Article

A Responsible Machine Learning Workflow

With Focus on Interpretable Models, Post-hoc Explanation, and Discrimination Testing

Navdeep Gill [1,†], Patrick Hall [1,3,†,*], Kim Montgomery [1,†], and Nicholas Schmidt [2,†]

[1] H2O.ai
[2] BLDS, LLC
[3] The George Washington University
* Correspondence: phall@h2o.ai; nschmidt@bldsllc.com
† All authors contributed equally to this work.

Version February 6, 2020 submitted to Information

Abstract: This manuscript outlines a viable approach for training and evaluating machine learning (ML) systems for high-stakes, human-centered, or regulated applications using common Python programming tools. The accuracy and intrinsic interpretability of two types of constrained models, monotonic gradient boosting machines (MGBMs) and explainable neural networks (XNNs), a deep learning architecture well-suited for structured data, are assessed on simulated data and publicly available mortgage data. For maximum transparency and the potential generation of personalized adverse action notices, the constrained models are analyzed using post-hoc explanation techniques including plots of partial dependence (PD) and individual conditional expectation (ICE) and with global and local Shapley feature importance. The constrained model predictions are also tested for disparate impact (DI) and other types of discrimination using measures with long-standing legal precedents, adverse impact ratio (AIR), marginal effect (ME), and standardized mean difference (SMD), and also with straightforward group fairness measures. By combining interpretable models, post-hoc explanations, and discrimination testing with accessible software tools, this text aims to provide a template workflow for important ML applications that require high accuracy and interpretability and that mitigate risks of discrimination.

Keywords: Deep Learning; Disparate Impact; Explanation; Fairness; Gradient Boosting Machine; Interpretable; Machine Learning; Neural Network; Python

---

0. Introduction

Responsible artificial intelligence (AI) has been variously conceptualized as AI-based products or projects that use transparent technical mechanisms, that create appealable decisions or outcomes, that perform reliably and in a trustworthy manner over time, that exhibit minimal social discrimination, and that are designed by humans with diverse experiences, both in terms of demographics and professional backgrounds.[1] Although responsible AI is today a somewhat broad and amorphous notion, at least one aspect is becoming clear. ML models, a common application of AI, can present serious risks. ML models can be inaccurate and unappealable black-boxes, even with the application of newer post-hoc explanation techniques [1].[2] ML models can perpetuate and exacerbate discrimination [2], [3], [4], and ML models can be hacked, resulting in manipulated model outcomes or the exposure

---

1 See: Responsible Artificial Intelligence, *Responsible AI: A Framework for Building Trust in Your AI Solutions*, PwC's Responsible AI, Responsible AI Practices.
2 See: *When a Computer Program Keeps You in Jail*.

of proprietary intellectual property or sensitive training data [5], [6], [7], [8]. This manuscript makes no claim that these interdependent issues of ML opaqueness, discrimination, privacy harms, and security vulnerabilities have been resolved, even as singular entities, and much less as complex intersectional phenomena. However, Sections 1, 2, and 3 do propose some specific technical countermeasures, mostly in the form interpretable models, post-hoc explanation, and DI and discrimination testing, that responsible practitioners can use to address a subset of these vexing problems.[3,4]

Section 1 describes methods and materials, including training datasets, interpretable and constrained models, post-hoc explanations, tests for DI and other social discrimination, and public and open source software resources associated with this text. In Section 2, interpretable and constrained modeling results are compared to less interpretable and unconstrained models, and post-hoc explanation and discrimination testing results are also presented for interpretable models. Of course, an even wider array of tools and techniques are likely helpful to fully minimize discrimination, inaccuracy, privacy, and security risks associated with ML models. Section 3 puts forward a more holistic responsible ML modeling workflow, and addresses the burgeoning Python ecosystem for responsible AI, along with appeal and override of automated decisions, and discrimination testing and remediation in practice. Section 4 closes this manuscript with a brief summary of the outlined methods, materials, results, and discussion.

1. Materials and Methods

The simulated data (see Subsection 1.1) is based on the well-known Friedman datasets. Its known feature importance and augmented discrimination characteristics are used to gauge the validity of interpretable modeling, post-hoc explanation, and discrimination testing techniques [10], [11]. The mortgage data (see Subsection 1.2) is sourced from the Home Mortgage Disclosure Act (HMDA) database and is a fairly realistic data source for demonstrating the template workflow.[5] To provide a sense of fit differences, performance of more interpretable constrained ML models and less interpretable unconstrained ML models is compared on simulated data and collected mortgage data. Because the unconstrained ML models, gradient boosting machines (GBMs, e.g. [12], [13]) and artificial neural networks (ANNs, e.g. [14], [15], [16], [17]), do not exhibit convincing accuracy benefits on the simulated or mortgage data and can also present the unmitigated risks discussed above, further explanation and discrimination analyses are applied only to the constrained, interpretable ML models [1], [18], [19]. Here, MGBMs[6] (See Subsection 1.3) and XNNs (see Subsection 1.4, [20] [21]) will serve as those more interpretable models for subsequent explanatory and discrimination analyses. MGBM and XNN interpretable model architectures are selected for the example workflow because they are straightforward variants of popular unconstrained ML models. If practitioners are working with GBM and ANN models, it should be relatively uncomplicated to also evaluate the constrained versions of these models.

The same can be said of the selected explanation methods and discrimination tests. Due to their post-hoc nature, they can often be shoe-horned into existing ML workflows and pipelines. Presented explanation techniques include PD and ICE (see Subsection 1.5) and Shapley values (see Subsection 1.6) [13], [22], [23], [24]. PD, ICE, and Shapley values provide direct, global, and local summaries and descriptions of constrained models without resorting to the use of intermediary and approximate

[3] This text and associated software are not, and should not be construed as, legal advice or requirements for regulatory compliance.

[4] In the United States (US), interpretable models, explanations, DI testing, and the model documentation they enable may be required under the Civil Rights Acts of 1964 and 1991, the Americans with Disabilities Act, the Genetic Information Nondiscrimination Act, the Health Insurance Portability and Accountability Act, the Equal Credit Opportunity Act (ECOA), the Fair Credit Reporting Act (FCRA), the Fair Housing Act, Federal Reserve Supervisory and Regulatory (SR) Letter 11-7, and the European Union (EU) General Data Protection Regulation (GDPR) Article 22 [9].

[5] See: Mortgage data (HMDA).

[6] As implemented in `XGBoost` or `h2o`.

surrogate models. Discussed discrimination testing methods (see Subsection 1.7) include AIR, ME, and SMD [2], [25], [26].[7] Accuracy and other confusion matrix measures are also reported by demographic segment [27]. All outlined materials and methods are implemented in open source Python code, and are made available on GitHub (see Subsection 1.8).

1.1. Simulated Data

Simulated data is created based on a signal-generating function, f, applied to input data, $\mathbf{X}$, first proposed in Friedman [10] and extended in Friedman *et al.* [11]:

$$f(\mathbf{X}) = 10\sin(\pi\, X_{\text{Friedman},1}\, X_{\text{Friedman},2}) + 20(X_{\text{Friedman},3} - 0.5)^2 + 10\, X_{\text{Friedman},4} + 5\, X_{\text{Friedman},5} \quad (1)$$

where each $X_{\text{Friedman},j}$ is a random uniform feature in $[0, 1]$. In Friedman's texts, a Gaussian noise term was added to create a continuous output feature for testing spline regression methodologies. In this manuscript, the signal generating function and input features are modified in several ways. Two binary features, a categorical feature with five discrete levels, and a bias term are introduced into f to add a degree of complexity that may more closely mimic real-world settings. For binary classification analysis, the Gaussian noise term is replaced with noise drawn from a logistic distribution and coefficients are re-scaled to be $\frac{1}{5}$ of the size of those used by Friedman, and any $f(\mathbf{X})$ value above 0 is classified as a positive outcome, while $f(\mathbf{X})$ values less than or equal to zero are designated as negative outcomes. Finally, f is augmented with two hypothetical protected class-control features with known dependencies on the binary outcome to allow for discrimination testing. The simulated data is generated to have eight input features, twelve after numeric encoding of categorical features, and a binary outcome, two class-control features, and 100,000 instances. The simulated data is then split into a training and test set, with 80,000 and 20,000 instances, respectively. Within the training set, a 5-fold cross-validation indicator is used for training all models. For an exact specification of the simulated data, see the software resources referenced in Subsection 1.8.

1.2. Mortgage Data

The mortgage dataset analyzed here is a random sample of consumer-anonymized loans from the HDMA database. These loans are a subset of all originated mortgage loans in the 2018 HMDA data that were chosen to represent a relatively comparable group of consumer mortgages. A selection of features is used to predict whether a loan is *high-priced*, i.e., the annual percentage rate (APR) charged was 150 basis points (1.5%) or more above a survey-based estimate of other similar loans offered around the time of the given loan. After data cleaning and preprocessing to encode categorical features and create missing markers, the mortgage data contains ten input features and the binary outcome, *high-priced*. The data is split into a training set with 160,338 loans and a marker for 5-fold cross-validation and a test set containing 39,662 loans. While lenders would almost certainly use more information than the selected features to determine whether to offer and originate a high-priced loan, the selected input features (loan to value (LTV) ratio, debt to income (DTI) ratio, property value, loan amount, introductory interest rate, customer income, etc.) are likely to be some of the most influential factors that a lender would consider. See the resources put forward in Section 1.8 and Appendix A for more information regarding the HMDA mortgage data.

1.3. Monotonic Gradient Boosting Machines

MGBMs constrain typical GBM training to consider only tree splits that obey user-defined positive and negative monotonicity constraints, with respect to each input feature, X_j, and a target feature, $\mathbf{y}$,

[7] See: Part 1607 - Uniform Guidelines on Employee Selection Procedures (1978) §1607.4.

independently. An MGBM remains an additive combination of B trees trained by gradient boosting, T_b, and each tree learns a set of splitting rules that respect monotonicity constraints, Θ_b^{mono}. For some instance, $\mathbf{x}$, a trained MGBM model, g^{MGBM}, takes the form:

$$g^{\text{MGBM}}(\mathbf{x}) = \sum_{b=0}^{B-1} T_b\left(\mathbf{x}; \Theta_b^{\text{mono}}\right) \tag{2}$$

As in unconstrained GBM, Θ_b^{mono} is selected in a greedy, additive fashion by minimizing a regularized loss function that considers known target labels, the predictions of all subsequently trained trees in g^{MGBM}, and the b-th tree splits applied to $\mathbf{x}$, $T_b(\mathbf{x}; \Theta_b^{\text{mono}})$, in a numeric loss function (e.g., squared loss, Huber loss), and a regularization term that penalizes complexity in the current tree. See Appendices B.1 and B.2 for details pertaining to MGBM training.

Herein, two g^{MGBM} models are trained. One on the simulated data and one on the mortgage data. In both cases, positive and negative monotonic constraints for each X_j are selected using domain knowledge, random grid search is used to determine other hyperparameters, and 5-fold cross-validation and test partitions are used for model assessment. For exact parameterization of the two g^{MGBM} models, see the software resources referenced in Subsection 1.8.

1.4. Explainable Neural Networks

XNNs are an alternative formulation of additive index models in which the ridge functions are neural networks [20]. XNNs also bear a strong resemblance to generalized additive models (GAMs) and so-called explainable boosting machines (EBMs or GA^{2}M), i.e., GAMs which consider main effects and a small number of 2-way interactions and may also incorporate boosting into their training [13], [28]. XNNs enable users to tailor interpretable neural network architectures to a given prediction problem and to visualize model behavior by plotting ridge functions. A trained XNN function, g^{XNN}, applied to some instance , $\mathbf{x}$, is defined as:

$$g^{\text{XNN}}(\mathbf{x}) = \mu_0 + \sum_{k=0}^{K-1} \gamma_k n_k \left(\sum_{j=0}^{J-1} \beta_{k,j} x_j\right) \tag{3}$$

where μ_0 is a global bias for K individually specified ANN subnetworks, n_k, with weights γ_k. The inputs to each n_k are themselves a linear combination of the J modeling inputs and their associated $\beta_{k,j}$ coefficients in the deepest projection layer of g^{XNN}.

Two g^{XNN} models are trained by mini-batch stochastic gradient descent (SGD) on the simulated data and mortgage data. Each g^{XNN} is assessed in 5 training folds and in a test data partition. L_1 regularization is applied to network weights to induce a sparse and interpretable model, where each n_k and corresponding γ_k are ideally associated with an important X_j or combination thereof. The g^{XNN} models appear highly sensitive to weight initialization and batch size. Be aware that g^{XNN} model architectures may require manual and judicious feature selection due to long training times. For more details regarding g^{XNN} training, see the software resources in Subsection 1.8 and Appendices B.1 and B.3.

1.5. One-dimensional Partial Dependence and Individual Conditional Expectation

PD plots are a widely-used method for describing and plotting the estimated average prediction of a complex model, g, across some partition of data, $\mathbf{X}$, for some interesting input feature, $X_j \in \mathbf{X}$ [13]. ICE plots are a newer method that describes the local behavior of g with regard to values of an input feature in a single instance, x_j. PD and ICE can be overlaid in the same plot to create a holistic global and local portrait of the predictions for some g and X_j [22]. When $\text{PD}(X_j, g)$ and $\text{ICE}(x_j, g)$ curves diverge, such plots can also be indicative of modeled interactions in g or expose flaws in PD estimation, e.g., inaccuracy in the presence of strong interactions and correlations [22], [29]. For details regarding the calculation of one-dimensional PD and ICE, see the software resources in Subsection 1.8 and Appendices B.1 and B.4.

