate the degree to which higher scores are reliably associated with lower bad rates (and vice-versa) in cross-sectional comparisons. Several types of statistics for rank ordering are possible for consideration:

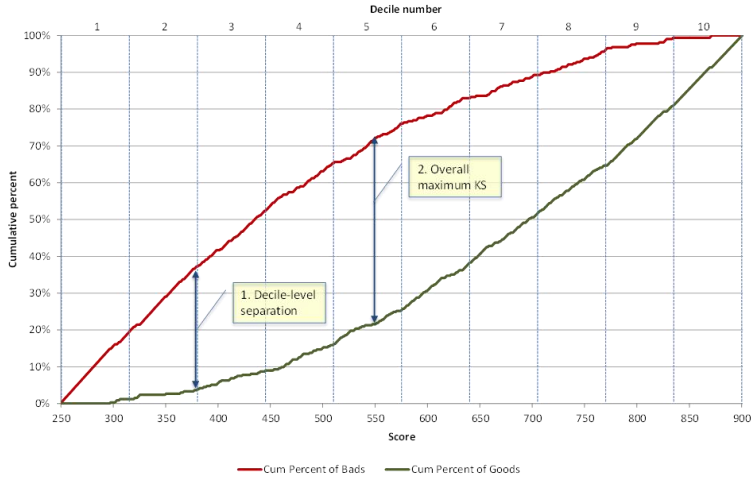
- Kolmogorov-Smirnov (KS) statistic
- Receiver Operating Characteristic (ROC) curve, area under that curve (AUC) and AUC Precision Recall (AUC-PR)
- Bad capture rate

The KS statistic is the maximum difference between the cumulative distribution of the "good" outcomes and the cumulative distribution the "bad" outcomes. This statistic measures how well the model scores can distinguish between the distribution of "good" and "bad" outcomes. Details of the KS calculation for Bureau performance testing is as follows:

- Population of interest is rank-ordered by score generated from a model, and the cumulative distribution of goods and bads by this rank-ordering is calculated (distribution is represented as frequency)
- Model separation is calculated as a difference between the cumulative distribution of goods and the cumulative distribution of bads at each individual observation
- Maximum model separation is calculated within deciles
- KS is calculated as the largest value among the decile-level maximum separations.

Below Figure 1 shows illustration of the KS calculation. In the figure, maximum model separation within deciles is denoted as "1. Decile-level separation" and KS is denoted as "2. Overall maximum KS".

Figure 1 – Illustration of KS Calculation.



Source: Authors.

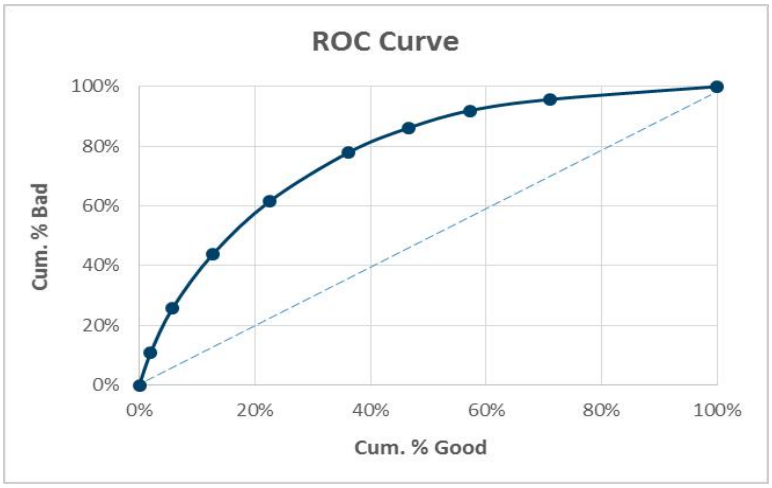
The value of the KS statistic can range from 0 to 100 after the calculation is multiplied by one hundred. Higher KS statistic values indicate that the variable being assessed shows high discriminatory power and is therefore a strong predictor of good versus bad.

The choice of the KS statistic as the primary metric for model discriminatory power is standard in the risk scoring industry. Given the importance of ensuring alignment against business intuition in the development of these models, this statistic was applied as the primary indicator of discriminatory power for ease and familiarity of interpretation by business experts.

Furthermore, the ROC curves are a graphical representation of the model's power to distinguish between goods and bads. ROC curves are constructed by scoring all loans and ordering the goods by score on the x-axis and then plotting the percentage of bads excluded at each score on the y-axis. It shows how well the model separates the goods from the bads. In statistical terms, it shows sensitivity vs. (1 – specificity).

The Area Under the Curve (AUC or AUROC) is a synthetic index calculated for ROC curves. It measures classifier performance across all score ranges and is a better measure of overall scorecard strength then the KS statistic. The AUC is the probability that the model correctly classifies an event as good or bad. A larger AUC value indicates better model performance, see Figure 2.

Figure 2 – Illustration of the ROC curve and AUC calculation.



Source: Authors.

An AUC value of 0.5 indicates the model is as good as a random mechanism to classify the observations as "good" or "bad," whereas AUC values near 1 indicate that the model is correctly classifying the data. In the illustration, the AUC is 0.65. See following Table 1 for further details:

Table 1 – AUC Discrimination Thresholds.

AUC	Interpretation
AUC = 0.5	No discrimination
$0.7 \leq AUC < 0.8$	Acceptable discrimination
$0.8 \leq AUC < 0.9$	Excellent discrimination
$AUC \geq 0.9$	Outstanding discrimination, that is unlikely to occur in reality

Source: Authors.

Like KS, the AUC is also useful for model selection. Its advantage is that the AUC compares the similarity of the entire distributions rather than just one point as in the KS statistic. A disadvantage of the AUC metric is that models with reasonably correct values of the AUC may not perform well in terms of discriminating outcomes if an inappropriate value of the threshold score is selected by the model user.

Another metric, the Area Under the Curve Precision Recall (AUC-PR) curve, is a common way to summarize a model's overall performance. In a perfect classifier, PR AUC =1 because your model always correctly predicts the positive and negative classes. Since precision-recall curves do not consider true negatives, AUC-PR is commonly used for heavily imbalanced datasets where you are optimizing for the positive class.

