Machine Learning Explainability in Mortgage Credit Approving: An QII Approach

Yichen Gong, Frank Zhao, Yihan Xu

Abstract

In this research, we aim to study the explainability issue of machine learning models in mortgage credit approving using a Quantitative Input Influence (QII) approach. In particular, we wish to examine whether discrimination exists in the algorithmic mortgage approval process. The non-transparent nature of machine learning models means their outputs lack explainability, so we employed the QII game theoretic approach to attempt explaining them using Shapley values. We found no evidence that our Xgboost model that make mortgage approval decisions is discriminative against the minority races. This research will contribute to existing literature on machine learning explainability and provide regulators with a better tool to monitor the potential racial or sexual discrimination in mortgage approving processes.

Introduction

Improvements in algorithmic decision-making techniques and better access to high quality borrower data have altered the way lenders make credit decisions [1]. Nowadays, machine learning programs are ubiquitous in financial institutions, ranging from banks, insurance companies, mortgage providers, etc. This modern technique is critical to financial institutions in the sense that it has stronger power in credit risk prediction than traditional methods and thus can bring higher profitability. However, there is also concern that the algorithmic decision-making process is often non-transparent [2].

An improvement that will be beneficial for the multiple stakeholders is to uncover the "black-box" decision-making process and add some explainability to the model. In addition, an explainable model can help researchers and policy makers examine the potential racial or sexual discrimination in mortgage approving practices. In essence, the machine learning explainability method attempts to highlight the model's salient features by reverse engineering it, providing insights into the complex model's inner workings. For example, Bracke, et al. (2019) used the Quantitative Input Influence (QII) method to predict mortgage defaults and uncover key drivers of mortgage defaults.

In this paper, we aim to use a similar machine learning explainability approach in a related applied setting: simulating the credit approval decision-making process and uncovering its determining factors. In essence, we are studying a classification process of whether a mortgage application would be approved or not, based on multiple lender-specific inputs, where the explainability techniques will provide a certain extent of transparency to the algorithmic decision-making process and enable us to examine whether any types of discrimination exist in mortgage approval.

Dataset

For our analysis, we use data from the U.S. mortgage market collected by the Consumer Financial Protection Bureau (CFPB). The information in the Home Mortgage Disclosure Act (HMDA) Dataset includes Loan Type, Loan Purpose, Action Taken, Race, Sex, Ethnicity, Loan-to-Value Ratio, Income, Debt to Income Ratio, etc. Among these features, "Action Taken" is at our predicting interest, as it "indicates the action [that the financial institution] took on the covered loan or application".

Stage/Year	2018	2019
Before Pre-processing	1,513,977	1,755,872
After Pre-processing	829,983	857,568

Table 1: Number of Observations

In particular, we are using the HMDA data in 2018 and 2019, downloaded directly from CFPB. Our rationale is that (1) a change of data schema occurred after 2017; (2) the dataset itself is large enough with around 150,000,000 observations per year; (3) recent data have more predictive power for future mortgage approval; (4) we would like to avoid the skewing effects of the Covid-19 pandemic. In line with the time-series nature of the HMDA Data, we would like to use the mortgage applications in 2018 as training and validation sets, and use the 2019 applications as the testing set. Notably, since the original data size of 150,000,000 requires too much computational power, we randomly down-sampled the dataset to 1% of its initial size, while still keeping enough observations per year (0.8 million) for training and testing purposes after preprocessing. The number of observations is reported in Table 1.

Feature	# Missing
activity_year	0
total_units	0
occupancy_type	0
construction_method	0
commercial_purpose	0
applicant_age	0
reverse_mortgage	0
lien_status	0
open_end_line_of_credit	0
loan_type	0
preapproval	0
action_taken	0
derived_sex	0
derived_race	0
derived_ethnicity	0
loan_purpose	0
submission_of_application	0
income	16,801
property_value	20,568
debt_to_income_ratio	30,206
combined_loan_to_value_ratio	47,766

Table 2: Feature Space and Missing Values

The HMDA dataset contains more than 90 different features, which is excessive for our study. Since we are interested in simulating the credit approval decision-making process and providing explainability, only 21 relevant features are used. Our selection of features is based on the HMDA documentation as well as relevant mortgage research [1] [3].

Upon close inspection, we found that there were many missing values in the dataset, which requires further processing. In particular, missing numerical values are replaced by the sample mean values, and categorical values listed as "N/A" are dropped from the dataset. Table 2 shows a summary of the feature space and missing values after preprocessing before filling null values with their mean. The relative number of missing values is acceptable. In addition, a normalization process is applied to the datasets.

Solution

To have a deeper look into our dataset and examine whether the algorithmic decision-making process has brought discrimination against certain groups without noticing its users, we propose three steps.

First Step: Briefly examine the internal structure of our dataset using correlation matrix and PCA. As machine learning is a "garbage in, garbage out" technique, A dataset with innate discrimination will definitely result in a discriminative model, hence we check it.

Second Step: Model training and parameter fine tuning. Before we start to interpret our model, it is important to have a model that can mimic the real-life decision-making process. As a result, we want to train models with relatively competitive AUROC for this classification problem. Also, we classify these models into three types according to their explainability.

(i) Explainable: Linear models which directly give weights to the inputs, so the coefficients will be exactly what we need for model explanation.

(ii) Probabilistic: Not directly interpretable, but give probabilistic estimates of the classification results. We would like to examine how the probability estimates will change by some intervention on the inputs.

(iii) Non-Probabilistic: Directly output classification results instead of probabilities. Explaining these models requires a better understanding of the context, but we still include the classification results of Random Forest for the readers' reference.