1.6. Shapley Values

Shapley explanations are a class of additive, locally accurate feature contribution measures with long-standing theoretical support [23], [30]. Shapley explanations are the only known locally accurate and globally consistent feature contribution values, meaning that Shapley explanation values for input features always sum to the model's prediction, $g(\mathbf{x})$, for any instance $\mathbf{x}$, and that Shapley explanation values should not decrease in magnitude for some instance of x_j when g is changed such that x_j truly makes a stronger contribution to $g(\mathbf{x})$ [23], [24]. Shapley values can be estimated in different ways, many of which are intractable for datasets with large numbers of input features. Tree SHAP (SHapley Additive exPlanations) is a specific implementation of Shapley explanations that relies on traversing internal decision tree structures to efficiently estimate the contribution of each x_j for some $g(\mathbf{x})$ [24]. Tree SHAP has been implemented in popular gradient boosting libraries such as `h2o`, `LightGBM`, and `XGBoost`, and Tree SHAP is used to calculate accurate and consistent global and local feature importance for MGBM models in Subsection 2.2.2 and Appendix E.1. Deep SHAP is an approximate Shapley value technique that creates SHAP values for ANNs [23]. Deep SHAP is implemented in the `shap` package and is used to generate SHAP values for the two g^{XNN} models discussed in Subsection 2.2.2 and Appendix E.1. For more information pertaining to the calculation of Shapley values, see Appendices B.1 and B.5.

1.7. Discrimination Testing Measures

Because many current technical discussions of fairness in ML appear inconclusive[8], this text will draw on regulatory and legal standards that have been used for years in regulated, high-stakes employment and financial decisions. The discussed measures are also representative of fair lending analyses and pair well with the mortgage data. See Appendix C for a brief discussion regarding different types of discrimination in US legal and regulatory settings and Appendix D for remarks on practical vs. statistical significance for discrimination measures. One such common measure of DI used in US litigation and regulatory settings is ME. ME is simply the difference between the percent of the control group members receiving a favorable outcome and the percent of the protected class members receiving a favorable outcome.

$$\text{ME} \equiv 100 \cdot (\Pr(\hat{\mathbf{y}} = 1 | X_c = 1) - \Pr(\hat{\mathbf{y}} = 1 | X_p = 1)) \tag{4}$$

where $\hat{\mathbf{y}}$ are the model decisions, X_p and X_c represent binary markers created from some demographic attribute, c denotes the control group (often whites or males), p indicates a protected group, and $\Pr(\cdot)$ is the operator for conditional probability. ME is a favored DI measure used by the the US Consumer Financial Protection Bureau (CFPB), the primary agency charged with regulating fair lending laws at the largest US lending institutions and for various other participants in the consumer financial market.[9] Another important DI measure is AIR, more commonly known as a *relative risk ratio* in settings outside of regulatory compliance.

$$\text{AIR} \equiv \frac{\Pr(\hat{\mathbf{y}} = 1 | X_p = 1)}{\Pr(\hat{\mathbf{y}} = 1 | X_c = 1)} \tag{5}$$

AIR is equal to the ratio of the proportion of the protected class that receives a favorable outcome and the proportion of the control class that receives a favorable outcome. Statistically significant AIR values

[8] See: Tutorial: 21 Fairness Definitions and Their Politics.
[9] See: Supervisory Highlights, Issue 9, Fall 2015.

below 0.8 can be considered *prima facia* evidence of discrimination. An additional long-standing and pertinent measure of DI is SMD. SMD is often used to assess disparities in continuous features, such as income differences in employment analyses, or interest rate differences in lending. It originates from work on statistical power, and is more formally known as *Cohen's d*. SMD is equal to the difference in the average protected class outcome, $\bar{\hat{\mathbf{y}}}_p$, minus the control class outcome, $\bar{\hat{\mathbf{y}}}_c$, divided by a measure of the standard deviation of the population, $\sigma_{\hat{\mathbf{y}}}$.[10] Cohen defined values of this measure to have *small*, *medium*, and *large* effect sizes if the values exceeded 0.2, 0.5, and 0.8, respectively.

$$\text{SMD} \equiv \frac{\bar{\hat{\mathbf{y}}}_p - \bar{\hat{\mathbf{y}}}_c}{\sigma_{\hat{\mathbf{y}}}} \tag{6}$$

The numerator in the SMD is roughly equivalent to ME but adds the standard deviation divisor as a standardizing factor. Because of this standardization factor, SMD allows for a comparison across different types of outcomes, such as inequity in mortgage closing fees or inequities in the interest rates given on certain loans. In this, one may apply definitions in Cohen [25] of *small*, *medium*, and *large* effect sizes, which represent a measure of *practical significance*, which is described in detail in Appendix D. Finally, confusion matrix measures in demographic groups, such as accuracy, false positive rate (FPR), false negative rate (FNR), and their ratios, are also considered as measures of DI in Subsection 2.2.3 and Appendix E.2.

1.8. Software Resources

Python code to reproduce discussed results is available at: `https://github.com/h2oai/article-information-2019`. The primary Python packages employed are: `numpy` and `pandas` for data manipulation, `h2o`, `keras`, `shap`, and `tensorflow` for modeling, explanation, and discrimination testing, and typically `matplotlib` for plotting.

2. Results

Results are laid out for the simulated and mortgage datasets. Accuracy is compared for unconstrained, less interpretable g^{GBM} and g^{ANN} models and constrained, more interpretable g^{MGBM} and g^{XNN} models. Then, for the g^{MGBM} and g^{XNN} models, intrinsic interpretability, post-hoc explanation, and discrimination testing results are explored.

2.1. Simulated Data Results

Fit comparisons between unconstrained and constrained models and XNN interpretability results are discussed in Subsections 2.1.1 and 2.1.2. As model training and assessment on the simulated data is a rough validation exercise meant to showcase expected results on data with known characteristics, and given that most of the techniques in the proposed workflow are already used widely or have been validated elsewhere, reporting of simulated data results in the main text will focus mostly on fit measures and the more novel g^{XNN} interpretability results. The bulk of the post-hoc explanation and discrimination testing results for the simulated data are left to Appendix E.

2.1.1. Constrained vs. Unconstrained Model Fit Assessment

Table 1 presents a variety of fit measures for g^{GBM}, g^{MGBM}, g^{ANN}, and g^{XNN} on the simulated test data. g^{XNN} exhibits the best performance, but the models exhibit only a fairly small range of fit results. Interpretability and explainability benefits of the constrained models appear to come at little cost to

[10] There are several measures of the standard deviation of the score that are typically used: **1.** the standard deviation of the population, irrespective of protected class status, **2.** a standard deviation calculated only over the two groups being considered in a particular calculation, or **3.** a pooled standard deviation, using the standard deviations for each of the two groups with weights.

overall model performance, or in the case of g^{ANN} and g^{XNN}, no cost at all. For the displayed measures, g^{MGBM} performs $\sim$2% worse on average than g^{GBM}. g^{XNN} performs $\sim$0.5% better on average than g^{XNN}, and g^{XNN} actually shows slightly better fit than g^{ANN} across all fit measures except specificity. Fit measures that require a probability cutoff are taken at the best F1 threshold for each model.

Table 1. Fit measures for $g^{\text{GBM}}(\mathbf{X})$, $g^{\text{MGBM}}(\mathbf{X})$, $g^{\text{ANN}}(\mathbf{X})$, and $g^{\text{XNN}}(\mathbf{X})$ on the simulated test data. Arrows indicate the direction of improvement for each measure.

Model	Accuracy ↑	AUC ↑	F1 ↑	Logloss ↓	MCC ↑	RMSE ↓	Sensitivity ↑	Specificity ↑
g^{GBM}	0.757	0.847	0.779	0.486	0.525	0.400	0.858	0.657
g^{MGBM}	0.744	0.842	0.771	0.502	0.504	0.407	0.864	0.625
g^{ANN}	0.757	0.850	0.779	0.480	0.525	0.398	0.858	0.657
g^{XNN}	0.758	0.851	0.781	0.479	0.528	0.397	0.867	0.648

2.1.2. Interpretability Results

For g^{XNN}, inherent interpretability manifests as plots of sparse γ_k output layer weights, n_k subnetwork ridge functions, and sparse $\beta_{j,k}$ weights in the bottom projection layer. Figure 1 provides detailed insights into the structure g^{XNN} (also described in Equation 3). 1**a** displays the sparse γ_k weights of the output layer, where only n_k subnetworks with $k \in \{1, 4, 7, 8, 9\}$ are associated with large magnitude weights. The n_k subnetwork ridge functions appear in 1**b** as simplistic but distinctive functional forms. Color-coding between 1**a** and 1**b** visually reinforces the direct feed-forward relationship between the n_k subnetworks and the γ_k weights of the output layer.

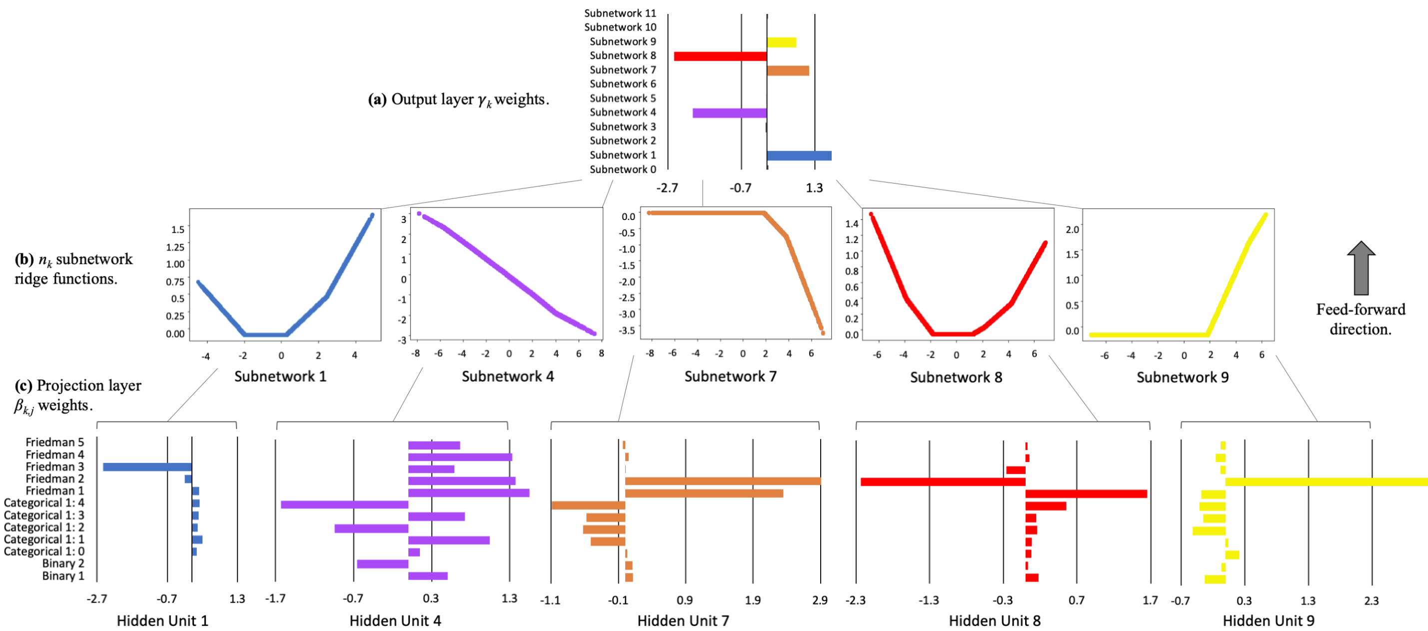

Figure 1. Output layer γ_k weights, corresponding n_k subnetwork ridge functions, and associated projection layer $\beta_{k,j}$ weights for g^{XNN} on the simulated data.

n_k subnetworks are plotted across the output values of their associated $\sum_j \beta_{k,j} x_j$ projection layer hidden units, and color-coding between 1**b** and 1**c** link the $\beta_{j,k}$ weights to their n_k subnetworks. Most of the heavily utilized n_k subnetworks have sparse weights in their $\sum_j \beta_{k,j} x_j$ projection layer hidden units. In particular, subnetwork n_1 appears to be almost solely a function of $X_{\text{Friedman},3}$ and appears to exhibit the expected quadratic behavior for $X_{\text{Friedman},3}$. Subnetworks n_7, n_8, and n_9 appear to be most associated with the globally important $X_{\text{Friedman},1}$ and $X_{\text{Friedman},2}$ features, likely betraying the effort required for g^{XNN} to model the nonlinear sin() function of the $X_{\text{Friedman},1}$ and $X_{\text{Friedman},2}$ product, and these subnetworks, especially n_7 and n_8, appear to display some noticeable sinusoidal characteristics. Subnetwork n_4 seems to be a linear combination of all the original input X_j features, but does weigh the linear $X_{\text{Friedman},4}$ and $X_{\text{Friedman},5}$ terms roughly in the correct two-to-one ratio. As a whole, Figure 1**a**, **b**, and **c** exhibit evidence that g^{XNN} has learned about the signal-generating function in Equation 1 and the displayed information should help practitioners understand which original input X_j features are weighed heavily in each n_k subnetwork, and which n_k subnetworks have a strong influence on $g^{\text{XNN}}(\mathbf{X})$ output. See Appendix B.3 for additional details regarding general XNN architecture.