AUC-PR needs two elements to calculate the performance metric: predicted values and actual performance. The process assigns binary labels to denote positive and negative classes. Actual performance is determined by the observed populations while predicted values are the output of the model. Table 2 describes how the elements are calculated.

Table 2 – AUC – PR calculation.

Axis	Metric	Statistical Jargon	Description	Denominator
X – axis	Recall / true positive rate (TPR)	TP / (TP + FN)	# True positives / # actual positives	Number of data points with positive actual labels
Y – axis	Precision / positive predictive value (PPV)	TP / (TP + FP)	# True positives / # predicted positive	Number of data points with positive prediction labels

Source: Authors.

Thresholds are optimized for the prediction score to generate prediction labels:

- Data points with prediction scores above the cutoff are given positive prediction labels.
- Data points with prediction scores below the cutoff are given negative prediction labels.

The PR curve is created by varying the threshold for predicting a positive or negative outcome and plotting the precision against the recall for each threshold.

The AUC-PR is the area under the PR curve and represents the overall performance of the model. A perfect model would have an AUC-PR of 1, while a random model would have an AUC-PR equal to the ratio of positive samples in the dataset. Like the AUC, the AUC-PR provides a single value that summarizes the model's overall performance and is particularly useful when comparing the performance of multiple models. In the figure above, the grey dotted line represents a "baseline" classifier — this classifier would simply predict that all instances belong to the positive class. The purple line represents an ideal classifier with perfect precision and recall at all thresholds.

The PR curve and AUC-PR provide a more accurate assessment of the model's performance than metrics such as accuracy or F1 score, which may be biased towards the majority class. In addition, they can provide insight into the trade-off between precision and recall and help to identify the optimal threshold for making predictions.

In addition to discriminatory power between "bad" and "good" offered by the KS statistic, it is further critical to evaluate performance, especially at the lower end of scores representing the riskiest customers. Commonly used in the lending industry, "bad capture rate" is defined as the percentage of bads captured in the top decile (the worst 10%). Use of these metrics allows an evaluation of how well the model performs in the lowest portion of the customer portfolio and thus provides additional criteria for success.

The Bureau model will be used in conjunction to FICO score in application decisioning, however, direct comparison

between the two scores cannot be documented, therefore Vantage score is being used as a benchmark. The VantageScore 3.0 and Vantage Score 4.0 models were provided as part of the data inputs. The VantageScore 3.0 development uses credit bureau data blended from two different timeframes, 2009–2011 and 2010–2012, to capture a broad development sample of recent consumer behaviors, including activity at the height of, and following, the economic crisis and used the target variable of 90+ DPD within 24 months. Like FICO, the score ranges from 300 to 850, indicating that higher scores indicate a lower level of consumer risk. This benchmarking process was intended to prove that our Bureau model can provide performance improvement over vendor models. VantageScore 4.0 was also obtained from our client's data sandbox, for backtesting and target validation for the 2007 data sample that will be used to validate the Bureau model in a stressed period. VantageScore 4.0 is the most updated model from Vantage, it uses 2014 to 2016 timeframe for development, like Vantage 3, has a score range from 300 to 850 and uses the 90+ DPD within 24 months as a target variable. Vantage 4 was developed using machine learning techniques and incorporates trended attributes as well as traditional credit information.

Our Bureau model will use the "SHAP" value base approach to determine the reason codes needed in Adverse Action Letters. SHAP is a well stablished technique for XGBoost models.

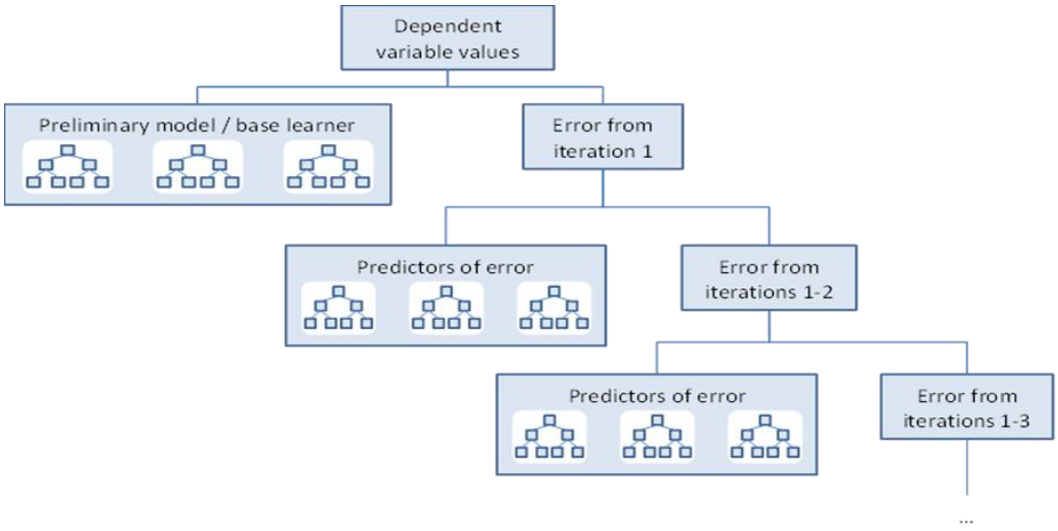
For Bureau development, the modeling team has considered using both traditional statistical model (logistic) and machine learning techniques (XGB).

Historically, the industry has been using logistic regression models to predict the probability of default as this approach is well-known and well-documented, also model results can easily be compared to business expectations with the ability to quantify risk drivers. Another advantage of logistic regression is that infrastructure to implement the model in production is less complicated and has relatively low computational requirements.

In recent years, however, some models in the industry started using machine learning techniques as computational capabilities have improved and many implementation challenges have been resolved. Gradient Boosting Model (GBM) models iteratively build a scoring model (shallow trees in this instance) by repeatedly adjusting a preliminary equation and optimizing a differentiable loss function. A common loss function is a multiple of the sum of squared errors calculated across all observations. For observations that have a binary outcome, for example good/bad, fraud/not fraud, it is more common to build a loss function from the log-likelihood of a logistic regression with a logit link function: L(y,f(x))=y logf_0(1+e^(-f(x)))+(1-y) logf_0(1+e^(f(x))). GBM begins with an initial prediction as constant value such as average target. After doing this, the machine then takes the errors generated by these predictions and includes new input variables to attempt to predict that error. This process of fitting to errors can be repeated numerous times. There could be hundreds of iterations depending on the model developer's specifications. While performing this iterative process, observations that were misclassified in the model's latest iteration (e.g. they were overlooked frauds or false positives) may receive higher weight than observations that were classified correctly.