Third Step: Model interpretation. This step sheds light on the main interest of our paper. As summarized above, different models have different levels of interpretability, and various model explanation methods exist, ranging from Model Specific ones (e.g. feature importance in decision trees) to Model Agnostic ones (e.g., LIME, QII). In this paper, we would like to explain the Probabilistic model that gives us the best AUROC, using the method of Quantitative Input Influence as proposed in Datta, et al [2]. QII captures the degree of influence that inputs have on the output of the model. We focus on Unary QII in this paper, which decorrelates the features and examines each feature's partial effect on the output [2][4].

Since the realization of the QII algorithm is only valid for Python 2, we pivot to the SHAP realization that is proposed by Lundberg and Lee [5], which is "designed to closely align with the Shapley regression, Shapley sampling, and quantitative input influence feature attributions, while also allowing for connections with LIME, DeepLIFT, and layer-wise relevance propagation." [5]. Using the SHAP realization, we are able to answer the following questions:

(i) For one specific person, does his race influence the model decision positively or negatively?

(ii) How will being of a certain race influence the model output collectively?

(iii) How will being of a certain race influence the model's judgment on other features?

Results and Discussion

We change the order of demonstration here for clarity.

A. Model Fitting (Second Step)

We do an exhaustive search over specified parameter values for our models. Since the limitation of computing power, we select 2 parameters every time. Each time, we choose the parameter pair with the highest rating and use them as hyperparameters in the next step. For the Xgboost model, we test 6 parameters. And for the Random Forest Model, we also test 6 parameters.

Parameters	Values
max_depth	5-10
min_child_weight	1-4
learning_rate	0.1, 0.5, 1
subsample	0.1, 0.5, 1
reg_alpha	10, 100, 1000
n_estimators	500, 1000, 1500

Table 3: Pram_gird of Xgboost

The AUROC of our models slightly improves after parameter fine-tuning, as shown in Table 4

	Before Grid Search					
	AUROC(%)		**Recall(%)**		**Precision(%)**	
	Testing	Training	Testing	Training	Testing	Training
LR	81.65	81.54	97.19	96.33	86.74	82.99
RF	83.15	99.99	95.51	99.97	88.71	99.89
XGBoost	85.65	86.94	97.14	95.81	88.73	86.48

	After Grid Search					
	AUROC(%)		**Recall(%)**		**Precision(%)**	
	Testing	Training	Testing	Training	Testing	Training
LR	81.65	81.54	97.19	96.33	86.74	82.99
RF	84.98	87.44	97.68	97.21	88.05	85.95
XGBoost	85.69	87.39	96.64	95.40	89.05	86.97

Table 4: Grid Search Results

Graphs of the fine-tuning process is reported in Appendix A.

B. Discrimination Analysis

a. Dataset Analysis (First Step)

As described above, our analysis of the dataset is about the correlation between an applicant's race and the results s/he got. Ideally, there should not be too much correlation between these two features so that our dataset is not an innately discriminative one.

Correlation Matrix: As shown in Table 5 the correlation between *action taken: approved* and race types is at the percent level. We believe that this correlation is small enough to indicate that we don't have too much innate discrimina-

tion. A more detailed correlation matrix can be found in Appendix B.

Correlation with Action_Taken_1:Approved	
2 or more minority races	-1.9%
American Indian or Alaska Native	-3.1%
Asian	-0.4%
Black or African American	-9.2%
Joint	0.9%
Native Hawaiian	-2.6%
White	7.8%

Table 5: Selected Feature Correlation

PCA: As PCA is a variance-explainer, we assume that: The more similarity information two features represent, the more linearity will display in their PCA vectors where PCA vectors are generated by the coefficients each feature has on the principle components

$$Feature_i = beta_{1i} * PC_1 + beta_{2i} * PC_2 + ...$$

$$+beta_{ki} * PC_k$$

As depicted in Figure 1, the feature *Approved* seems to be almost orthogonal to the features related to race on the two first components, which also supports the validity of our dataset.

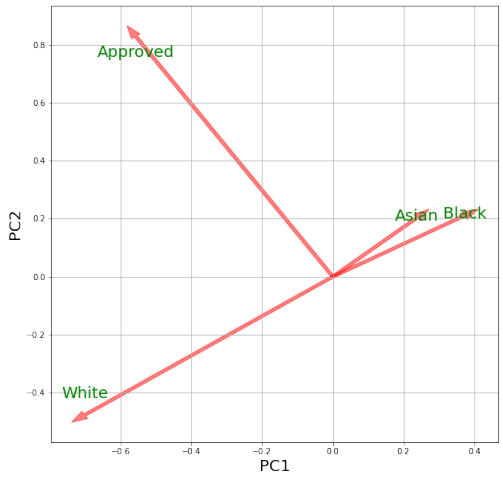

Figure 1: PCA Vectors of Selected Features

According to our Dataset Analysis, both correlation matrix and PCA vector show that the relationship between a person's race and his application results exists, but is by far from strong. Following this observation, we can proceed to examine whether our models will keep reflecting the relationship in our dataset, exaggerate the relationship and make an even more biased decision, or get rid of the biasedness.

b. Model Analysis (Third Step)

In this part, we will first demonstrate the feature importance of our Logistic Regression model. Then we pivot to answer the three questions proposed at the end of Section III, using a combination of SHAP value and Xgboost model.