2.2. Mortgage Data Results

Results for the mortgage data are presented in Subsections 2.2.1 – 2.2.3 to showcase the example workflow. g^{ANN} and g^{XNN} outperform g^{GBM} and g^{MGBM} on the mortgage data, but as in Subsection 2.1.1 the constrained variants of both model architectures do not show large differences in model fit with respect to unconstrained variants. Assuming that in high-stakes applications small fit differences on static test data do not outweigh the need for enhanced model debugging facilitated by high interpretability, only g^{MGBM} and g^{XNN} interpretability, post-hoc explainability, and discrimination testing results are presented.

2.2.1. Constrained vs. Unconstrained Model Fit Assessment

Table 2 shows that g^{ANN} and g^{XNN} noticeably outperform g^{GBM} and g^{MGBM} on the mortgage data for most of the fit measures. This is at least partially due to the preprocessing required to present directly comparable post-hoc explainability results and to use neural networks and tensorflow, e.g., numerical encoding of categorical features and missing values. This preprocessing appears to hamstring some of the tree-based models' inherent capabilities. g^{GBM} models trained on non-encoded data with missing values repeatedly produced area under the curve (AUC) values of $\sim$0.81 (not shown, but available in resources discussed in Subsection 1.8).

Table 2. Fit measures for $g^{\text{GBM}}(\mathbf{X})$, $g^{\text{MGBM}}(\mathbf{X})$, $g^{\text{ANN}}(\mathbf{X})$, and $g^{\text{XNN}}(\mathbf{X})$ on the mortgage test data. Arrows indicate the direction of improvement for each measure.

Model	Accuracy ↑	AUC ↑	F1 ↑	Logloss ↓	MCC ↑	RMSE ↓	Sensitivity ↑	Specificity ↑
g^{GBM}	0.795	0.828	0.376	0.252	0.314	0.276	0.634	0.813
g^{MGBM}	0.765	0.814	0.362	0.259	0.305	0.278	0.684	0.773
g^{ANN}	0.865	0.871	0.474	0.231	0.418	0.262	0.624	0.891
g^{XNN}	0.869	0.868	0.468	0.233	0.409	0.263	0.594	0.898

Regardless of the fit differences between the two families of models, the difference between the constrained and unconstrained variants within the two types of models is small for the GBMs and smaller for the ANNs, $\sim$3.5% and $\sim$1% worse fit respectively, averaged across the measures in Table 2.

2.2.2. Interpretability and Post-hoc Explanation Results

For $g^{\text{MGBM}}(\mathbf{X})$, intrinsic interpretability is evaluated with PD and ICE plots of mostly monotonic prediction behavior for several important X_j, and post-hoc Shapley explanation analysis is used to create global and local feature importance. Global Shapley feature importance for $g^{\text{MGBM}}(\mathbf{X})$ on the mortgage test data is reported in Figure 2. g^{MGBM} places high importance on LTV ratio, perhaps too high, and also weighs DTI ratio, property value, loan amount, and introductory rate period heavily in many of its predictions. Tree SHAP values are reported in the margin space, prior to the application of the logit link function, and the reported numeric values can be interpreted as the mean absolute impact of each X_j on $g^{\text{MGBM}}(\mathbf{X})$ in the mortgage test data in the $g^{\text{MGBM}}(\mathbf{X})$ margin space. The potential over-emphasis of LTV ratio, and the de-emphasis of income, likely an important feature from a business perspective, and the de-emphasis of the encoded no introductory rate period flag feature may also contribute to the decreased performance of $g^{\text{MGBM}}(\mathbf{X})$ as compared to $g^{\text{XNN}}(\mathbf{X})$.

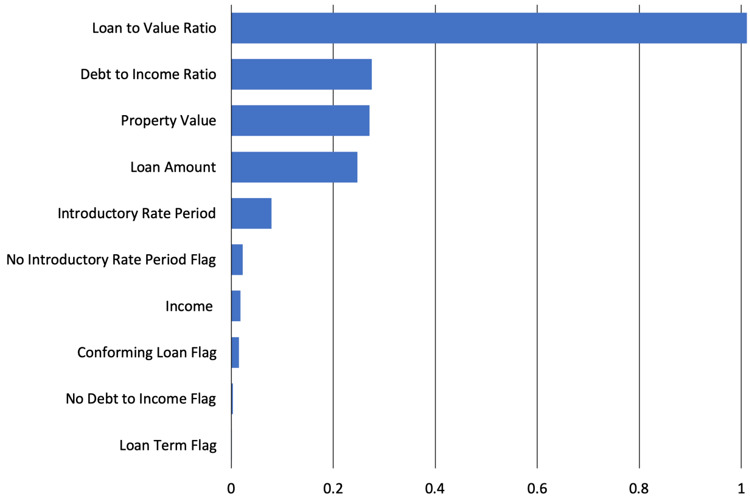

Figure 2. Global mean absolute Tree SHAP feature importance for $g^{\text{MGBM}}(\mathbf{X})$ on the mortgage test data.

Domain knowledge was used to positively constrain DTI ratio and LTV ratio and to negatively constrain income and the loan term flag under g^{MGBM}. The monotonicity constraints for DTI ratio and LTV ratio are confirmed for $g^{\text{MGBM}}(\mathbf{X})$ on the mortgage test data in Figure 3. Both DTI ratio and LTV ratio display positive monotonic behavior at all selected percentiles of $g^{\text{MGBM}}(\mathbf{X})$ for ICE and on average with PD. Because PD curves generally follow the patterns of the ICE curves for both features, it's also likely that no strong interactions are at play for DTI ratio and LTV ratio under g^{MGBM}. Of course, the monotonicity constraints themselves can dampen the effects of non-monotonic interactions under g^{MGBM}, even if they do exist in the training data (e.g., LTV ratio and the no introductory rate period flag, see Figure 6). This rigidity could also play a role in the performance differences between $g^{\text{MGBM}}(\mathbf{X})$ and $g^{\text{XNN}}(\mathbf{X})$ in the mortgage data not observed for the simulated data, wherein strong interactions appear to be between features with the same monotonicity constraints (e.g., $X_{\text{Friedman},1}$ and $X_{\text{Friedman},2}$, see Figure 1).

PD and ICE are displayed with a histogram to highlight any sparse regions in an input feature's domain. Because most ML models will always issue a prediction on any instance with a correct schema, it's crucial to consider whether a given model learned enough about an instance to make an accurate prediction. Viewing PD and ICE along with a histogram is a convenient method to visually assess whether a prediction is reasonable and based on sufficient training data. DTI ratio and LTV ratio do appear to have sparse regions in their univariate distributions. The monotonicity constraints likely play to the advantage of g^{MGBM} in this regard, as $g^{\text{MGBM}}(\mathbf{X})$ appears to carry reasonable predictions learned from dense domains into the sparse domains of both features.

Figure 3 also displays PD and ICE for the unconstrained feature property value. Unlike DTI ratio and LTV ratio, PD for property value does not always follow the patterns established by ICE curves. While PD shows monotonically increasing prediction behavior on average, apparently influenced by large predictions at extreme $g^{\text{MGBM}}(\mathbf{X})$ percentiles, ICE curves for individuals at the 40$^{\text{th}}$ percentile of $g^{\text{MGBM}}(\mathbf{X})$, and lower, exhibit different prediction behavior with respect to property value. Some individuals at these lower percentiles display monotonically decreasing prediction behavior while others appear to show fluctuating prediction behavior. Property value is strongly right-skewed, with little data regarding high-value property from which g^{MGBM} can learn. For the most part, reasonable predictions do appear to be carried from more densely populated regions to more sparsely populated regions. However, prediction fluctuations at lower $g^{\text{MGBM}}(\mathbf{X})$ percentiles are visible, and appear in a sparse region of property value. This divergence of PD and ICE can be indicative of an interaction affecting property value under g^{MGBM} [22], and analysis by surrogate decision tree did show evidence

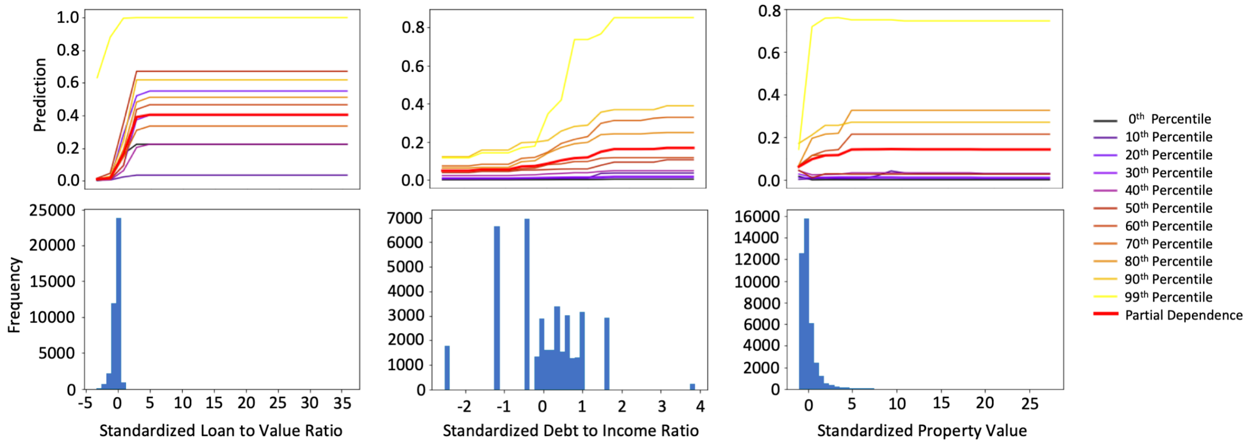

Figure 3. PD, ICE for 10 individuals across selected percentiles of $g^{\text{MGBM}}(\mathbf{X})$, and histograms for the three most important input features of g^{MGBM} on the mortgage test data.

of numerous potential interactions in lower predictions ranges of $g^{\text{MGBM}}(\mathbf{X})$ [31] (not shown, but available in resources discussed in Subsection 1.8). Fluctuations in ICE could also be caused by overfitting or by leakage of strong non-monotonic signal from important constrained features into the modeled behavior of non-constrained features.

In Figure 4, local Tree SHAP values are displayed for selected individuals at the 10$^{\text{th}}$, 50$^{\text{th}}$, and 90$^{\text{th}}$ percentiles of $g^{\text{MGBM}}(\mathbf{X})$ in the mortgage test data. Each Shapley value in Figure 4 represents the difference in $g^{\text{MGBM}}(\mathbf{x})$ and the average of $g^{\text{MGBM}}(\mathbf{X})$ associated with this instance of some input feature x_j [32]. Accordingly, the logit of the sum of the Shapley values and the average of $g^{\text{MGBM}}(\mathbf{X})$ is $g^{\text{MGBM}}(\mathbf{x})$, the prediction in the probability space for any $\mathbf{x}$.

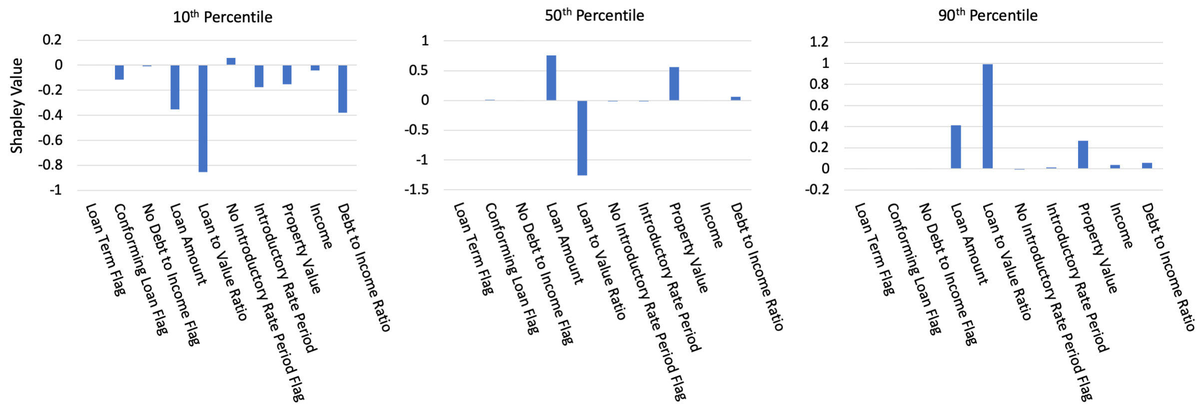

Figure 4. Tree SHAP values for three individuals across selected percentiles of $g^{\text{MGBM}}(\mathbf{X})$ for the mortgage test data.

The selected individuals show an expected progression of mostly negative Shapley values at the 10$^{\text{th}}$ percentile, a mixture of positive and negative Shapley values at the 50$^{\text{th}}$ percentile, mostly positive Shapley values the 90$^{\text{th}}$ percentile, and with globally important features driving most local model decisions. Deeper significance for Figure 4 lies in the ability of Tree SHAP to accurately and consistently summarize any single $g^{\text{MGBM}}(\mathbf{x})$ prediction in this manner, which is generally important for enabling logical appeal or override of ML-based decisions, and is specifically important in the context of lending, where applicable regulations often require lenders to provide consumer-specific reasons for denying credit to an individual. In the US, applicable regulations are typically ECOA and FCRA, and the consumer-specific reasons are commonly known as *adverse actions codes*.