One way to visualize the process is through an ensemble of shallow trees as shown below in Figure 3.

Figure 3 – Visualization of a GBM modeling process.



Source: Authors.

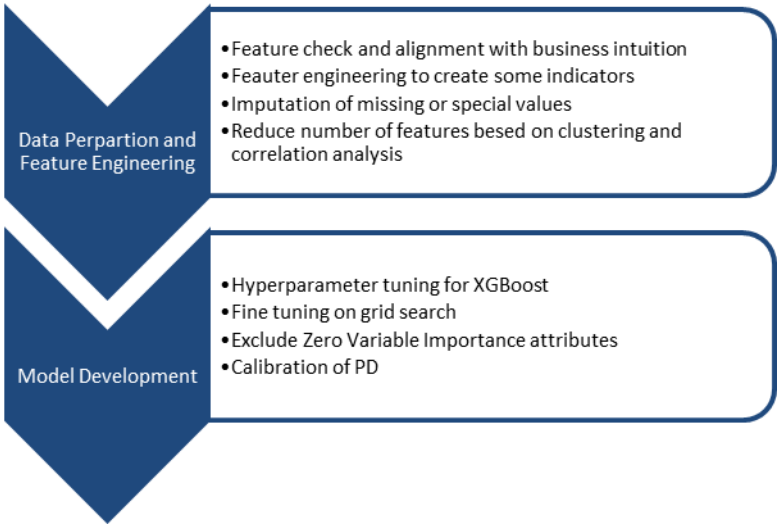
For the first branching, one node is the set of scores produced by the preliminary model and the other is the error from that preliminary attempt. The preliminary model comprises a tree ensemble, which means that it is a set of multi-level binary decision trees with variables that appear in multiple trees. The terminal nodes of each tree correspond to a score, and the scores across different trees add together to generate an estimated value. The algorithm creates and manipulates trees until it has minimized errors. After the shallow tree's preliminary model node is set, the second node (the one for errors) then branches into two: one of the resulting nodes repeats the tree ensemble procedure to model the error, and the other node captures the error from the neighboring node's attempt to do so.

This branching between predicted errors and remaining errors repeats for each iteration. After each attempt to model the error, the terms created in that attempt are added to the terms from all previous iterations. The new terms are adjusted by scalar multipliers to minimize the sum of squared errors calculated using both the new and the preexisting terms. The "gradient boosting" part of the process's name comes from this repeated modeling of the error remaining after each iteration: it is possible to see the process as minimizing the model's error (or loss) function, and it does this by using each iteration's negative gradient to follow the steepest path to a minimized squared error.

From a regulatory standpoint, machine learning methods are scrutinized more because of their lack of transparency compared to simpler statistical models. Significant effort has been invested to address the interpretability concern associated with machine learning models. The use of SHAP values addresses this concern.

The overall approach selected by our team was to use the extreme gradient boosting (XGBoost) algorithm. XGBoost is a more regularized form of GBM. XGBoost uses advanced regularization (L1 & L2), which improves model generalization capabilities. The decision to choose XGBoost was based on model performance. In general, machine learning models require weaker assumptions and are more granular than logistic regression models and are therefore more powerful. It was clearly observed that the XGBoost model can provide much better risk separation compared to logistic regression. All models were built on the training dataset and evaluated on both in-time in-sample (holdout) dataset and out-of-time validation dataset. Next Figure shows the overall XGBOOST model development process.

Figure 4 – ML Model Development Process.



Source: Authors.

XGBoost has the following advantages over the logistic model:

- Improved prediction accuracy: XGBoost's sequential construction and complexity allows it to identify more subtle patterns from a larger set of explanatory variables than a classifier such as logistic regression. The improvement in accuracy is more pronounced in situations where:
- the pattern to be identified is nonlinear and non-monotonic.
- there are interaction effects among the independent variables.
- there are many initial candidate variables, and a large amount of input data; and
- the pattern to be identified is dynamic (changes over time)
- Less restrictive assumptions: Because XGBoost is a non-parametric method, its pattern recognition is not limited by a functional form. Furthermore, the algorithm is also capable of processing highly correlated candidate variables without adverse impact on estimation.
- Less manual effort: XGBoost can be programmed to remove manual steps from development and implementation. Input data require only minimal cleaning. The model developers just need to ensure that inputs are appropriate, then XGBoost method can select and transform variables automatically, and thus save time and reduce the risk of mistakes from a manual process of variable transformation and selection.
- Less overfitting issue: One advantage of XGBoost over other machine learning models such as Stochastic Gradient Boosting is that the regularization to hinder overfitting is built into the method's standard operation. Thus, the modeling team does not have to experiment with the form of the complexity penalty.

XGBoost is a commonly used machine learning method, and it frequently wins machine learning competitions (Chen and Guestrin 2016). The advantages of XGBoost over other machine learning methods are accuracy as well as speed, and:

- Scalability: XGBoost can be run on parallel and distributed computing platforms.
- Memory use: XGBoost uses an out-of-core computation method that uses disk space memory as well as processor memory.
- Sorting: XGBoost sorts the data once, and then stores the data into blocks of in-memory units rather than re-sorting the data repeatedly.

- Categorical variables: The XGBoost method for encoding categorical variables and implementing them into the procedure is to convert the categorical variable into a vector that can be sorted rather than considering the outcomes from all of the categorical values. This encoding method results in a sparse data set, but XGBoost is designed to efficiently handle sparse data.
- Missing values: The XGBoost method handles missing values without imputation or proxies; and
- Loss function: The XGBoost method works by minimizing a computationally more convenient approximate loss function rather than the actual loss function.