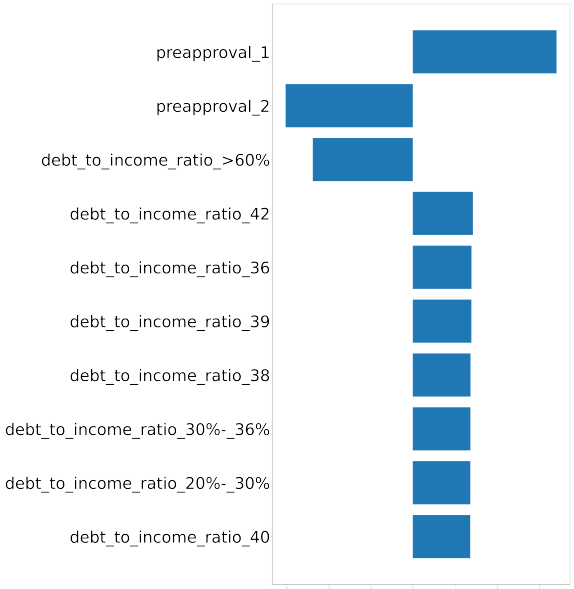

Figure 2: 10 Important Features in LR model

Logistic Regression: In our baseline model, i.e., LR regression, no race features are among the top 10 most important features, as shown in Figure 2. This is going to say, when applying the most interpretable model, we can be quite assured that it doesn't attribute too much weight to an applicant's race. In other words, model users can directly use these measures to respond to the applicants and regulators about the limited biasedness during the decision-making process. However, there are always tradeoffs between model accuracy and interpretability. Should model users pivot to more complex models for better classification results at the risk of violating discrimination tests? Or should they keep using the simplistic model? Our results from the Xgboost model may give some practical insights into this dilemma.

Xgboost: As shown in IV.A, Xgboost has an approximately 4% improvement in AUC, but be-

fore moving to the model, we believe it's of great importance to check what the model is doing by answering the three questions that we proposed above using SHAP value.

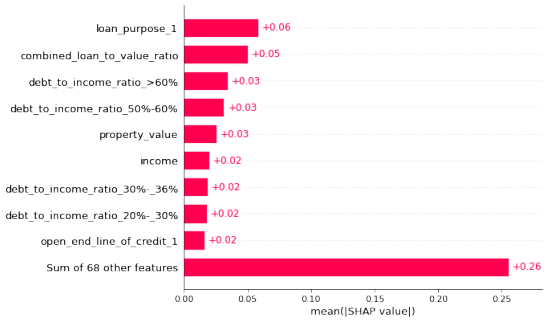

Figure 3: Shap Values of Important Features

1. Effect on a Certain Applicant: To demonstrate this effect, we randomly select three people from our testing set, and plot out how the model finally makes the decision regarding their applications. The plots can be found in Appendix C. Red means that a certain feature pushes the application to be approved, and blue vise versa. For applicant 199 who got rejected, we find his sex is the most critical reason, and for applicant 800, his low loan-to-value ratio has almost secured a mortgage for him. Yet another interesting observation on applicant 300, who also get rejected, is that his race is among the 10 most important features. Is this going to say that our model is discriminative in applicant 300's case? Not necessarily. Although race is listed among the top 10 features, its impact is almost neglectable compared with the impact of a high debt to income ratio. As a result, the model user can possibly defend the model's unbiasedness by demonstrating these graphs to applicants in a case-by-case manner, which is hardly realizable using traditional interpretation methods.

2. Average Effects over a Subsample: The graph below is the "global feature importance plot, where the global importance of each feature is taken to be the mean absolute value for that feature over all the given samples" [6]. In this paper, we only randomly take 1000 applicants from the testing set for demonstration. None of the race-related features are among the top 10. It is quite an assuring result that our model, holistically (or on average), is not discriminative against a certain race. At the same time, the model's decision method is in great accordance with human experience and logic. Loan purpose, lTV ratio, and DTI ratios which have been proven important by economic theories are given high importance [4].

In Appendix C we give a beeswarm plot for all 1000 selected applicants. According to SHAP documentation, it is "an information-dense summary" of the model, where the SHAP values of all selected applicants on all model features are plotted. Readers may look through the graph for further understanding of the model.

3. Interaction between Race and other Features:

The final question that we would like to give thoughts on is whether *Race* will influence the model's decision on other features. In accordance with the results before, our model shows no direct signs to support the hypothesis. From Figure 4, it's easy to see that the distribution of SHAP values of *Debt to Income ratio (DTL ratio)* is quite similar between Whites and Non-Whites. For example, although our model punishes a high DTL ratio quite heavily, we cannot find a significant difference between the punishment of Whites (red dots) and Non-Whites (blue dots). This is another strong evidence to support our model's unbiasedness. The graphs about *Loan Purpose* and *Loan to Value ratio* can be found in Appendix C.

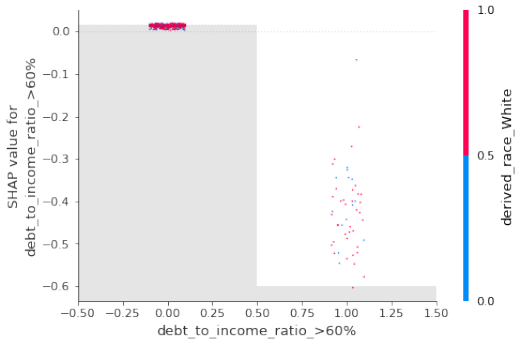

Figure 4: Shap Interaction Plot, White vs DTI

Conclusions

Borrowing the idea of Shapely values from economics and mathematicians, more and more handy methods are developed for model developers to explain the crafts that they trained. Our

work demonstrates how we can use one of the methods in the Arsenal, namely, SHAP, to give a brief analysis of the model's biasedness. Fortunately, by examining the model through three perspectives, we don't find any direct signals that our model is race-discriminative. This result is important in the sense that mortgage companies may use it to pass the discrimination tests from the regulators (Q2, Q3), and applicants can get a better understanding of why they got rejected (Q1). However, this is only a small step forward and many improvements can be made in the future. Firstly, our analysis only focuses on race discrimination, while biasedness can be everywhere, ranging from age to sex. Secondly, a more dedicated feature selection/engineering process may make our classification result better. Thirdly, it is humans who have the right to interpret the SHAP values. The current analysis method relies heavily on human beings' understanding of the SHAP results, instead of applying objective metrics or methods. A standardized process should be developed in the future regarding the "decoding" of the algorithmic decision-making process, and the final goal is always to make the "black-box" much more transparent.