Figure 5 displays global feature importance for $g^{\text{XNN}}(\mathbf{X})$ on the mortgage test data. Deep SHAP values are reported in the probability space, after the application of the logit link function. They are also calculated from the projection layer of g^{XNN}. Thus, the Deep SHAP values in Figure 5 are the estimated average absolute impact of each input, X_j, in the projection layer and probability space of $g^{\text{XNN}}(\mathbf{X})$ for the mortgage test data. g^{XNN} distributes importance more evenly across business drivers and puts stronger emphasis on the no introductory rate period flag feature than does g^{MGBM}. Like g^{MGBM}, g^{XNN} puts little emphasis on the other flag features.

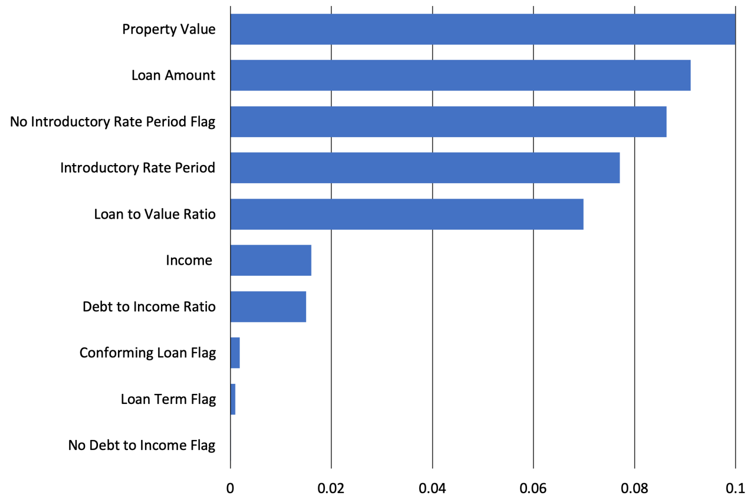

Figure 5. Global mean absolute Deep SHAP feature importance for $g^{\text{XNN}}(\mathbf{X})$ on the mortgage test data.

As compared to g^{MGBM}, g^{XNN} assigns higher importance to property value, loan amount, and income, and lower importance on LTV ratio and DTI ratio.

The capability of g^{XNN} to model nonlinear phenomenon and high-degree interactions, and to do so in an interpretable manner, is on display in Figure 6. 6**a** presents the sparse γ_k weights of the g^{XNN} output layer in which the n_k subnetworks with $k \in \{0, 1, 2, 3, 5, 8, 9\}$ have large magnitude weights and n_k subnetworks, $k \in \{4, 6, 7\}$, have small or near-zero magnitude weights. Distinctive ridge functions that feed into those large magnitude γ_k weights are highlighted in 6**b** and color-coded to pair with their corresponding γ_k weight. As in the Subsection 2.1.2, n_k ridge function plots vary with the output of the corresponding projection layer $\sum_j \beta_{k,j} x_j$ hidden unit, with weights displayed in matching colors in 6**c**. In both the simulated and mortgage data, n_k ridge functions appear to be elementary functional forms that the output layer learns to combine to generate accurate predictions, reminiscent of basis functions for the modeled space. 6**c** displays the sparse $\beta_{j,k}$ weights of the projection layer $\sum_j \beta_{k,j} x_j$ hidden units that are associated with each n_k subnetwork ridge function. For instance, subnetwork n_3 is influenced by large weights for LTV ratio, no introductory rate period flag, and introductory rate period, whereas subnetwork n_9 is nearly completely dominated by the weight for income. See Appendix B.3 for details regarding general XNN architecture.

To compliment the global interpretability of g^{XNN}, Figure 7 displays local Shapley values for selected individuals, estimated from the projection layer using Deep SHAP in the g^{XNN} probability space. Similar to Tree SHAP, local Deep SHAP values should sum to $g^{\text{XNN}}(\mathbf{x})$. While the Shapley values appear to follow the roughly increasing pattern established in Figures A4, A6, and 4, their true value is their ability to be calculated for any $g^{\text{XNN}}(\mathbf{x})$ prediction, as a means to summarize model reasoning and allow for appeal and override of specific ML-based decisions.

2.2.3. Discrimination Testing Results

Tables 3**a** and 3**b** show the results of the discrimination tests using the mortgage data for two sets of class-control groups: blacks as compared to whites, and females as compared to males. As with the simulated data in Table A1, several measures of disparities are shown, with the SMDs calculated using the probabilities from $g^{\text{MGBM}}(\mathbf{X})$ and $g^{\text{XNN}}(\mathbf{X})$, and the accuracy, FPRs, and FPR ratios, MEs, and AIRs calculated using a binary outcome based on a cutoff of 0.20 (anyone with probabilities of

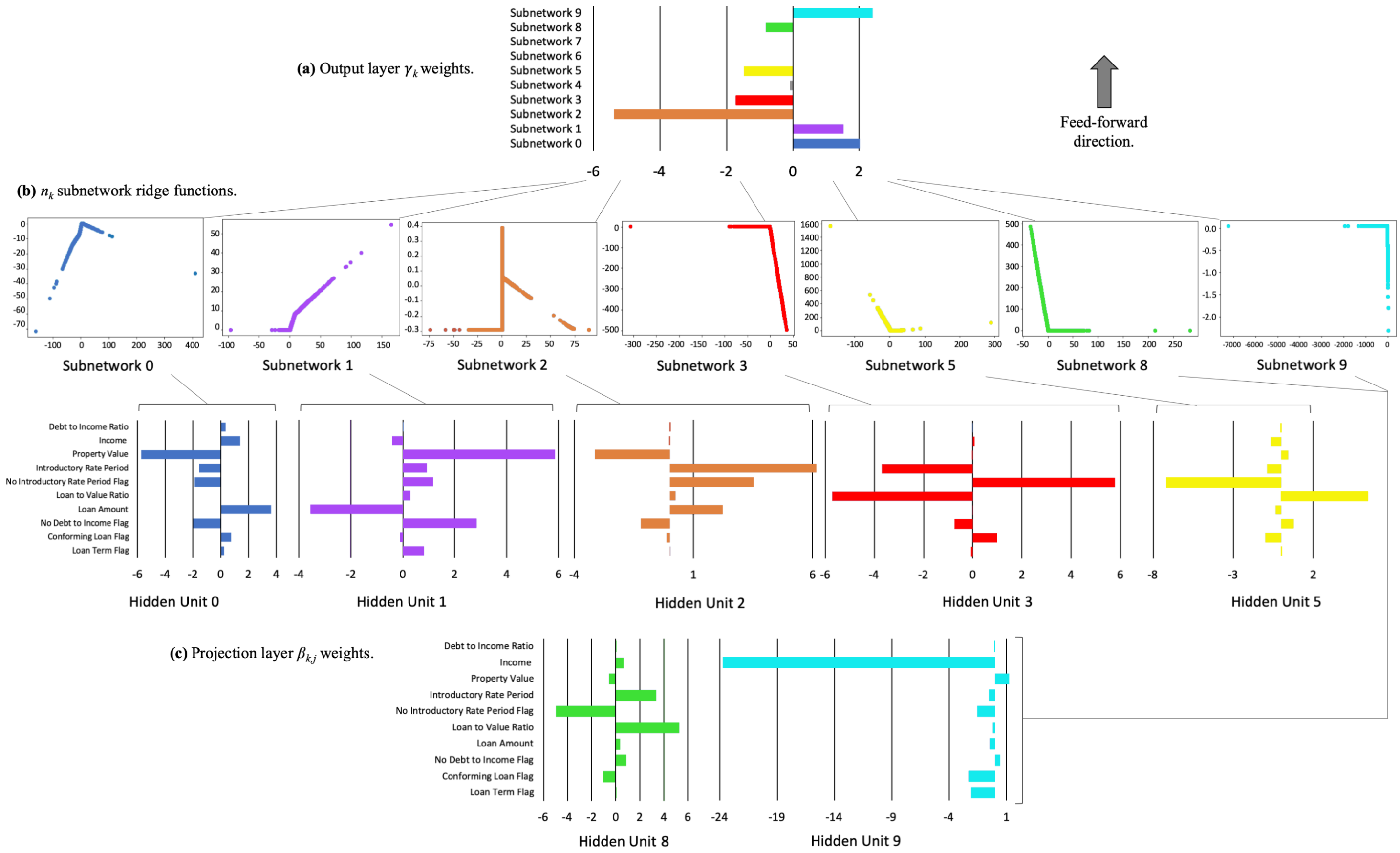

Figure 6. Output layer γ_k weights, corresponding n_k ridge functions, and associated projection layer $\beta_{k,j}$ weights for g^{XNN} on the mortgage data.

0.2 or less receives the favorable outcome).[11] Since g^{MGBM} and g^{XNN} are predicting the likelihood of receiving a high-priced loan, g^{MGBM} and g^{XNN} assume that a lower score is favorable. Thus, one might consider FPR ratios as a measure of the class-control disparities. FPR ratios are higher under g^{XNN} than g^{MGBM} (2.45 vs. 2.10) in Table 3**b**, but overall FPRs are lower for blacks under g^{XNN} (0.295 vs. 0.315) in Table 3**a**. This is the same pattern seen in the simulated data results in Appendix E.2, leading to the question of whether a fairness goal should not only consider class-control relative rates, but also intra-class improvements in the chosen fairness measure. Similar results are found for the female-male comparison, but the relative rates are less stark: 1.15 for $g^{\text{MGBM}}(\mathbf{X})$ and 1.21 for $g^{\text{XNN}}(\mathbf{X})$.

Both ME and AIR show higher disparities for blacks under g^{XNN} than g^{MGBM}. Blacks receive high-priced loans 21.4% more frequently using g^{XNN} vs. 18.3% for g^{MGBM}. Both g^{MGBM} and g^{XNN} show AIRs that are statistically significantly below parity (not shown, but available in resources discussed in Subsection 1.8), and which are also below the EEOC's 0.80 threshold. This would typically indicate need for further review to determine the cause and validity of these disparities, and a few relevant remediation techniques for such discovered discrimination are discussed in Subsection 3.3. On the other hand, women improve under g^{XNN} vs. g^{MGBM} (MEs of 3.6% vs. 4.1%; AIRs of 0.955 vs. 0.948). The AIRs, while statistically significantly below parity, are well above the EEOC's threshold of 0.80. In most situations, the values of these measures alone would not likely flag a model for further review. Black SMDs for $g^{\text{XNN}}(\mathbf{X})$ and $g^{\text{MGBM}}(\mathbf{X})$ are similar: 0.621 and 0.628, respectively. These exceed Cohen's guidelines of 0.5 for a medium effect size and would likely trigger further review.

---

[11] See Appendix F for comments pertaining to discrimination testing and cutoff selection.

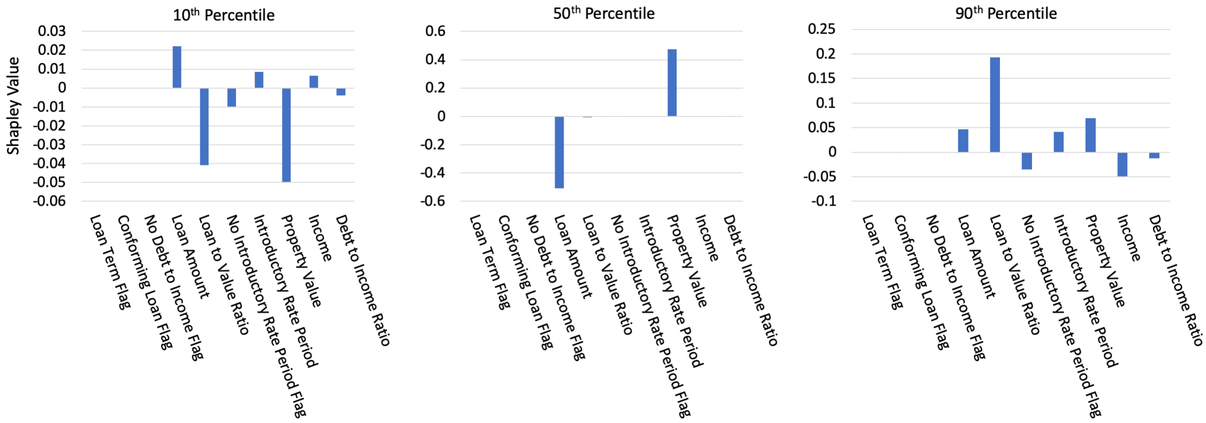

Figure 7. Deep SHAP values for three individuals across selected percentiles of $g^{\text{XNN}}(\mathbf{X})$ on the mortgage test data.

(a) Group size, accuracy, and FPR for $g^{\text{MGBM}}(\mathbf{X})$ and $g^{\text{XNN}}(\mathbf{X})$ on the mortgage test data.