However, XGBoost also has drawbacks. XGBoost model that performs too many iterations may overfit the data. In practice, this concern is alleviated by tuning the hyperparameters in the algorithm to control the complexity of the model (e.g., number of trees, maximum depth, learning rate, data and feature sampling rate, and regularization weight), and using cross-validation to evaluate the actual performance of the model. A greater concern is that an XGBoost is difficult to interpret on a case-by-case basis. The machine performing XGBoost can transform and manipulate the data, introduce interaction terms that incorporate multiple inputs, assign unequal weights to observations when building the model, and aggregate terms from different iterations. The process is therefore a "black box" as model developers cannot reverse engineer a final model's form. An XGBoost model's complexity also limits how easily a particular output can be explained to a person who is not intimately familiar with the model. For example, if a logistic regression with independent variables A, B, C, and D assigns a high estimated probability to a card, then the reason for the high estimate must be one of those four variables. All four of those variables probably have an intuitive relationship with the probability estimate, so it is easy to link cause and effect. An XGBoost model, on the other hand, is more difficult to parse for an individual case as it includes hundreds of variables and interaction terms. A card might therefore have a high default probability because of a combination of numerous factors that are not necessarily intuitive on an individual basis.

As alternative modeling approach, the modeling team considered traditional segmented Logistic regression. As discussed above, the modeling team decided to select XGBoost using random grid search to tune the hyperparameter as the champion model. The descriptions of the alternative modeling approach as well as analytical comparisons with XGBoost are presented in this section.

The modeling team a developed a segmented logistic regression model for the alternative approach. The key objective of segmentation is to differentiate responses to risk drivers and to define a set of subpopulations that, when modeled individually and then combined altogether, rank risk more effectively than a single model built on the overall population. In an effective segmentation scheme, the responses to key risk drivers should be not only homogenous within each segment but also meaningfully differentiated across segments. For instance, segments may utilize different predictors, or place significantly different weights on shared predictors. A series of segmented regression models will outperform a single model if the predictors differ for the populations of each segment or if populations within segments respond differently to the same variables. Following Table 3 illustrates the segmentation scheme of alternative Bureau logistic regression model.

Table 3 – Logistic Model Segmentation Schema.

Segment Number	Description	Total	%Pop	Bads	Bad Rate
1	**Severe Delinquent**	581,525	11%	44,003	7.57%
2	**Clean, Recent Credit**	957,982	18%	37,201	3.88%
3	**Clean, No Recent Credit, Low Utilization**	3,183,085	61%	19,243	0.60%
4	**Clean, No Recent Credit, High Utilization**	464,394	9%	10,760	2.32%
	Total	**5,186,986**	**100%**	**111,207**	**2.14%**

Source: Authors.

All the segments have enough observations and bad volume to develop a robust logistic regression model. The bad rate is distinctly different by segments across all samples indicating sufficient separation between the segments and enough homogeneity within the segment.

The next step is the development of a logistic regression model for each segment. As noted previously, the overall objective of the model(s) is to have strong rank ordering in terms of default risk. A series of steps for variable selection is carried out to ensure that variables with the strongest predictive power of default are included in the final model. Starting from an initial list of 586 STAR Attributes, the modeling team developed 4 logistic regression equations in 4 segments. Additional details for each type of data as well as the number of attributes within each subtype are summarized below.

Table 4 - Development Attribute Types.

Type	#	Sub-Type	#
Trade Lines	**560**	Age	51
		Amount	2
		Authorized User Trades	8
		Balance	104
		BK_CO_Derog	38
		Collections	4
		Decomissioned	14
		Deferred Status	9
		Delinquency	33
		Limit	36
		Months_Since	45
		No_of_Tradelines	129
		Open to buy	2
		Other	26
		Payment	12
		Unknown	1
		Utilization	35
		Velocity	6
		Worst Status	5
# of Inquiries	**11**	# of Inquiries	10
		Months_Since Inquiries	1
Public Records	**9**	# of Public Records	4
		Amount of Public Records	1
		BK_CO_Derog	1
		Months_Since Public Records	3

Source: Authors.

For different product type trade lines, combinations of different dimensions of information are used to derive attributes related to trade lines. Those dimensions include: the number of trade lines, trade line growth rate, age of trade lines, balances, limit, utilization, payment status, time since delinquency, frequency of delinquency, delinquency status and others. These dimensions comprehensively capture past consumer behavior on all products, which is ultimately used to infer future behavior. Examples of STAR attributes for trade lines are provided below.

Table 5 - STAR Trade Line Examples (revolving trades).

Variable Name	Description
S00017_T_REV_AGE_OLD	S00017 Age of Oldest Revolving Trade
S00021_T_REV_NUM	S00021 Number of Revolving Trade Lines
S00027_T_REV_BAL	S00027 Total Revolving Balance
S00029_T_REV_BAL_OPND_LT12M	S00029 Total Bal Revolving Trades Opened LT12M
S00030_T_REV_LMT	S00030 Total Revolving Limit
S00032_T_REV_UTILIZATION	S00032 Revolving Utilization
S00806_T_REV_NUM_UTZ_GE75	S00806 Number of Revolving with Utilization GE 75%
S00807_T_REV_NUM_UTZ_GE90	S00807 Number of Revolving with Utilization GE 90%
S00815_T_REV_PCT_OPNLT6M	S00815 Percentage of Revolving Open LT 6 Months
S00820_T_REV_PCT_SAT	S00820 Percentage of Satisfactory Revolving TLs

Source: Authors.

Further, STAR attributes include a comprehensive set of special values to distinguish different reasons why a particular attribute may not be available. Below is the list of special values along with their definitions:

- 990 = no data items for this attribute
- 991 = no category specific data items
- 992 = no credit record requested
- 993 = no hit at the credit bureau
- 994 = no data items outside of exclusion criteria (Disputed, Deceased, lost/Stolen, Transferred, Refinanced/Terminated)
- 995 = attribute decommission
- 996 = no recently updated data items
- 997 = ratio calculation has zero denominator / Cannot compute age due to missing date
- 998 = no delinquency reported to measure time since delinquency
- 999 = currently not assigned

There are no limitations or boundaries for inputs outside of which the model will not work properly. Extrapolation risks are not a concern since extreme input values are capped.

3. Results and Discussion

As the first set of results, Table 6 shows the performance comparison between the logistic model and a preliminary ML model for the development and out-of-time (OOT) samples:

Table 6 - Logistic vs ML Model Comparison.

	KS			AUC		
	ML	Logistic	Difference	ML	Logistic	Difference
Development	58.46	57.58	0.87	86.50	85.85	0.66
OOT1	57.97	56.83	1.14	86.13	85.60	0.53
OOT2	58.12	57.24	0.88	86.38	85.79	0.59

Source: Authors.