References

[1] A. Fuster, P. Goldsmith-Pinkham, T. Ramadorai, and A. Walther, "Predictably unequal? the effects of machine learning on credit markets," *The Journal of Finance*, 2021.

[2] A. Datta, S. Sen, and Y. Zick, "Algorithmic transparency via quantitative input influence: Theory and experiments with learning systems," *Proceedings of IEEE Symposium on Security Privacy 2016*, pp. 598–617, 2016.

[3] "A guide to hmda reporting," *Federal Financial Institutions Examination Council*, 2021. [Online]. Available: ffiec.gov/hmda/pdf/2021Guide.pdf

[4] C. L. Foote and P. S. Willen, "Mortgage-default research and the recent foreclosure crisis," *Annual Review of Financial Economic*, 2018.

[5] S.-I. Lee and S. M. Lundberg, "A unified approach to interpreting model predictions," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[6] S. M. Lunderberg, "Shap documentation." [Online]. Available: shap.readthedocs.io/en/latest/example_notebooks/api_examples/plots/beeswarm.html

[7] P. Bracke, A. Datta, C. Jung, and S. Sen, "Machine learning explainability in finance: An application to default risk analysis," *Bank of England Staff Working Paper*, vol. 816, 2019.

[8] G. Ratnayaka, "Probability and machine learning? probabilistic vs non- probabilistic machine learning models," *Medium*, 2020. [Online]. Available: medium.com/nerd-for-tech/probability-and-machine-learning-570815bad29d

[9] "Covid-19: The impacts on global residential mortgage markets," *Deloitte*, 2020.

[10] "Quantitative input influence tool," *cmu-transparency*. [Online]. Available: github.com/cmu-transparency/tool-qii

Appendix A Fine Tuning Process

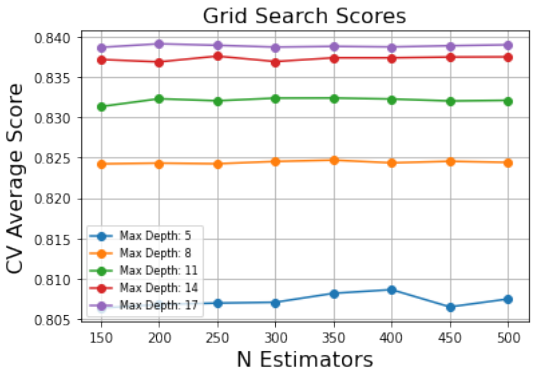

Figure 5: Grid search for Random Forest parameters =n_estimators & max_depth

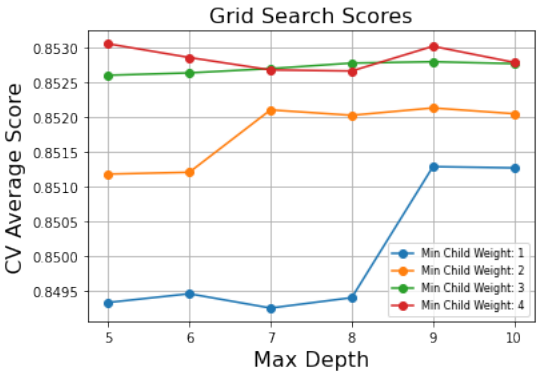

Figure 8: Grid search for Xgboost parameters = max_depth & min_child_weight

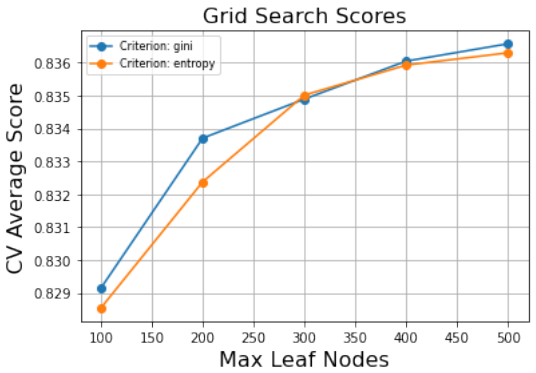

Figure 6: Grid search for Random Forest parameters = max_leaf_nodes & criterion

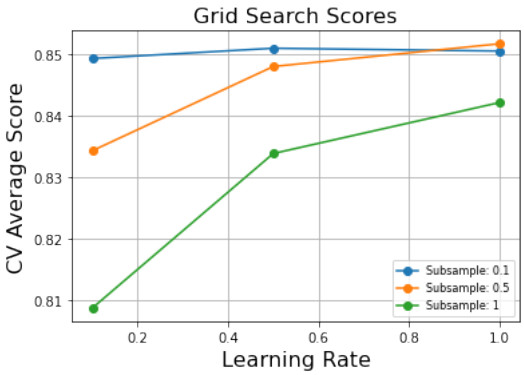

Figure 9: Grid search for Xgboost parameters = learning_rate & subsample

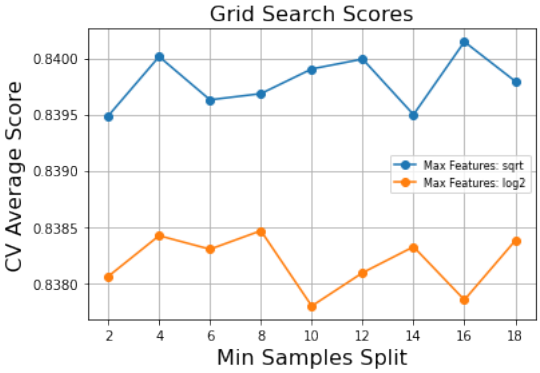

Figure 7: Grid search for Random Forest parameters = min_samples_split & max_features