Class	N	Model	Accuracy↑	FPR↓
Black	2,608	g^{MGBM}	0.654	0.315
		g^{XNN}	0.702	0.295
White	28,361	g^{MGBM}	0.817	0.150
		g^{XNN}	0.857	0.120
Female	8,301	g^{MGBM}	0.768	0.208
		g^{XNN}	0.822	0.158
Male	13,166	g^{MGBM}	0.785	0.182
		g^{XNN}	0.847	0.131

(b) AIR, ME, SMD, and FPR ratio for $g^{\text{MGBM}}(\mathbf{X})$ and $g^{\text{XNN}}(\mathbf{X})$ on the mortgage test data.

Model	Protected Class	Control Class	AIR↑	ME↓	SMD↓	FPR Ratio↓
g^{MGBM}	Black	White	0.776	18.3%	0.628	2.10
	Female	Male	0.948	4.1%	0.084	1.15
g^{XNN}	Black	White	0.743	21.4%	0.621	2.45
	Female	Male	0.955	3.6%	0.105	1.21

Table 3. Discrimination measures for the mortgage test data. Arrows indicate the direction of improvement for each measure.

Female SMDs are well below Cohen's definition of small effect size: 0.105 and 0.084 for $g^{\text{XNN}}(\mathbf{X})$ and $g^{\text{MGBM}}(\mathbf{X})$, respectively. Similar to results for female AIR, these values alone are unlikely to prompt further review.

3. Discussion

3.1. The Burgeoning Python Ecosystem for Responsible Machine Learning

Figure 8 displays a holistic approach to ML model training, assessment, and deployment meant to decrease discrimination, inaccuracy, privacy, and security risks for high-stakes, human-centered, or regulated ML applications.[12] While all the methods mentioned in Figure 8 play an important role in increasing human trust and understanding of ML, a few pertinent references and Python resources are highlighted below as further reading to augment this this text's focus on certain interpretable models, post-hoc explanation, and discrimination testing techniques.

[12] See: Toward Responsible Machine Learning for details regarding Figure 8.

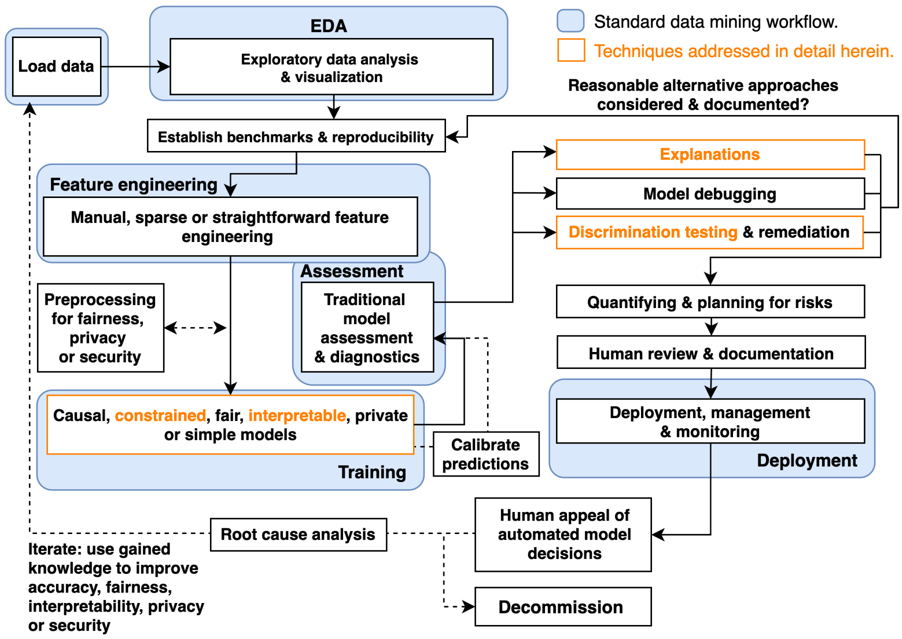

Figure 8. An example responsible ML workflow in which interpretable models, post-hoc explanations, discrimination testing and remediation techniques, among several other processes, can create an understandable and trustworthy ML system for high-stakes, human-centered, or regulated applications.

Any discussion of interpretable ML models would be incomplete without references to the seminal work of the Rudin group at Duke University and EBMs, or GA^{2}Ms, pioneered by researchers at Microsoft and Cornell. In keeping with a major theme of this manuscript, models from these leading researchers and several other kinds of interpretable ML models are now available as open source Python packages. Among several types of currently available interpretable models, practitioners can now use Python to evaluate EBM in the `interpret` package, optimal sparse decision trees, GAMs in the `pyGAM` package, a variant of Friedman's RuleFit in the `skope-rules` package, monotonic calibrated interpolated lookup tables in `tensorflow/lattice`, and *this looks like that* interpretable deep learning [33], [34], [35], [36].[13,14] Additional, relevant references and Python functionality include:

- **Exploratory data analysis (EDA)**: `H2OAggregatorEstimator` in `h2o` [37].
- **Sparse feature extraction**: `H2OGeneralizedLowRankEstimator` in `h2o` [38].
- **Preprocessing & models for privacy**: `diffprivlib`, `tensorflow/privacy` [39], [40], [41], [42].
- **Causal inference & probabilistic programming**: dowhy, PyMC3 [43].
- **Post-hoc explanation**: structured data explanations with `alibi` and `PDPbox`, image classification explanations with `DeepExplain`, and natural language explanations with `allennlp` [44], [45], [46].
- **Discrimination testing**: `aequitas`, `Themis`.
- **Discrimination remediation**: Reweighing, adversarial de-biasing, learning fair representations, and reject option classification with `AIF360` [47], [48], [49], [50].
- **Model debugging**: `foolbox`, `SALib`, `tensorflow/cleverhans`, and `tensorflow/model-analysis` [51], [52], [53], [54].
- **Model documentation**: models cards [55], e.g., GPT-2 model card, Object Detection model card.

See Awesome Machine Learning Interpretability for a longer, community-curated metalist of related software packages and resources.

---

13 See: Optimal sparse decision trees.
14 See: *This looks like that* interpretable deep learning.

3.2. Appeal and Override of Automated Decisions

Interpretable models and post-hoc explanations can play an important role in increasing transparency into model mechanisms and predictions. As seen in Section 2, interpretable models often enable users to enforce domain knowledge-based constraints on model behavior, to ensure that models obey reasonable expectations, and to gain data-derived insights into the modeled problem domain. Post-hoc explanations generally help describe and summarize mechanisms and decisions, potentially yielding an even clearer understanding of ML models. Together they can allow for human learning from ML, certain types of regulatory compliance, and crucially, human appeal or override of automated model decisions [31]. Interpretable models and post-hoc explanations are likely good candidates for ML uses cases under the FCRA, ECOA, GDPR and other regulations that may require explanations of model decisions, and they are already used in the financial services industry today for model validation and other purposes.[15,16] Writ large, transparency in ML also facilitates additional responsible AI processes such as model debugging, model documentation, and logical appeal and override processes, some of which may also be required by applicable regulations.[17] Among these, providing persons affected by a model with the opportunity to appeal ML-based decisions may deserve the most attention. ML models are often wrong[18] and appealing black-box decisions can be difficult.[2] For high-stakes, human-centered, or regulated applications that are trusted with mission- or life-critical decisions, the ability to logically appeal or override inevitable wrong decisions is not only a possible prerequisite for compliance, but also a failsafe procedure for those affected by ML decisions.

3.3. Discrimination Testing and Remediation in Practice

A significant body of research has emerged around exploring and fixing ML discrimination [57]. Methods can be broadly placed into two groups: more traditional methods that mitigate discrimination by searching across possible algorithmic and feature specifications, and many approaches that have been developed in the last 5–7 years that alter the training algorithm, preprocess training data, or post-process predictions in order to diminish class-control correlations or dependencies. Whether these more recent methods are suitable for a particular use case depends on the legal environment where a model is deployed and on the use case itself. For comments on why some recent techniques could result in regulatory non-compliance in certain scenarios, see Appendix G.

Of the newer class of fairness enhancing interventions, within-algorithm discrimination mitigation techniques that do not use class information may be more likely to be acceptable in highly regulated settings today. These techniques often incorporate a loss function where more discriminatory paths or weights are penalized and only used by the model if improvements in fit overcome some penalty. (The relative level of fit-to-discrimination penalty is usually determined via hyperparameter.) Other mitigation strategies that only alter hyperparameters or algorithm choice are also likely to be acceptable. Traditional feature selection techniques (e.g., those used in linear models and decision trees) are also likely to continue to be accepted in regulatory environments. For further discussion of techniques that can mitigate DI in US financial services, see Schmidt and Stephens [58].

Regardless of the methodology chosen to minimize disparities, advances in computing have enhanced the ability to search for less discriminatory models. Prior to these advances, only a small number of alternative algorithms could be tested for lower levels of disparity without causing infeasible

---

[15] See: *Deep Insights into Explainability and Interpretability of Machine Learning Algorithms and Applications to Risk Management*.

[16] Unfortunately, many non-consistent explanation methods can result in drastically different global and local feature importance values across different models trained on the same data or even for refreshing a model with augmented training data [32]. Consistency and accuracy guarantees are perhaps a factor in the growing momentum behind Shapley values as a candidate technique for generating consumer-specific adverse action notices for explaining and appealing automated ML-based decisions in highly-regulated settings, such as credit lending [56].

[17] E.g.: US Federal Reserve Bank SR 11-7: Guidance on Model Risk Management.

[18] "All models are wrong, but some are useful." – George Box, Statistician (1919 - 2013)

delays in model implementation. Now, large numbers of models can be quickly tested for lower discrimination and better predictive quality. An additional opportunity arises as a result of ML itself: the well-known Rashomon effect, or the multiplicity of good ML models for most datasets. It is now feasible to train more models, find more good models, and test more models for discrimination, and among all those tested models, there are likely to be some with high predictive performance and low discrimination.

3.4. Intersectional and Non-static Risks in Machine Learning

The often black-box nature of ML, the perpetuation or exacerbation of discrimination by ML, or the privacy harms and security vulnerabilities inherent in ML are each serious and difficult problems on their own. However, evidence is mounting that these harms can also manifest as complex intersectional challenges, e.g., the *fairwashing* or *scaffolding* of biased models with ML explanations, the privacy harms of ML explanations, or the adversarial poisoning of ML models to become discriminatory [8], [18], [19].[19,20,21] Practitioners should of course consider the discussed interpretable modeling, post-hoc explanation, and discrimination testing approaches as at least partial remedies to the black-box and discrimination issues in ML. However, they should also consider that explanations can ease model stealing, data extraction, and membership inference attacks, and that explanations can mask ML discrimination. Additionally, high-stakes, human-centered, or regulated ML systems should generally be built and tested with robustness to adversarial attacks as a primary design consideration, and specifically to prevent ML models from being poisoned or otherwise altered to become discriminatory. Accuracy, discrimination, and security characteristics of a system can change over time as well. Simply testing for these problems at training time, as presented in Section 2, is not adequate for high-stakes, human-centered, or regulated ML systems. Accuracy, discrimination, and security should be monitored in real-time and over time, as long as a model is deployed.

4. Conclusion

This text puts forward results on simulated data to provide some validation of constrained ML models, post-hoc explanation techniques, and discrimination testing methods. These same modeling, explanation, and discrimination testing approaches are then applied to more realistic mortgage data to provide an example of a responsible ML workflow for high-stakes, human-centered, or regulated ML applications. The discussed methodologies are solid steps toward interpretability, explanation, and minimal discrimination for ML decisions, which should ultimately enable increased fairness and logical appeal processes for ML decision subjects. Of course, there is more to the responsible practice of ML than interpretable models, post-hoc explanation, and discrimination testing, even from a technology perspective, and Section 3 also points out numerous additional references and open source Python software assets that are available to researchers and practitioners today to increase human trust and understanding in ML systems. While the complex and messy problems of racism, sexism, privacy violations, and cyber crime can probably never be solved by technology alone, this work (and many others) illustrate numerous ways for ML practitioners to mitigate such risks.

Author Contributions: NG, data cleaning, GBM and MGBM assessment and results; PH, primary author; KM, ANN and XNN implementation, assessment, and results; NS, secondary author, data simulation and collection, and discrimination testing.

Funding: This work received no external funding.

---

[19] See: *Tay, Microsoft's AI chatbot, gets a crash course in racism from Twitter*.

[20] While the focus of this paper is not ML security, proposed best-practices from that field do point to transparency of ML systems as a mitigating factor for some ML attacks and hacks [54]. High system complexity is sometimes considered a mitigating influence as well [59]. This is sometimes known as the *transparency paradox* in data privacy and security, and it likely applies to ML security as well, especially in the context of interpretable ML models and post-hoc explanation.

[21] See: *The AI Transparency Paradox*.

Acknowledgments: Wen Phan for work in formalizing notation. Sue Shay for editing. Andrew Burt for ideas around the transparency paradox.

Conflicts of Interest: XNN was first made public by the corporate model validation team at Wells Fargo bank. Wells Fargo is a customer of, and investor in, H2O.ai and a client of BLDS, LLC. However, communications regarding XNN between Wells Fargo and the authors have been extremely limited prior to and during the drafting of this manuscript. Moreover, Wells Fargo exerted absolutely no editorial control over the text or results herein.