ML shows a slight advantage over logistic for the overall development population and OOT periods. In addition, to better represent the potential use of the model in production, performance was tested for the population that currently doesn't meet approval standards, FICO <=620. Using this population allows the comparison to focus in the area where a custom model has the most impact. The sub population shows a clear distinction between the ML and Logistic models with an ML improvement of 1.8% in KS and 1.4% in AUC for the development population, with higher improvements seen in the OOT periods. Table 7 indicates the results for the subset population.

Table 7 - Logistic vs ML Individual Loans FICO <= 620.

	KS			AUC		
	ML	Logistic	Difference	ML	Logistic	Difference
Development	27.63	25.75	1.83	69.36	67.92	1.44
OOT1	29.55	26.80	2.79	70.22	68.53	1.69
OOT2	28.42	26.59	1.92	69.53	68.12	1.41

Source: Authors.

As with any model that influences credit decisions to a business or individual, the Bureau machine learning model presented in this document adheres to external regulatory requirements. It complies with the Equal Credit Opportunity Act of 1975 that prohibits a credit lending decision from discriminating against certain protected classes. The Federal Reserve released Regulation B ("Reg. B") to outline rules for compliance with the Equal Credit Opportunity Act.

Throughout the course of model development, any variable that could be discriminatory has been excluded or removed prior to any finalization of models. To further ensure that no unintentional discrimination is introduced from any application or usage of this model, additional disparate impact analysis will be conducted to demonstrate that disproportionate "adverse impact" on persons in a protected class is limited.

Where applicable, up to four adverse action codes are generated in alignment with the Equal Credit Opportunity Act as implemented by Regulation B. The Bureau model adverse action codes and verbiages will be reviewed and approved by Legal, Compliance and Fair Lending. For HL, these codes/verbiage may be mapped to different verbiage in the downstream letter generation process, which is owned by the Business. The Bureau model directly takes the STAR attributes as inputs and as such do not have any feeder models.

Regarding the development of this model, we obatined the following numbers in Table 8, which indicates the development Sample population, as well as the two Out-of-Time (OOT).

Table 8 - Development Sample Population.

2018 Development Data		Volume	Volume%	Bad Volume	Bad Rate%
Q1	OOT1	1,126,760	16%	22,785	2.02%
Q2 & Q3	Train	1,989,389	29%	43,240	2.17%
	Test	852,596	12%	18,289	2.15%
Q4	OOT2	1,218,241	18%	26,893	2.21%
Overall		5,186,986	75%	111,207	2.14%

Source: Authors.

It was found that applicants may have multiple mortgage tradelines that meet the modeling criteria, however only one tradeline is needed for analysis. A deduping logic was created to obtain only one mortgage tradeline that meets the criteria. The following logic was applied:

- If an applicant has multiple Mortgage trades opened in 2018 with months on books between 23 and 25 then prioritize the trade that
  - Has target bad =1
  - Is a client's tradeline
  - Balance amount is not blank
- If the tradelines are complete duplicates, then just apply deduping by Applicant key.

Table 9 indicates the tradeline deduping waterfall results:

Table 9 - Deduping Waterfall results.

Filter Name	Total Count	Unique Key Count
Total number of records	143,655,753	6,993,465
Total number of tradelines after considering non-disputed or non-deceased mortgages that opened in 2018	8,784,206	6,946,625
Total number of tradelines after applying months on books filter	8,562,799	6,946,625
Total number of tradelines after applying deduplication logic	6,945,403	6,945,403

Source: Authors.

Two main factors were considered when selecting 60+ DPD or foreclosure as model target:

- Business and strategy teams use 60+ DPD as the most common metric for performance reporting.
- Delinquency is very low, 2.14%, using 90+ DPD (1.3%) or 120+DPD (.94%) significantly reduces the bad volume available for model training. Table 10 shows the volume for each target in the development dataset.

Table 10 - Development Data Volume.

Development Period	Volume	60+ Bad Volume	60+ Bad Rate%	90+ Bad Volume	90+ Bad Rate%	120+ Bad Volume	120+ Bad Rate%
Train	1,989,389	43,240	2.17%	26,174	1.32%	18,888	0.95%
Test	852,596	18,289	2.15%	11,036	1.29%	7,865	0.92%
OOT1	1,126,760	22,785	2.02%	13,866	1.23%	10,111	0.90%
OOT2	1,218,241	26,893	2.21%	16,208	1.33%	11,705	0.96%
Overall	**5,186,986**	**111,207**	**2.14%**	**67,284**	**1.30%**	**48,569**	**0.94%**

Source: Authors.

The model development data was used to conduct analysis comparing different target options. First analysis included developing a preliminary model using the 60+ DPD target and evaluating the results using the 90+ DPD and 120+ DPD options. Table 11 indicates that performance is comparable between 3 targets, choosing 60+ DPD allows for greater "bad" volume.

Table 11 - Target Definition Performance.

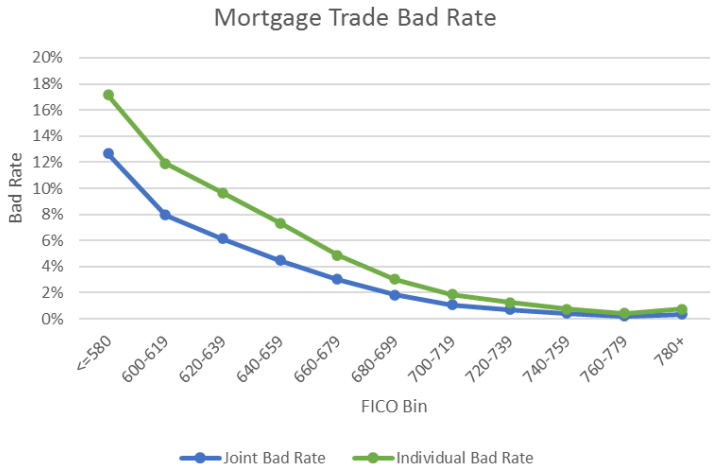
Development Period	KS			AUC		
	60+	90+	120+	60+	90+	120+
Train	57.59	58.30	58.19	85.57	85.75	85.67
Test	57.69	58.59	58.59	85.66	85.92	85.93
OOT1	56.99	58.19	58.19	85.28	85.74	85.64
OOT2	57.30	57.99	57.90	85.48	85.59	85.53

Source: Authors.