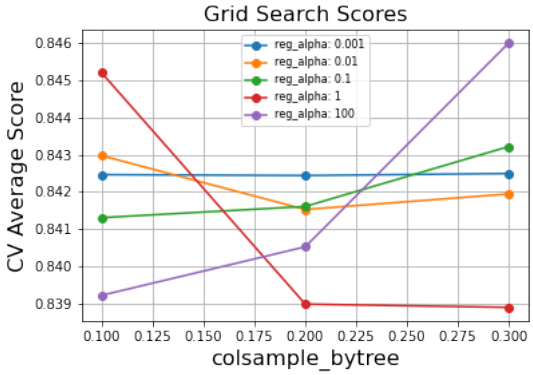

Figure 10: Grid search for Xgboost parameters = reg_alpha & n_estimators

Appendix B Correlation Matrix

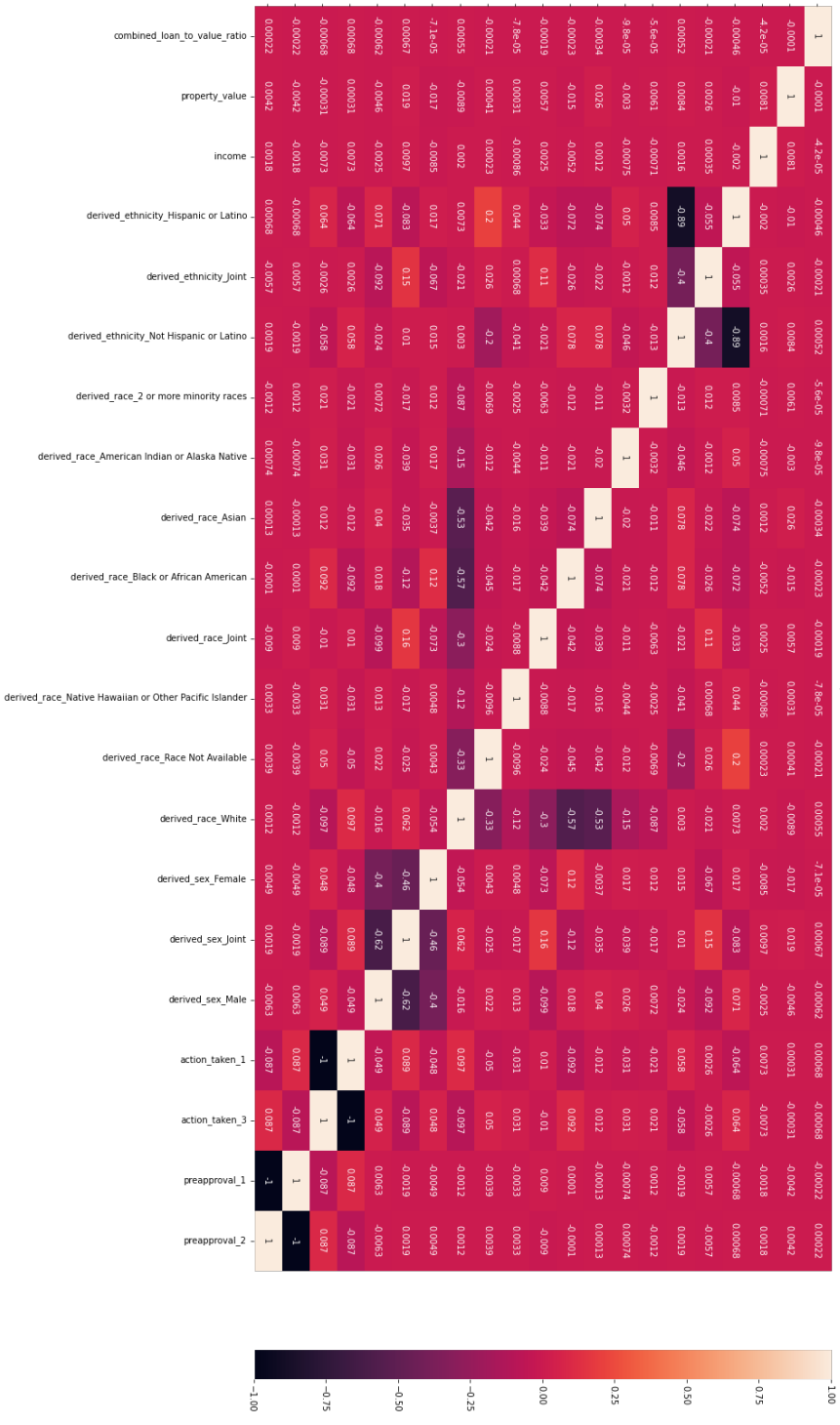

Figure 11: Correlation Matrix

Appendix C Other SHAP Results

Beeswarm Plot of the SHAP Values

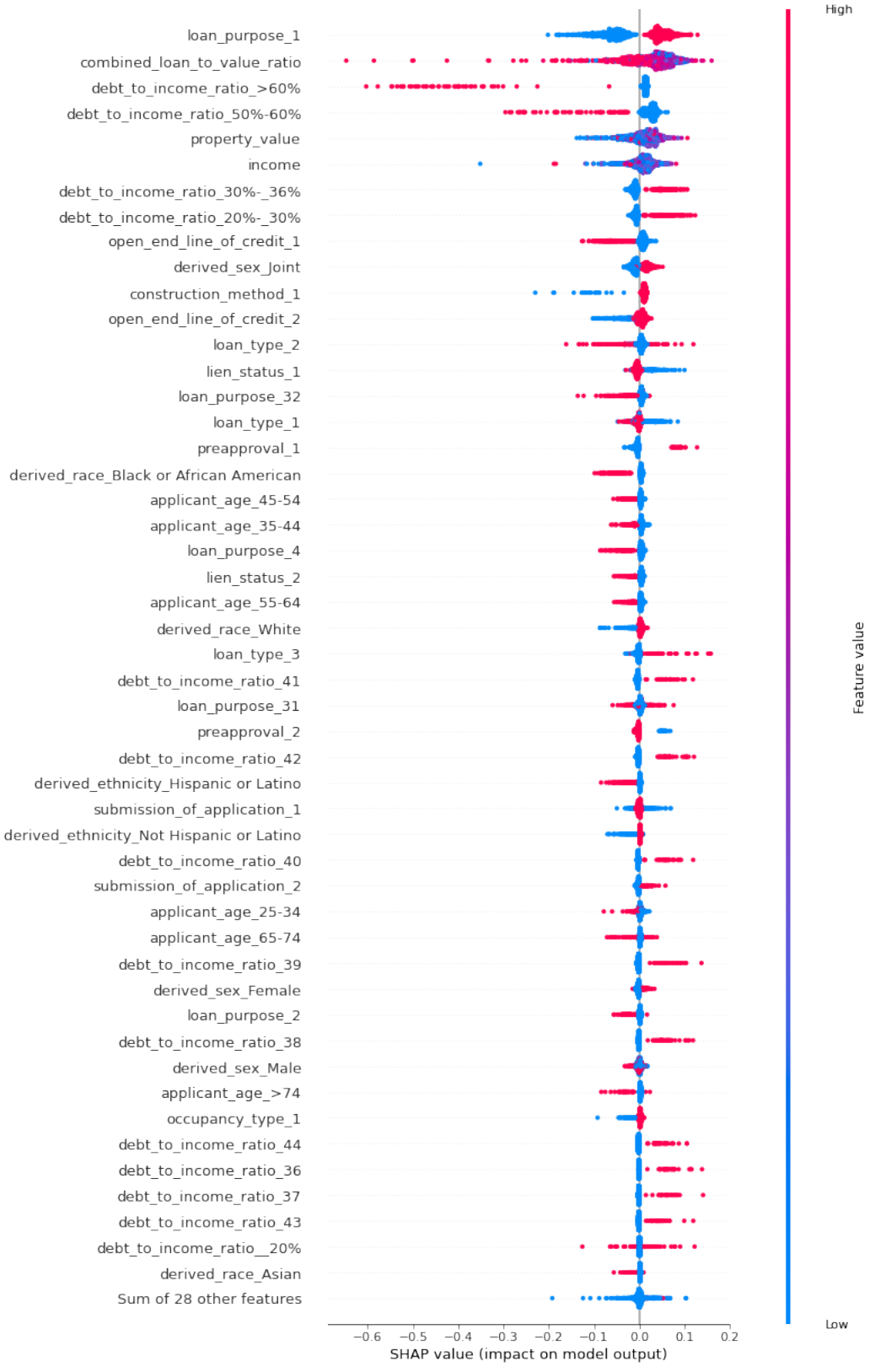

Figure 12: Beeswarm Plot, Xgboost

SHAP Values for Certain Applicants

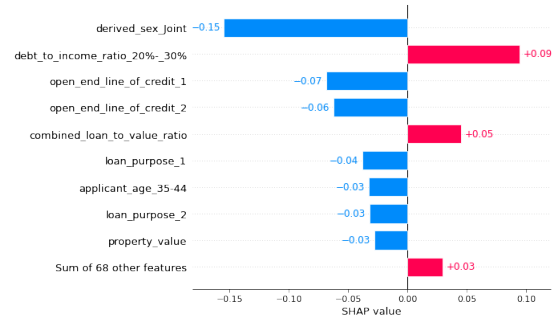

Figure 13: SHAP Value, Applicant 199

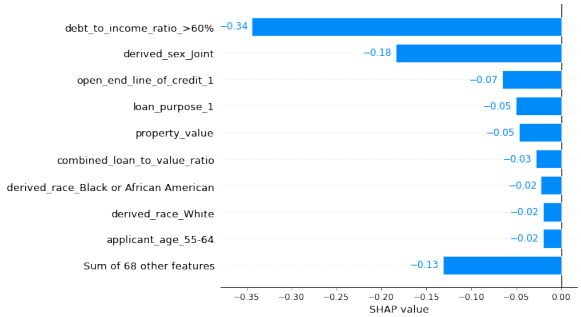