Abbreviations

The following abbreviations are used in this text: AI – artificial intelligence, AIR - adverse impact ratio, ALE - accumulated local effect, ANN – artificial neural network, APR – annual percentage rate, AUC – area under the curve, CFPB – Consumer Financial Protection Bureau, DI – disparate impact, DT – disparate treatment, DTI – debt to income, EBM or GA^2M – explainable boosting machine, i.e. variants GAMs that consider two-way interactions and may incorporate boosting into training, EEOC – Equal Employment Opportunity Commission, ECOA - Equal Credit Opportunity Act, EDA – exploratory data analysis, EU – European Union, FCRA – Fair Credit Reporting Act, FNR – false negative rate, FPR – false positive rate, GAM – generalized additive model, GBM – gradient boosting machine, GDPR - General Data Protection Regulation, HMDA – Home Mortgage Disclosure Act, ICE – individual conditional expectation, LTV – loan to value, MCC – Matthews correlation coefficient, ME – marginal effect, MGBM – monotonic gradient boosting machine, ML – machine learning, PD – partial dependence, RMSE – root mean square error, SGD – stochastic gradient descent, SHAP – SHapley Additive exPlanation, SMD - standardized mean difference, SR – supervision and regulation, US – United States, XNN – explainable neural network.

Appendix A. Mortgage Data Details

The US HMDA law, originally enacted in 1975, requires many financial institutions that originate mortgage products to provide certain loan-level data about many of the types of mortgage-related products on an annual basis. This information is first provided to the CFPB, which subsequently releases some of the data to the public. Regulators often use HMDA data to, "...show whether lenders are serving the housing needs of their communities; they give public officials information that helps them make decisions and policies; and they shed light on lending patterns that could be discriminatory."[5] In addition to regulatory use, public advocacy groups use these data for similar purposes, and the lenders themselves use the data to benchmark their community outreach relative to their peers. The publicly available data that the CFPB releases includes information such as the lender, the type of loan, loan amount, LTV ratio, DTI ratio, and other important financial descriptors. The data also include information on each borrower and co-borrower's race, ethnicity, gender, and age. Because the data includes information on these protected class characteristics, certain measures that can be indicative of discrimination in lending can be calculated directly using the HDMA data. Ultimately, the HMDA data represent the most comprehensive source of data on highly-regulated mortgage lending that is publicly available, which makes it an ideal dataset to use for the types of analyses set forth in Sections 1 and 2.

Appendix B. Selected Algorithmic Details

Appendix B.1. Notation

To facilitate descriptions of data and modeling, explanatory, and discrimination testing techniques, notation for input and output spaces, datasets, and models is defined.

Appendix B.1.1. Spaces

- Input features come from the set $\mathcal{X}$ contained in a P-dimensional input space, $\mathcal{X} \subset \mathbb{R}^P$. An arbitrary, potentially unobserved, or future instance of $\mathcal{X}$ is denoted $\mathbf{x}$, $\mathbf{x} \in \mathcal{X}$.
- Labels corresponding to instances of $\mathcal{X}$ come from the set $\mathcal{Y}$.

- Learned output responses of models are contained in the set $\hat{\mathcal{Y}}$.

Appendix B.1.2. Data

- The input dataset $\mathbf{X}$ is composed of observed instances of the set $\mathcal{X}$ with a corresponding dataset of labels $\mathbf{Y}$, observed instances of the set $\mathcal{Y}$.
- Each i-th observed instance of $\mathbf{X}$ is denoted as $\mathbf{x}^{(i)} = [x_0^{(i)}, x_1^{(i)}, \ldots, x_{P-1}^{(i)}]$, with corresponding i-th labels in $\mathbf{Y}$, $\mathbf{y}^{(i)}$, and corresponding predictions in $\hat{\mathbf{Y}}$, $\hat{\mathbf{y}}^{(i)}$.
- $\mathbf{X}$ and $\mathbf{Y}$ consist of N tuples of observed instances: $[(\mathbf{x}^{(0)}, \mathbf{y}^{(0)}), (\mathbf{x}^{(1)}, \mathbf{y}^{(1)}), \ldots, (\mathbf{x}^{(N-1)}, \mathbf{y}^{(N-1)})]$.
- Each j-th input column vector of $\mathbf{X}$ is denoted as $X_j = [x_j^{(0)}, x_j^{(1)}, \ldots, x_j^{(N-1)}]^T$.

Appendix B.1.3. Models

- A type of ML model g, selected from a hypothesis set $\mathcal{H}$, is trained to represent an unknown signal-generating function f observed as $\mathbf{X}$ with labels $\mathbf{Y}$ using a training algorithm $\mathcal{A}$: $\mathbf{X}, \mathbf{Y} \xrightarrow{\mathcal{A}} g$, such that $g \approx f$.
- g generates learned output responses on the input dataset $g(\mathbf{X}) = \hat{\mathbf{Y}}$, and on the general input space $g(\mathcal{X}) = \hat{\mathcal{Y}}$.
- The model to be explained and tested for discrimination is denoted as g.

Appendix B.2. Monotonic Gradient Boosting Machine Details

As in unconstrained GBM, Θ_b^{mono} is selected in a greedy, additive fashion by minimizing a regularized loss function that considers known target labels, $\mathbf{y}$, the predictions of all subsequently trained trees in the in g^{MGBM}, $g_{b-1}^{\text{MGBM}}(\mathbf{X})$, and the b-th tree splits, $T_b(\mathbf{x}^{(i)}; \Theta_b^{\text{mono}})$, in a numeric error function (e.g., squared error, Huber error), l, and a regularization term that penalizes complexity in the current tree, $\Omega(T_b)$. For the b-th iteration, the loss function, $\mathcal{L}_b$, can generally be defined as:

$$\mathcal{L}_b = \sum_{i=0}^{N-1} l(y^{(i)}, g_{b-1}^{\text{MGBM}}(\mathbf{x}^{(i)}), T_b(\mathbf{x}^{(i)}; \Theta_b^{\text{mono}})) + \Omega(T_b) \tag{A1}$$

In addition to $\mathcal{L}_b$, g^{MGBM} training is characterized by additional splitting rules and constraints on tree node weights. Each binary splitting rule, $\theta_{b,j,k} \in \Theta_b$, is associated with a feature, X_j, is the k-th split associated with X_j in T_b, and results in left and right child nodes with a numeric weights, $\{w_{b,j,k,L}, w_{b,j,k,R}\}$. For terminal nodes, $\{w_{b,j,k,L}, w_{b,j,k,R}\}$ can be direct numeric components of some g^{MGBM} prediction. For two values of some feature X_j, $x_j^\alpha \leq x_j^\beta$, g^{MGBM} is positive monotonic with respect to some X_j if $g^{\text{MGBM}}(x_j^\alpha) \leq g^{\text{MGBM}}(x_j^\beta)\ \forall\ x_j^\alpha \leq x_j^\beta \in X_j$. The following rules and constraints ensure positive monotonicity in Θ_b, where the prediction for each value results in $T_b(x_j^\alpha; \Theta_b) = w_\alpha$ and $T_b(x_j^\beta; \Theta_b) = w_\beta$.

1. For the first and highest split in T_b involving X_j, any $\theta_{b,j,0}$ resulting in $T(x_j; \theta_{b,j,0}) = \{w_{b,j,0,L}, w_{b,j,0,R}\}$ where $w_{b,j,0,L} > w_{b,j,0,R}$, is not considered.
2. For any subsequent left child node involving X_j, any $\theta_{b,j,k\geq 1}$ resulting in $T(x_j; \theta_{b,j,k\geq 1}) = \{w_{b,j,k\geq 1,L}, w_{b,j,k\geq 1,R}\}$ where $w_{b,j,k\geq 1,L} > w_{b,j,k\geq 1,R}$, is not considered.
3. Moreover, for any subsequent left child node involving X_j, $T(x_j; \theta_{b,j,k\geq 1}) = \{w_{b,j,k\geq 1,L}, w_{b,j,k\geq 1,R}\}$, $\{w_{b,j,k\geq 1,L}, w_{b,j,k\geq 1,R}\}$ are bound by the associated $\theta_{b,j,k-1}$ set of node weights, $\{w_{b,j,k-1,L}, w_{b,j,k-1,R}\}$, such that $\{w_{b,j,k\geq 1,L}, w_{b,j,k\geq 1,R}\} < \frac{w_{b,j,k-1,L} + w_{b,j,k-1,R}}{2}$.
4. (1) and (2) are also applied to all right child nodes, except that for right child nodes $w_{b,j,k,L} \leq w_{b,j,k,R}$ and $\{w_{b,j,k\geq 1,L}, w_{b,j,k\geq 1,R}\} \geq \frac{w_{b,j,k-1,L} + w_{b,j,k-1,R}}{2}$.

Note that for any one X_j and $T_b \in g^{\text{MGBM}}$ left subtrees will alway produce lower predictions than right subtrees, and that any $g^{\text{MGBM}}(\mathbf{x})$ is an addition of each T_b output, with the application of a monotonic logit or softmax link function for classification problems. Moreover, each tree's root node corresponds to some constant node weight that by definition obeys monotonicity constraints, $T(x_j^{\alpha};\theta_{b,0}) = T(x_j^{\beta};\theta_{b,0}) = w_{b,0}$. Together these additional splitting rules and node weight constraints ensure that $g^{\text{MGBM}}(x_j^{\alpha}) \leq g^{\text{MGBM}}(x_j^{\beta})\ \forall\ x_j^{\alpha} \leq x_j^{\beta} \in X_j$. For a negative monotonic constraint, i.e., $g^{\text{MGBM}}(x_j^{\alpha}) \geq g^{\text{MGBM}}(x_j^{\beta})\ \forall\ x_j^{\alpha} \leq x_j^{\beta} \in X_j$, left and right splitting rules and node weight constraints are switched. Also consider that MGBM models with independent monotonicity constraints between some X_j and $\mathbf{y}$ likely restrict non-monotonic interactions between multiple X_j. Moreover, if monotonicity constraints are not applied to all $X_j \in \mathbf{X}$, any strong non-monotonic signal in training data associated with some important X_j maybe forced onto some other arbitrary unconstrained X_j under some g^{MGBM} models, compromising the end goal of interpretability.

Appendix B.3. Explainable Neural Network Details

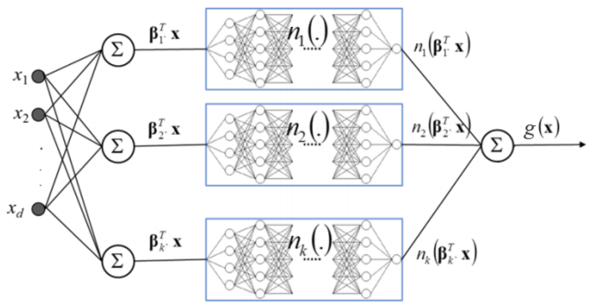

Figure A1. A diagram of an XNN with three metalayers: the bottom combination layer with K linear $\sum_j \beta_{k,j} x_j$ hidden units, the middle metalayer with with K hidden and separate n_k ridge functions, and the output combination layer that generates g^{XNN} predictions. Figure adapted from Vaughan *et al.* [20].

g^{XNN} is comprised of 3 meta-layers:

1. The first and deepest meta-layer, composed of K linear $\sum_j \beta_{k,j} x_j$ hidden units, which should learn higher magnitude weights for important X_j, is known as the *projection layer*. It is fully connected to each input X_j. Each hidden unit in the projection layer may optionally include a bias term.
2. The second meta-layer contains K hidden and separate n_k ridge functions, or *subnetworks*. Each n_k is a neural network, which can be parameterized to suit a given modeling task. To facilitate direct interpretation and visualization, the input to each subnetwork is the 1-dimensional output of its associated projection layer hidden unit, $\sum_j \beta_{k,j} x_j$. Each n_k can contain several bias terms.
3. The output meta-layer, called the *combination layer*, is another linear unit comprised of a global bias term, μ_0, and the K weighted 1-dimensional outputs of each subnetwork, $\gamma_k n_k(\sum_j \beta_{k,j} x_j)$. Again, subnetwork output is restricted to 1-dimension for interpretation and visualization purposes.

Appendix B.4. One-dimensional Partial Dependence and Individual Conditional Expectation Details

Following Friedman *et al.* [13] a single feature $X_j \in \mathbf{X}$ and its complement set $\mathbf{X}_{\mathcal{P}\setminus\{j\}} \in \mathbf{X}$ (where $X_j \cup \mathbf{X}_{\mathcal{P}\setminus\{j\}} = \mathbf{X}$) is considered. $\text{PD}(X_j, g)$ for a given feature X_j is estimated as the average output of the learned function $g(\mathbf{X})$ when all the observed instances of X_j are set to a constant $x \in \mathcal{X}$ and $\mathbf{X}_{\mathcal{P}\setminus\{j\}}$ is left unchanged. $\text{ICE}(x_j, g)$ for a given instance $\mathbf{x}$ and feature x_j is estimated as the output of $g(\mathbf{x})$ when x_j is set to a constant $x \in \mathcal{X}$ and all other features $\mathbf{x} \in \mathbf{X}_{\mathcal{P}\setminus\{j\}}$ are left untouched. PD and ICE curves are usually plotted over some set of constants $x \in \mathcal{X}$, as displayed in Subsection 2.2.2 and Appendix E.1. Due to known problems for PD in the presence of strong correlation and interactions, PD should not be used alone. PD should be paired with ICE or be replaced with accumulated local effect (ALE) plots [22], [29].