Mortgage loans have a longer term (15 to 30 years) than other finance products like Auto and Credit Cards. It is well documented that Auto loans have a life cycle between 18 and 24 months and it is a commonly used period in scorecard modeling. HL business and strategy teams observe most of the delinquency between the 4th and 5th year. The 24 months performance window was selected following FICO score development definition and other LOBs common practices, this assumption was tested using the 2007 data.

Credit bureau archive includes information at the applicant level; however, the type of mortgage tradelines include individual and joint loans. For Joint loans two applicants share the same trade performance while their credit information is different. Figure 5 plots the bad rate by FICO score bin for individual and Joint populations, for Joint loans both applicants are included.

Figure 5 – Bad Rate by Application Type.

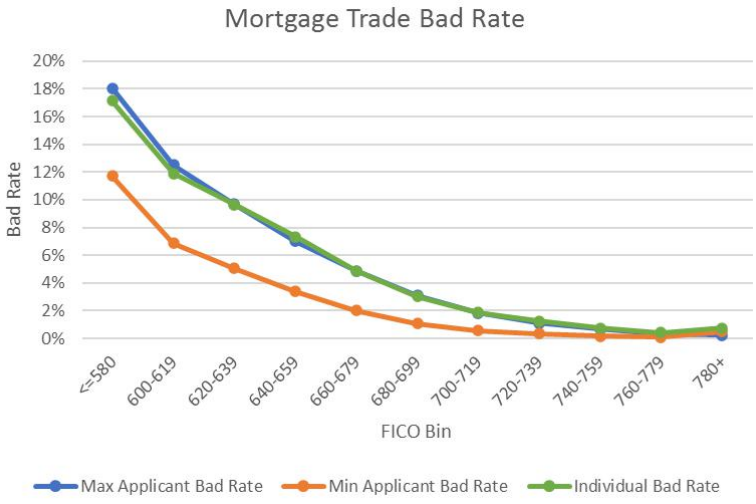


Source: Authors.

Figure 5 indicates an apparent performance difference between the two types of loans, with Joint showing lower bad rates than individual. This performance difference is associated with the advantage of having two people responsible for the loan, primarily due to higher income, as it is combined. The performance difference introduces a bias to the development of an applicant level model. Other LOBs, particularly Auto, have dealt with this issue in their model development but excluding all joint loans and using the individual population only, however, joint loans account for 50% of the applicant population in the HL archive, excluding joint population would significantly reduce the development volume and bad performance rate in an already limited period (2018).

To further analyze the data, we separated the applicants for joint loans as Min and Max applicant, designating the applicant with the lowest FICO score as "Min" and highest FICO score as "Max." Figure 6 shows the bad rate distribution for Min and Max applicant plotted against Individual loans.

Figure 6 – Bad Rate by Min and Max Applicants.



Source: Authors.

Figure 6 illustrates that when separating the applicants, the Min applicant still shows significantly better performance than Individual and Max applicants, for the same score bin, the modeling team concluded that the Min applicant gets the most advantage from having a coapplicant while the Max applicant, having an overlapping distribution with Individual shows that it is a better representation of the loan risk at each score level. Based on these results the development data will include Individual loan applicants and the Max applicant from joint applications. Table 12 shows the final distribution of the development data by application type.

Table 12 - Final Application Type Distribution.

	Count	Percentage
Total number of joint loans	1,667,212	32.06%
Total number of individual loans	3,533,317	67.94%
Total Count	**5,200,529**	100.00%

Source: Authors.

Note that keeping one applicant from joint tradelines would give a total proportion of 75% loans classified as individual and 25% classified as Joint. After data clean up including keeping only applicants with valid credit information the final distribution of Individual vs Joint is 68% and 32%.

Final development dataset excludes applicants without a valid Credit score for a total development sample of 5.18 million records, with a 2.14% bad rate.

As described before, the Bureau model was developed using the XGBoost methodology. The model fitting process starts with the pre-processed datasets, after exclusion of unintuitive variables, and treatment of special values and missing values. The most important step in XGboost model is to tune the hyperparameter and create a robust, not over fit, and parsimonious ML Model.

In order to do the local explanation and generate adverse reason code, we force monotonicity in the XGBoost model by inputting the feature trend. The feature trend represents the direction of the relationship of the feature and risk of the applicant by business intuition.

We developed a default model to use as comparison against Random Search parameter tunning. Random Grid Research was favored over Grid Search due to the propensity of the Grid Search to grow exponentially, with as few as 4 parameters which becomes impractical and resource intensive. To optimize with random search, the function evaluates a set number of random configurations based in the parameter space and improves the chances of finding the optimal parameter.

The algorithm picks 10 random combinations, and the 5-fold Cross Validation is performed on those 10 models with different parameter sets. From the CV results, the top ten models are chosen based on the mean AUC-PR score and compared their parameters to pick the best model. Table 13 shows the grid search results. Results indicate that model 1 is the optimum model however modelers selected model 4, as there is minimum loss in performance and complexity is reduced.

Table 13 - Top 10 RGS Models.

Model #	subsample	Reg lambda	Reg alpha	N estimators	Min child weight	Max depth	Learning rate	gamma	Colsample bytree	Mean test score	Mean fit time
1	0.7	10	20	700	10	6	0.1	0	0.5	0.151979	2782
2	0.7	10	20	500	5	6	0.1	0	0.5	0.151905	2470
3	0.7	10	20	500	10	3	0.05	0	1	0.15086	2817
4	1	10	20	300	5	4	0.1	0	1	0.150841	1999
5	0.5	1	0	300	10	5	0.2	0	1	0.150761	2127
6	0.5	1	10	100	1	4	0.2	5	1	0.149799	914
7	1	0	20	100	5	4	0.1	0	0.5	0.149241	472
8	1	1	20	300	1	3	0.2	10	1	0.14852	1763
9	1	1	0	100	5	3	0.1	5	1	0.148138	620
10	0.5	1	1	100	5	4	0.01	10	0.5	0.125758	536

Source: Authors.

The final model selected includes 48 features, Table 14 indicates the hyperparameters used in the final model.

Table 14 - Final Model Hyperparameters.

Number of Estimators	Learning Rate	Max Depth	Reg Lambda	Alpha	Gamma	Subsample	Col Sample by Tree	Min Child Weight	Monotonicity
300	0.1	4	10	20	0	1	1	5	yes

Source: Authors.