Figure 14: SHAP Value, Applicant 300

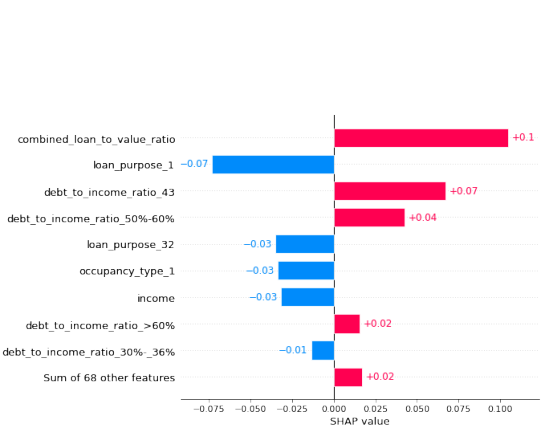

Figure 15: SHAP Value, Applicant 800

SHAP Interaction Plots

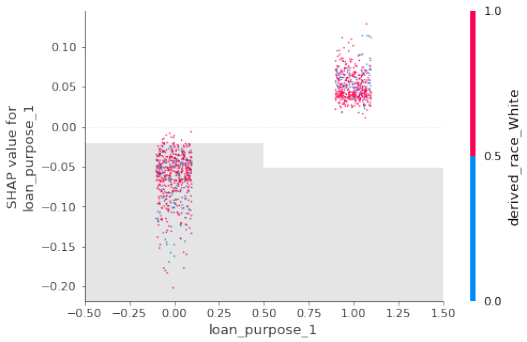

Figure 16: Shap Interaction Plot, White vs Loan Purpose

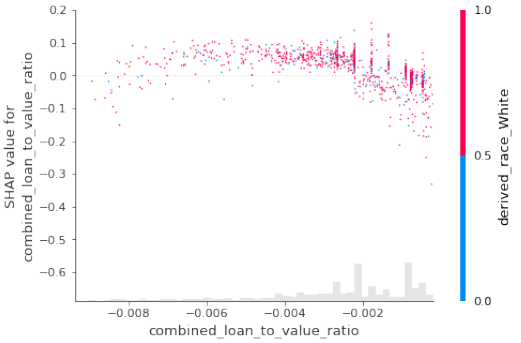

Figure 17: Shap Interaction Plot, White vs LTV

Appendix D Results

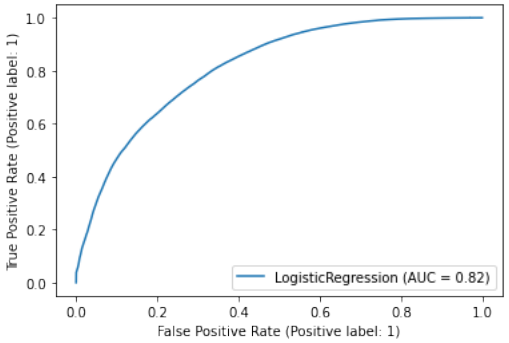

Figure 18: LR_AUC_training

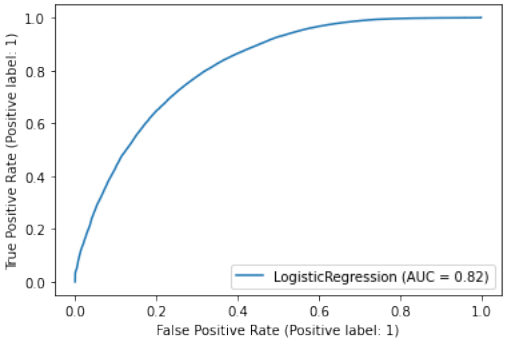

Figure 19: LR_AUC_testing