Appendix B.5. Shapley Value Details

For some instance $\mathbf{x} \in \mathcal{X}$, Shapley explanations take the form:

$$g(\mathbf{x}) = \phi_0 + \sum_{j=0}^{j=\mathcal{P}-1} \phi_j \mathbf{z}_j \tag{A2}$$

In Equation A2, $\mathbf{z} \in \{0,1\}^{\mathcal{P}}$ is a binary representation of $\mathbf{x}$ where 0 indicates missingness. Each ϕ_j is the local feature contribution value associated with x_j, and ϕ_0 is the average of $g(\mathbf{X})$. Each ϕ_j is a weighted combination of model scores, $g_x(\mathbf{x})$, with x_j, $g_x(S \cup \{j\})$, and the model scores without x_j, $g_x(S)$, for every subset of features S not including j, $S \subseteq \mathcal{P} \setminus \{j\}$, where g_x incorporates the mapping between $\mathbf{x}$ and the binary vector $\mathbf{z}$.

$$\phi_j = \sum_{S \subseteq \mathcal{P} \setminus \{j\}} \frac{|S|!(\mathcal{P} - |S| - 1)!}{\mathcal{P}!} [g_x(S \cup \{j\}) - g_x(S)] \tag{A3}$$

Local, per-instance explanations using Shapley values tend to involve ranking x_j by ϕ_j values or delineating a set of the X_j names associated with the k-largest ϕ_j values for some $\mathbf{x}$, where k is some small positive integer, say 5. Global explanations are typically the absolute mean of the ϕ_j associated with a given X_j across all of the instances in some set $\mathbf{X}$.

Appendix C. Types of Machine Learning Discrimination in US Legal and Regulatory Settings

It is important to explain and draw a distinction between the two major types of discrimination recognized in US legal and regulatory settings, disparate treatment (DT), and disparate impact (DI). DT (which is loosely referred to as *intentional discrimination*) occurs most often in an algorithmic setting when a model explicitly uses protected class status (e.g., race, sex) as an input feature or uses a feature that is so similar to protected class status that it essentially proxies for class membership. With some limited exceptions, the use of these factors in an algorithm is illegal under several statutes in the US.[4] DI, colloquially known as *unintentional discrimination*, occurs when some element of a decisioning process includes a *facially neutral* factor (i.e, a reasonable and valid predictor of response) that results in a disproportionate share of a protected class receiving an unfavorable outcome. In modeling, this is most typically driven by a statistically important feature that is distributed unevenly across classes, which causes more frequent unfavorable outcomes for the protected class. However, other factors, such as hyperparameter or algorithm choices, can drive DI. Crucially, legality hinges on whether changing the model, for example exchanging one feature for another or altering the hyperparameters of an algorithm, can lead to a similarly predictive model with lower DI.

Appendix D. Practical vs. Statistical Significance for Discrimination Testing

A finding of *practical significance* means that discovered disparity is not only statistically significant, but also passes beyond a chosen threshold that would constitute *prima facia* evidence of illegal discrimination. Practical significance acknowledges that any large dataset is likely to show statistically significant differences in outcomes by class, even if those differences are not truly meaningful. It further recognizes that there are likely to be situations where differences in outcomes are beyond a model user's ability to correct them without significantly degrading the quality of the model. Moreover, practical significance is also needed by model builders and compliance personnel to determine whether a model should undergo remediation efforts before it is put into production. Unfortunately, guidelines for practical significance, i.e., the threshold at which any statistically significant disparity would be considered evidence of illegal discrimination, are not as frequently codified as the standards for statistical significance. One exception, however, is in employment discrimination analyses, where the US Equal Employment Opportunity Commission (EEOC) has stated that if the AIR is below 0.80 and statistically significant, then this constitutes *prima facia* evidence of discrimination, which the model user must rebut in order for the DI not to be considered illegal discrimination.[22] It is important to note that the 0.80 measure of practical significance, also known as the *80% rule* and the *4/5ths rule*, is explicitly used in relation to AIR, and it is not clear that the use of this threshold is directly relevant to testing fairness for measures other than the AIR.

The legal thresholds for determining statistical significance is clearer and more consistent than that for practical significance. The first guidance in US courts occurred in a case involving discrimination in jury selection, *Castaneda vs. Partida*.[23] Here, the US Supreme Court wrote that, "As a general rule for such large samples, if the difference between the expected value and the observed number is greater than two or three standard deviations, then the hypothesis that the jury drawing was random would be suspect to a social scientist." This "two or three standard deviations" test was then applied to employment discrimination in *Hazelwood School Districts vs. United States*.[24] Out of this, a 5% two-sided test (z=1.96), or an equivalent 2.5% one-sided test, has become a common standard for determining whether evidence of disparities is statistically significant.

Appendix E. Additional Simulated Data Results

As seen in Subsection 2.1.1, little or no trade-off is required in terms of model to fit to use the constrained models. Hence, intrinsic interpretability, post-hoc explainability, and discrimination are explored further for the g^{MGBM} and g^{XNN} models in Appendices E.1 - E.2. Intrinsic interpretability for g^{MGBM} is evaluated with PD and ICE, and post-hoc explainability is highlighted via global and local Shapley explanations. For g^{XNN}, Shapley explanation techniques are also used to generate global and local feature importance to augment interpretability results exhibited in Subection 2.1.2. Both g^{MGBM} and g^{XNN} are evaluated for discrimination using AIR, ME, SMD, and other measures.

Appendix E.1. Interpretability and Post-hoc Explanation Results

Global mean absolute Shapley value feature importance for $g^{\text{MGBM}}(\mathbf{X})$ on the simulated test data is displayed in Figure A2.

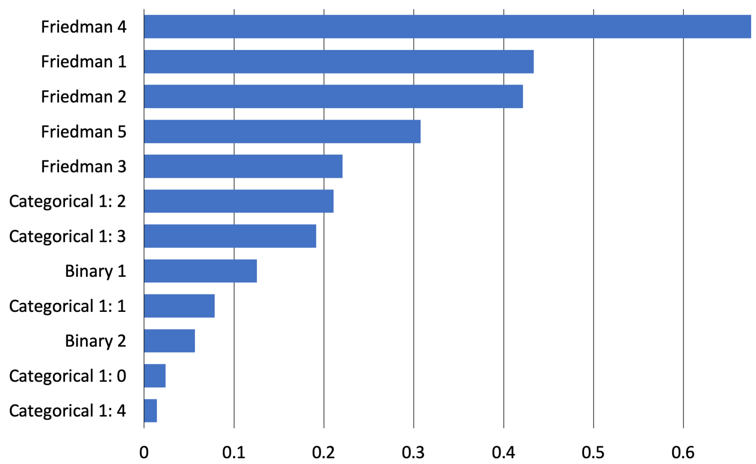

Figure A2. Global mean absolute Tree SHAP feature importance for $g^{\text{MGBM}}(\mathbf{X})$ on the simulated test data.

[22] Importantly, the standard of 0.80 is not a law, but a rule of thumb for agencies tasked with enforcement of discrimination laws. "Adoption of Questions and Answers To Clarify and Provide a Common Interpretation of the Uniform Guidelines on Employee Selection Procedures," Federal Register, Volume 44, Number 43 (1979).

[23] *Castaneda vs. Partida*, 430 US 482 - Supreme Court (1977)

[24] *Hazelwood School Dist. vs. United States*, 433 US 299 (1977)

As expected, the $X_{\text{Friedman},j}$ features from the original Friedman [10] and Friedman *et al.* [11] formula are the main drivers of $g^{\text{MGBM}}(\mathbf{X})$ predictions, with encoded versions of the augmented categorical and binary features contributing less on average to $g^{\text{MGBM}}(\mathbf{X})$ predictions.

Figure A3 highlights PD, ICE, and histograms of the most important features from Figure A2.

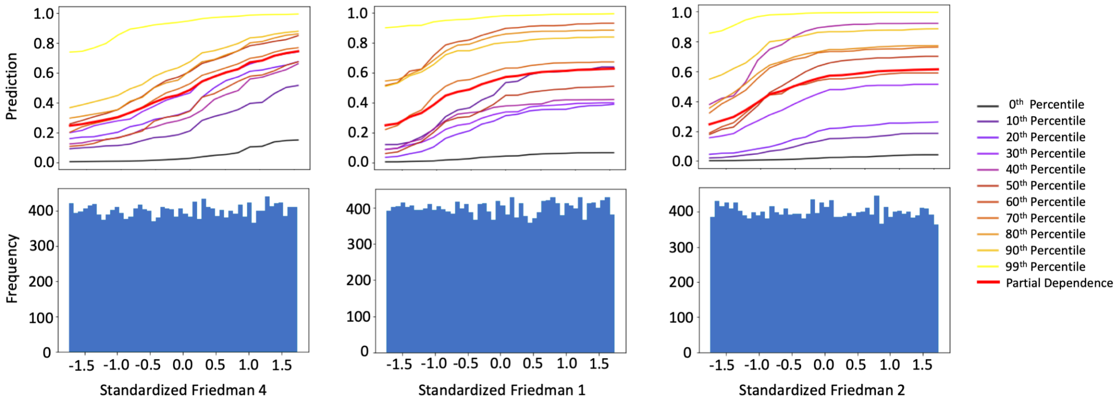

Figure A3. PD, ICE for 10 instances across selected percentiles of $g^{\text{MGBM}}(\mathbf{X})$, and histograms for the three most important input features of g^{MGBM} on the simulated test data.

$X_{\text{Friedman},1}$, $X_{\text{Friedman},2}$, and $X_{\text{Friedman},4}$ were positively monotonically constrained under g^{MGBM} for the simulated data, and positive monotonicity looks to be confirmed on average with PD and at numerous local percentiles of $g^{\text{MGBM}}(\mathbf{X})$ with ICE. Also, as the PD curves generally follow the patterns of the ICE curves, PD is likely an accurate representation of average feature behavior for $X_{\text{Friedman},1}$, $X_{\text{Friedman},2}$, and $X_{\text{Friedman},4}$. Since PD and ICE curves do not obviously diverge, g^{MGBM} is probably not modeling strong interactions, despite the fact that known interactions are included in the simulated data signal generating function in Equation 1. The one-dimensional monotonic constraints may hinder the modeling of such interactions, but do not strongly affect overall g^{MGBM} accuracy, perhaps due to noise in the simulated data. This is an interesting result for responsible ML practitioners. In some noisy scenarios, monotonicity constraints can increase model interpretability without causing a drastic drop in model accuracy, even when known interactions and non-monotonic behavior exist in training data.

Local Shapley values for records at the 10^{th}, 50^{th}, and 90^{th} percentiles of $g^{\text{MGBM}}(\mathbf{X})$ in the simulated test data are displayed in Figure A4.

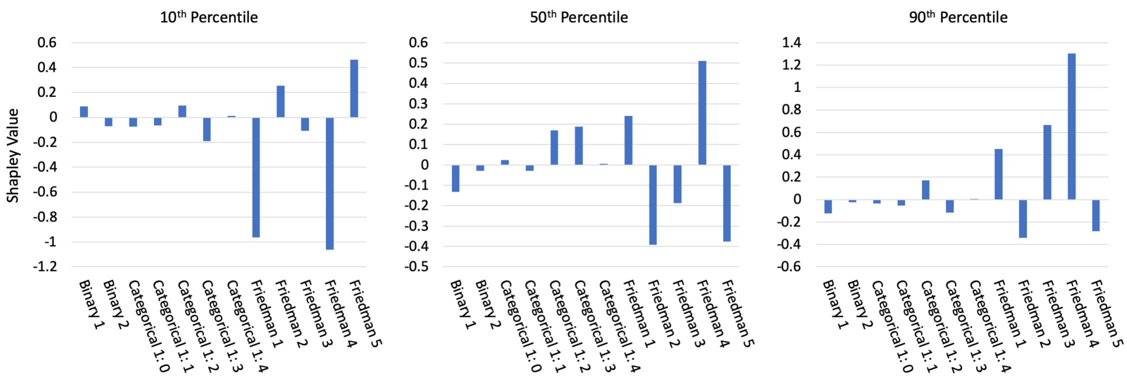

Figure A4. Tree SHAP values for three instances across selected percentiles of $g^{\text{MGBM}}(\mathbf{X})$ for the simulated test data.

The Shapley values in Figure A4 appear to be a logical result. For the lower prediction at the 10^{th} percentile of $g^{\text{MGBM}}(\mathbf{X})$, the largest local contributions are negative and the majority of local contributions are also negative. At the median of $g^{\text{MGBM}}(\mathbf{X})$, local contributions are roughly split between positive and negative values, and at the 90$^{\text{th}}$ of $g^{\text{MGBM}}(\mathbf{X})$, most large contributions are positive. In each case, large local contributions generally follow global importance results in Figure A2 as well.

Figure A5 shows global mean absolute Shapley feature importance for $g^{\text{XNN}}(\mathbf{X})$ on the simulated test data, using the approximate Deep SHAP technique.

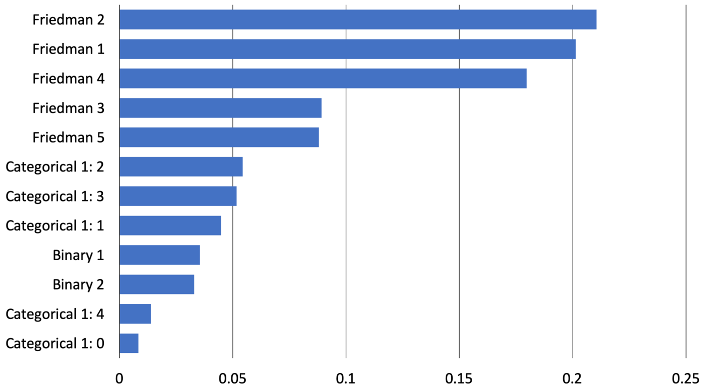

Figure A5. Global mean Deep SHAP feature importance for $g^{\text{XNN}}(\mathbf{X})$ on the simulated test data.