Table 15 Shows the top 10 model attributes selected used in the model.

Table 15 - Top 10 Model Attributes.

S.No.	Description	ML Feature Importance
1	S00830 Percentage of TLs and PR Ever 30P or Derog	0.2577
2	S00608 Percentage of Trade Lines Open LT 24 Months	0.0933
3	S00745 Number of Non-ILs with Utilization GE 75%	0.0666
4	S00162 Age of Oldest Credit Card Updated 6M	0.0575
5	S00838 Num Non-Bank IQs past 24M deduped EX 7D	0.0549
6	S00624 Worst Status of a Trade in the Last 12 Mos - DEROG	0.0474
7	S00792 Percentage of Satisfactory Real Estate TLs	0.0438
8	S00819 Percentage of Rev TLs with balance GT 0	0.0376
9	S00026 Number of Revolving Trades Opened GE24M	0.0370
10	S00655 Number of Open Bankcards Open GE 24 M	0.0354

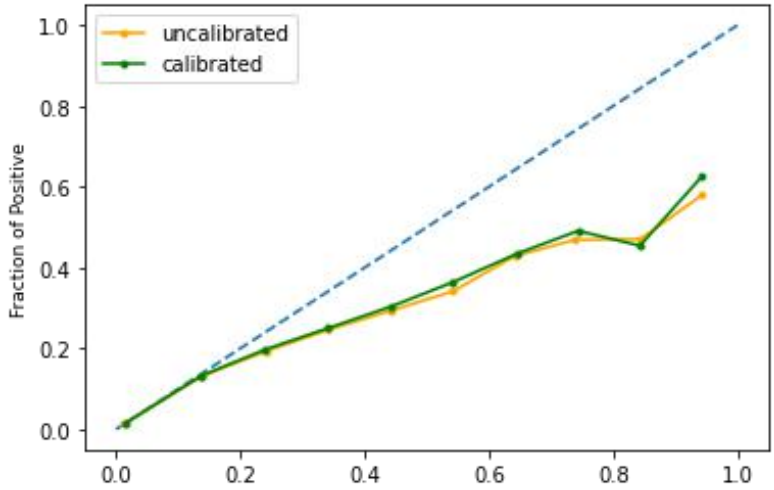
Source: Authors.

Machine learning models do not typically produce calibrated probabilities. In other words, the score produced by a model does not necessarily represent the percentage of data points belonging to one class rather than the other. Uncalibrated probabilities may be over-confident in some cases and under-confident in other cases, especially when the data is imbalanced

(Brownlee, 2020). Predicted PD won't be able to be distributed well on two ends due to the imbalance data (few number of bads in HL data). In addition, in boosted decision trees, the predicted probabilities are away from 0 and 1 when trained if not calibrated (Niculescu-Mizil et al., 2005). We used sigmoid regressor to calibrate the output.

By the calibration on prediction, a monotonic transformation is performed so that the ranking order will remain the same after calibration and all the ranking order performance metrics will not be changed while the actual PD vs. predicted PD will change on different groups of population. The Logistic Regression default function from the python's Sklearn library is used to calculate the predicted values. The function has the option to output the predicted probability, this value is then compared to the uncalibrated predicted probability of the XGB model, shown in Figure 7.

Figure 7 – Uncalibrated vs Calibrated Probabilities.



Source: Authors.

The recalibrated PD is more aligned with actual bad rate than the uncalibrated PD for the worst 30% of the population.

The predicted probability output created by the Bureau model will be transformed into a score that aligns with FICO, to make model output more intuitive and easier to interpret for model users. Development data was used to analyze and approximate the PDO and odds needed to scale to FICO V3 which is the version used in HL underwriting decisioning. For the model proposed, a score of 680 will correspond to a 27:1 good/bad ratio, with every 30 points doubling the odds. This is achieved by dividing the natural log scale score by the natural log of 2 and multiplying the result by the number of points that represents a doubling of the odds. In general, the relationship between odds and scores can be presented as a linear transformation:

$$odds=p/(1-p)$$

$$Score=Offset+Factor*\ln(odds).$$

If the score is being developed using specified odds at the reference score and specified 'points to double the odds' (PDO), the factor and offset can be easily calculated by using the following equations:

$$Factor=PDO/\ln(2);$$

$$Offset=Reference\ Score-Factor*\ln(odds)$$

For Bureau model, which is designed to be an approximation of FICO, the points to double the odds are 30 and the

reference score is 680 at which the good to bad odds is 27:1

$$Factor=30/\ln(2)$$

$$Offset=680-30/\ln(2)*\ln(27)$$

The score at any given odds can be calculated as

$$Score=round(680-30/(\ln(2))*\ln(27)+30/(\ln(2))*\ln(odds))$$

After calculating it with the formula above, the PDO-aligned score is subsequently rounded and capped at the pre-defined minimum (300) and the pre-defined maximum (850).Note that the choice of scaling does not affect the predictive strength of the model. It is an operational decision to facilitate ease of understanding and consistency with existing scores.

Customers who are declined for credit are provided a disclosure referred to as an Adverse Action (AA) notice. This notice contains reasons for the decline of the credit action (called adverse action reasons), which are required by law. Up to four adverse action reason codes are outputted for Bureau model.

In order to incorporate local explainability and generate adverse action reason codes, the monotonicity constraint is enforced in the XGBoost model by inputting the pre-determined trend for each feature. For Bureau model, adverse action reason codes are generated using the GroupSHAP package with positive switch turned on and with the preset reference point for each attribute. Reference point for each attribute is chosen in a way to correspond to the bad rate of the overall Training sample, which ensures that all features in the model are compared at a common ground, therefore, have a fair chance to show up as an adverse action reason.

4. Conclusion

The research as well as the work implemented in this paper will be crucial in helping our client have a fundamental understanding regarding how an ML Bureau model can be leveraged and perform better than FICO score during mortgage applications. The establishment of an in-house bureau model for mortgage originations represents a transformative opportunity for financial institutions navigating the complexities of the home lending landscape. By centralizing the applicant evaluation process, lenders can significantly enhance operational efficiency, reduce turnaround times, and improve customer satisfaction level. The integration of advanced data analytics and ML techniques further empowers institutions to tailor their services to meet the unique needs of borrowers, fostering stronger relationships and promoting long-term loyalty.