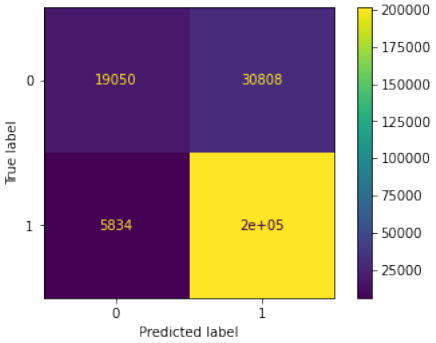

Figure 20: LR_confusion_testing

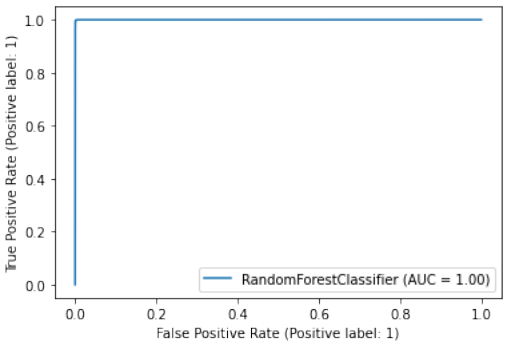

Figure 21: RFC_AUC_training_before_GV

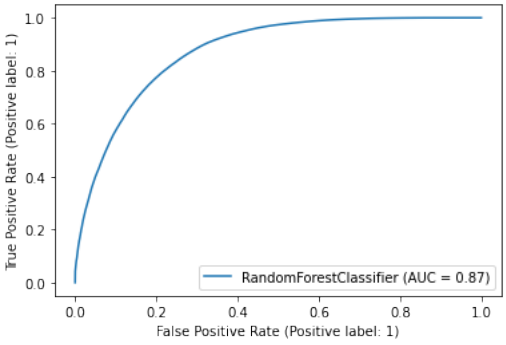

Figure 22: RFC_AUC_training_after_GV

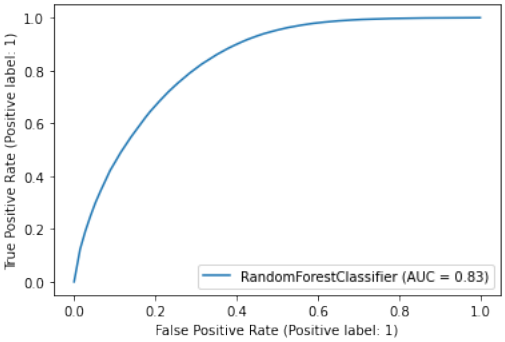

Figure 23: RFC_AUC_testing_before_GV

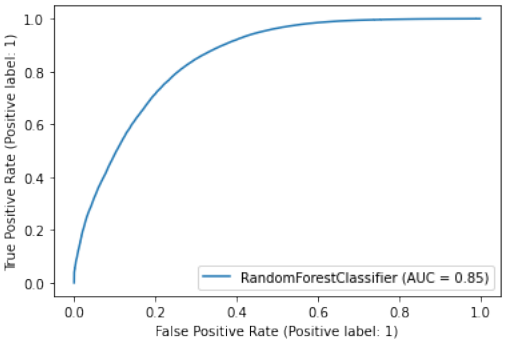

Figure 24: RFC_AUC_testing_after_GV

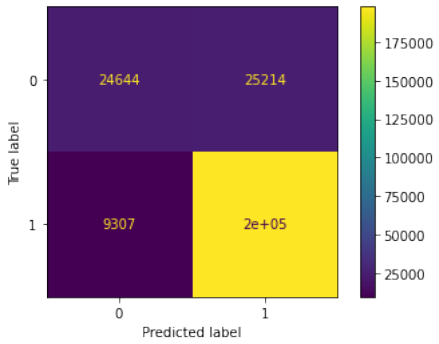

Figure 25: RFC_confusion_testing_before_GV

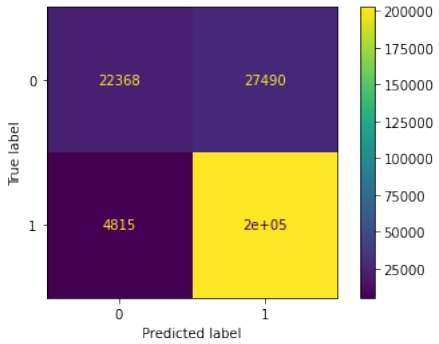