The Bureau model is a completely new model introduced to the HL environment in our client, the score is an effort to modernize and improve underwriting decision strategies and expand the current risk profiles. The use of machine learning techniques and the implementation of STAR credit bureau attributes in model development represent an enhancement over using only traditional FICO score. In addition to that, more research can be done to continue evaluating new third party data such as LexisNexis and Rent Bureau and determine if new information can add further value to the model.

We suggest for future research that more regression techniques should be tried in the benchmark model, like Random Forest, Support Vector Machine or Naïve Bayes, as they may provide even better results and then having a better competitor for the XGBoost model, always taking into consideration the fine tuning and choosing the right hyperparameters if any. Also, it would be great if more historical data could be included in the development set in order to get more accurate results.

Acknowledgments

The authors thank our “Alma Mater” Purdue University for the guidance during this research. We would also love to

acknowledge our parents Fernando Camilo San Martin Berrocal, Celestina Marina Galindo Quintanilla, Cesar Izquierdo Vargas, and Delicia Beatriz Muñoz Llanos, for their encouragement, sacrifice and eternal support. We will always want them to be happy and we strive to give them back everything they did for us.

Conflict of Interest

The authors inform that there is not any conflict of interest during the elaboration and publishing of this research.

References

Bao, W.; Lianju, N.; & Yue, K. (2019) Integration of unsupervised and supervised machine learning algorithms for credit risk assessment. *Expert Syst. Appl. 128*, 301–315.

Barroso J. B. R. B., Silva T. C., & Souza S. R. S. d. (2018), Identifying systemic risk drivers in financial networks, *Physica A: Statistical Mechanics and its Applications*. 503, 650–674, https://doi.org/10.1016/j.physa.2018.02.144, 2-s2.0-85043784193.

Brownlee, J. (2020, Feb) *How to calibrate probabilities for imbalanced classification*. https://machinelearningmastery.com/probability-calibration-for-imbalanced-classification

Chaudhuri, T., & Yulei, F. (2020). Machine Learning Applications in Real Estate: Methods and Challenges. Journal of Real Estate Finance and Economics, 61(2), 192-210. https://doi.org/10.1007/s11146-019-09732-8

Chen, T., & Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 785–794). New York, NY, USA: ACM. https://doi.org/10.1145/2939672.2939785.

Cox, D. R. (1958). The Regression Analysis of Binary Sequences. *Journal of the Royal Statistical Society. Series B (Methodological), 20(2)*, 215–242. http://www.jstor.org/stable/2983890

Deepchecks Community Blog (2023). Understanding F1 Score, Accuracy, ROC-AUC, and PR-AUC Metrics for Models

Hodges, H., Garrity, C., & Pope, J. (2024). Deep Learning, Feature Selection, and Model Bias with Home Mortgage Loan Classification. In M. Castrillon-Santana, M. De Marsico, & A. Fred (Eds.), *Proceedings of the 13th International Conference on Pattern Recognition Applications and Methods* (Vol. 1, pp. 248-255). (International Conference on Pattern Recognition Applications and Methods; Vol. 1). SciTePress. https://doi.org/10.5220/0012326800003654

Khemakhem, S.; & Boujelbene, Y. (2017) Artificial Intelligence for Credit Risk Assessment: Artificial Neural Network and Support Vector Machines. *ACRN Oxf. J. Financ. Risk Perspect.6*, 1–17.

Krasovytskyi, D., & Stavytskyy, A. (2024). Predicting Mortgage Loan Defaults Using Machine Learning Techniques. *Ekonomika*, *103*(2), 140–160. https://doi.org/10.15388/Ekon.2024.103.2.8

Lundberg, S. M., & Lee, S.-I. (2017). *A Unified Approach to Interpreting Model Predictions*. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan & R. Garnett (ed.), Advances in Neural Information Processing Systems 30 (pp. 4765--4774). Curran Associates, Inc.

Lundberg, S.M., Erion, G.G., & Lee, S. (2018). Consistent Individualized Feature Attribution for Tree Ensembles. *ArXiv*, *abs/1802.03888*.

Mili M., Sahut J. M., & Teulon F. (2018), Modeling recovery rates of corporate defaulted bonds in developed and developing countries, *Emerging Markets Review*. 36, 28–44, https://doi.org/10.1016/j.ememar.2018.03.001, 2-s2.0-85045029245.

Niculescu-Mizil, A., & Caruana, R. (2005, July). Obtaining Calibrated Probabilities from Boosting. In UAI (Vol. 5, pp. 413-20).

Nielsen, D. (2016). Tree Boosting With XGBoost - *Why Does XGBoost Win "Every" Machine Learning Competition?*

Ozturkkal, B., & Wahlstrøm, R. (2022), Explaining mortgage defaults using SHAP and LASSO. http://dx.doi.org/10.2139/ssrn.4212836

Prado, J. W.; de Castro Alcântara, V.; de Melo Carvalho, F.; Vieira, K. C.; Machado, L. K. C.; & Tonelli, D. F. (2016) Multivariate Analysis of Credit Risk and Bankruptcy Research Data: A Bibliometric Study Involving Different Knowledge Fields (1968–2014). *Scientometrics*, *106*, 1007–1029.

Roberts, A. (2022). *What Is PR AUC?* https://arize.com/blog/what-is-pr-auc/#:~:text=Amber%20Roberts,-Machine%20Learning%20Engineer&text=AUC%2C%20short%20for%20area%20under,the%20positive%20and%20negative%20classes.

Sirmans, G. S., MacDonald, L., & Macpherson, D. A. (2006). The Value of Housing Characteristics: A MetaAnalysis. Journal of Real Estate Finance and Economics, 33(3), 215-240. https://doi.org/10.1007/s11146-006-9983- 5

Wang, F.; Ding, L.; Yu, H.; & Zhao, Y. (2020) Big data analytics on enterprise credit risk evaluation of E-Business platform. *Inf. Syst. E-Bus. Manag. 18*, 311–350.

XGBoost developers (2018). xgboost, release 0.80, September, https://media.readthedocs.org/pdf/xgboost/latest/xgboost.pdf.

Zhang M. J. (2018) Risk and Prevention of Commercial Bank Mortgage *Economic and Trade Practice* 18 155-157