Figure 26: RFC_confusion_testing_after_GV

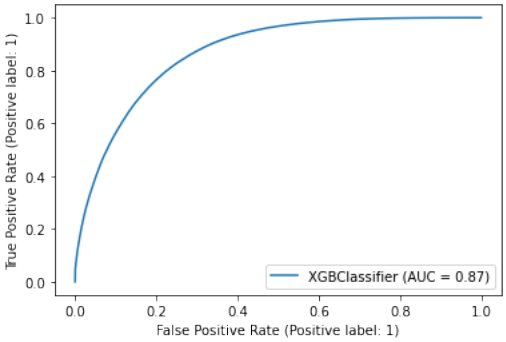

Figure 27: XGB_training_before_GV

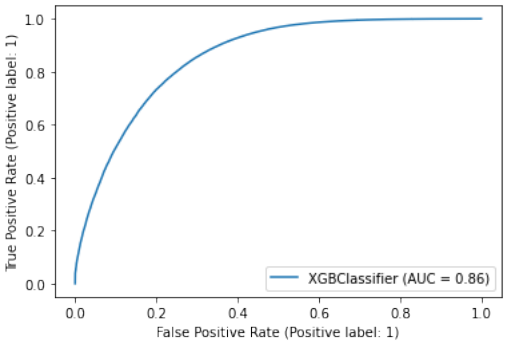

Figure 28: XGB_testing_before_GV

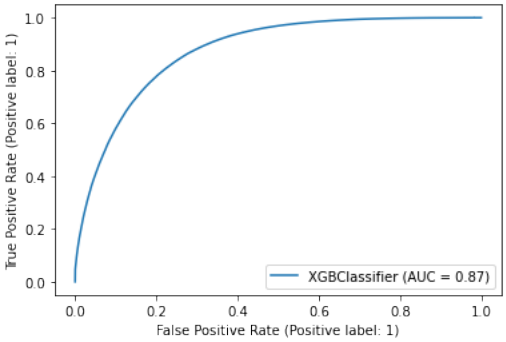

Figure 29: XGB_AUC_training_after_GV

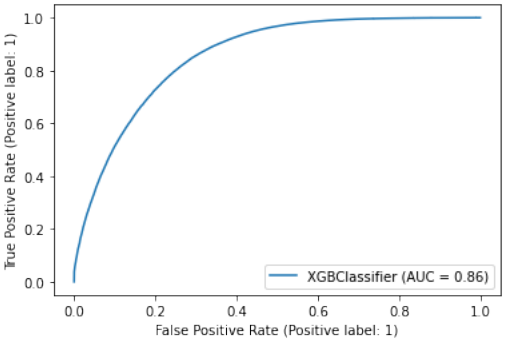

Figure 30: XGB_AUC_testing_after_GV

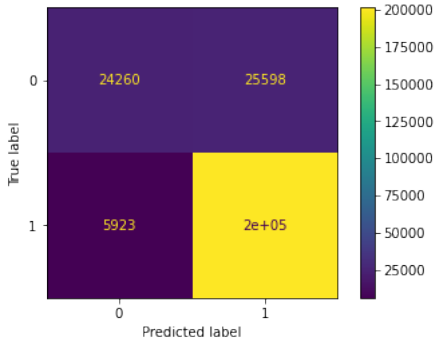

Figure 31: XGB_confusion_testing_before_GV

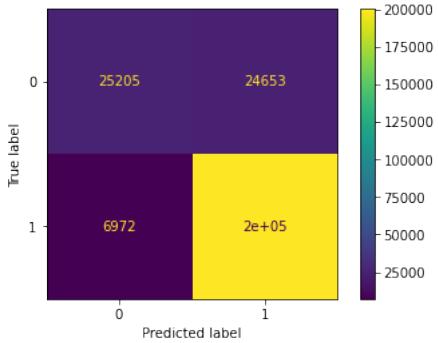

Figure 32: XGB_confusion_testing_after_GV