Like g^{MGBM}, g^{XNN} ranks the $X_{\text{Friedman},j}$ features higher in terms of importance than the categorical and binary features. The consistency between the feature rankings of g^{MGBM} and g^{XNN} is somewhat striking, given their different hypothesis families and architectures. Both g^{MGBM} and g^{XNN} rank $X_{\text{Friedman},1}$, $X_{\text{Friedman},2}$, and $X_{\text{Friedman},4}$ as the most important features, both place $X_{\text{Categorical},2}$ and $X_{\text{Categorical},3}$ above the $X_{\text{Binary},1}$ and $X_{\text{Binary},2}$ features, both rank $X_{\text{Binary},1}$ above $X_{\text{Binary},2}$, and both place the least importance on $X_{\text{Categorical},4}$ and $X_{\text{Categorical},0}$.

Local Deep SHAP feature importance in Figure A6 supplements the global interpretability of g^{XNN} displayed in Figures A5 and 1. Local Deep SHAP values are extracted from the projection layer of g^{XNN} and reported in the probability space. Deep SHAP values can be calculated for any arbitrary $g^{\text{XNN}}(\mathbf{x})$, allowing for detailed, local summarization of individual model predictions.

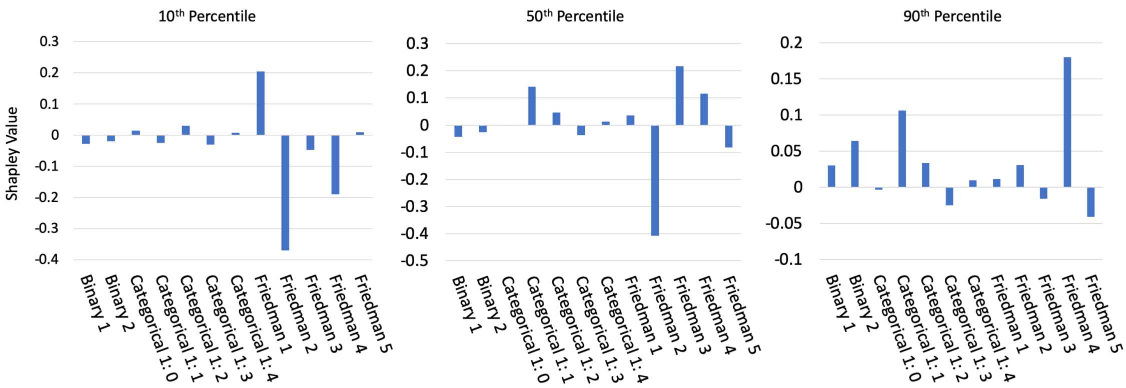

Figure A6. Deep SHAP values for three instances across selected percentiles of $g^{\text{XNN}}(\mathbf{X})$ on the simulated test data.

As expected, Deep SHAP values generally increase from the 10$^{\text{th}}$ percentile of $g^{\text{XNN}}(\mathbf{X})$ to the 90$^{\text{th}}$ percentile of $g^{\text{XNN}}(\mathbf{X})$, with primarily important global drivers of model behavior contributing to the selected local $g^{\text{XNN}}(\mathbf{x}^{(i)})$ predictions.

Appendix E.2. Discrimination Testing Results

Tables A1**a** and A1**b** show the results of the disparity tests using the simulated data for two hypothetical sets of class-control groups. Several measures of disparities are shown, with the SMDs calculated using the probabilities from g^{MGBM} and g^{XNN}, FNRs, their ratios, MEs, and AIRs calculated using a binary outcome based on a cutoff of 0.6 (anyone with probabilities of 0.6 or greater receives the favorable outcome).[11]

Since g^{MGBM} and g^{XNN} assume that a higher score is favorable (as might be the case if the model were predicting responses to marketing offers), one might consider the relative FNRs as a measure of the class-control disparities. Table A1**b** shows that protected group 1 has higher relative FNRs under g^{XNN} (1.13 vs. 1.06). However, in Table A1**a** the overall FNRs were lower for g^{XNN} (0.357 vs. 0.401). This illustrates a danger in considering relative class-control measures in isolation when comparing across models: despite the g^{MGBM} appearing to be a relatively fairer model, more protected group 1 members experience negative outcomes using g^{MGBM}. This is because FNR accuracy improves for both the protected group 1 and control group 1, but members of control group 1 benefit more than those in protected group 1. Of course, the choice of which model is truly fairer is a policy question.

(a) Group size, accuracy, and FNR for $g^{\text{MGBM}}(\mathbf{X})$ and $g^{\text{XNN}}(\mathbf{X})$ on the simulated test data.

Class	N	Model	Accuracy↑	FNR↓
Protected 1	3,057	g^{MGBM} g^{XNN}	0.770 0.771	0.401 0.357
Control 1	16,943	g^{MGBM} g^{XNN}	0.739 0.756	0.378 0.314
Protected 2	9,916	g^{MGBM} g^{XNN}	0.758 0.762	0.331 0.302
Control 2	10,084	g^{MGBM} g^{XNN}	0.729 0.756	0.420 0.332

(b) AIR, ME, SMD, and FNR ratio for $g^{\text{MGBM}}(\mathbf{X})$ and $g^{\text{XNN}}(\mathbf{X})$ on the simulated test data.

Model	Protected Class	Control Class	AIR↑	ME↓	SMD↓	FNR Ratio↓
g^{MGBM}	1 2	1 2	0.752 1.10	9.7% -3.6%	-0.206 0.106	1.06 0.788
g^{XNN}	1 2	1 2	0.727 0.976	12.0% 1.0%	-0.274 0.001	1.13 0.907

Table A1. Discrimination measures for the simulated test data. Arrows indicate the direction of improvement for each measure.

For g^{XNN}, 12.0% fewer control group 1 members receive the favorable offer under the ME column in Table A1**b**. Of note is that 12.0% is not a meaningful difference without context. If the population of control group 1 and control group 2 were substantially similar in relevant characteristics, 12.0% could represent an extremely large difference and would require remediation. But if they represent substantially different populations, then 12.0% could represent a reasonable deviation from parity. As an example, if a lending institution that has traditionally focused on high credit quality clients were to expand into previously under-banked communities, an 12.0% class-control difference in loan approval rates might be expected because the average credit quality of the new population would be lower than that of the existing population. Protected group 1's AIR under g^{XNN} is 0.727, below the EEOC 4/5ths rule threshold. It is also highly statistically significant (not shown, but available in resources discussed in Subsection 1.8). Together these would indicate that there may be evidence of illegal DI. As with ME and other measures, the reasonableness of this disparity is not clear outside of context. However, most regulated institutions that do perform discrimination analyses would find an AIR of this magnitude concerning and warranting further review. Some pertinent remediation strategies for discovered discrimination are discussed in Subsection 3.3.

SMD is used here to measure $g^{\text{MGBM}}(\mathbf{X})$ and $g^{\text{MGBM}}(\mathbf{X})$ probabilities prior to being transformed into classifications. (This measurement would be particularly relevant if the probabilities are used in combination with other models to determine an outcome.) The results show that g^{MGBM} has less DI than g^{XNN} (-0.206 vs. -0.274), but both are close to Cohen's small effect threshold of -0.20. Whether a small effect would be a highlighted concern would depend on a organization's chosen threshold for flagging models for further review.

Appendix F. Discrimination Testing and Cutoff Selection

The selection of which cutoff to use in production is typically based on the model's use case, rather than one based solely on the statistical properties of the predictions themselves. For example, a model developer at a bank might build a credit model where the F1 score is maximized at a delinquency probability cutoff of 0.15. For purposes of evaluating the quality of the model, she may review confusion matrix statistics (accuracy, recall, precision, etc.) using cutoffs based on the maximum F1 score. But, because of its risk tolerance and other factors, the bank itself might be willing to lend to anyone with a delinquency probability of 0.18 or lower, which would mean that anyone who is scored at 0.18 or lower would receive an offer of credit. Because disparity analyses are concerned with how people are affected by the deployed model, it is essential that any confusion matrix-based measures of disparity be calculated on the in-production classification decisions, rather than the cutoffs that are not related to what those affected by the model will experience.

Appendix G. Recent Fairness Techniques in US Legal and Regulatory Settings

Great care must be taken in order to ensure that the appropriate fairness measures are chosen, because certain measures may not be appropriate for some use cases. Additionally, the effects of changing the model must be viewed holistically. For example, the mortgage data disparity analysis in Subsection 2.2.3 shows that if one were to choose g^{MGBM} over g^{XNN} because g^{MGBM} has a lower FPR ratio for blacks, it would ultimately lead to a higher FPR for blacks overall, which may represent doing more harm than good.

Furthermore, using some recently developed discrimination mitigation methods may lead to non-compliance with anti-discrimination laws and regulations. A fundamental maxim of US anti-discrimination law is that (to slightly paraphrase), "similarly situated people should be treated similarly."[25] A model developed without inclusion of class status (or proxies thereof) considers similarly situated people the same on the dimensions included in the model: people who have the same feature values will have the same model output (though there may be some small or random differences in outcomes due to computational issues). Obviously, the inclusion of protected class status will change model output by class. With possible rare exceptions, this is likely to be viewed with legal and regulatory skepticism today, even if including class status is done with fairness as the goal.[26] Preprocessing and post-processing techniques may be similarly problematic, because industries that must provide explanations to those who receive unfavorable treatment (e.g., adverse action notices in US financial services) may have to incorporate the class adjustments into their explanations as well.

References

1. Rudin, C. Please Stop Explaining Black Box Models for High Stakes Decisions and Use Interpretable Models Instead. *arXiv preprint arXiv:1811.10154* **2018**. URL: https://arxiv.org/pdf/1811.10154.pdf.
2. Feldman, M.; Friedler, S.A.; Moeller, J.; Scheidegger, C.; Venkatasubramanian, S. Certifying and Removing Disparate Impact. Proceedings of the 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 2015, pp. 259–268. URL: https://arxiv.org/pdf/1412.3756.pdf.
3. Dwork, C.; Hardt, M.; Pitassi, T.; Reingold, O.; Zemel, R. Fairness Through Awareness. Proceedings of the 3rd Innovations in Theoretical Computer Science Conference. ACM, 2012, pp. 214–226. URL: https://arxiv.org/pdf/1104.3913.pdf.

[25] In the pay discrimination case, *Bazemore vs. Friday*, 478 US 385 (1986), the US Supreme Court found that,"Each week's paycheck that delivers less to a black than to a similarly situated white is a wrong actionable ..." Beyond the obvious conceptual meaning, what specifically constitutes *similarly situated* is controversial and its interpretation differs by circuit.

[26] In a reverse discrimination case, *Ricci v Desafano*, 557 US 557 (2009), the court found that any consideration of race which is not justified by correcting for past proven discrimination is illegal and, moreover, a lack of fairness is not necessarily evidence of illegal discrimination.

4. Buolamwini, J.; Gebru, T. Gender Shades: Intersectional Accuracy Disparities in Commercial Gender Classification. Conference on Fairness, Accountability and Transparency, 2018, pp. 77–91. URL: http://proceedings.mlr.press/v81/buolamwini18a/buolamwini18a.pdf.
5. Barreno, M.; Nelson, B.; Joseph, A.D.; Tygar, J. The Security of Machine Learning. *Machine Learning* **2010**, *81*, 121–148. URL: http://people.ischool.berkeley.edu/~tygar/papers/SML/sec_mach_learn_journal.pdf.
6. Tramèr, F.; Zhang, F.; Juels, A.; Reiter, M.K.; Ristenpart, T. Stealing Machine Learning Models via Prediction APIs. 25th {USENIX} Security Symposium ({USENIX} Security 16), 2016, pp. 601–618. URL: https://www.usenix.org/system/files/conference/usenixsecurity16/sec16_paper_tramer.pdf.
7. Shokri, R.; Stronati, M.; Song, C.; Shmatikov, V. Membership Inference Attacks Against Machine Learning Models. 2017 IEEE Symposium on Security and Privacy (SP). IEEE, 2017, pp. 3–18. URL: https://arxiv.org/pdf/1610.05820.pdf.
8. Shokri, R.; Strobel, M.; Zick, Y. Privacy Risks of Explaining Machine Learning Models. *arXiv preprint arXiv:1907.00164* **2019**. URL: https://arxiv.org/pdf/1907.00164.pdf.
9. Williams, M.; others. *Interpretability*; Fast Forward Labs, 2017. URL: https://www.cloudera.com/products/fast-forward-labs-research.html.
10. Friedman, J.H. A Tree-structured Approach to Nonparametric Multiple Regression. In *Smoothing Techniques for Curve Estimation*; Springer, 1979; pp. 5–22. URL: http://inspirehep.net/record/140963/files/slac-pub-2336.pdf.
11. Friedman, J.H.; others. Multivariate Adaptive Regression Splines. *The Annals of Statistics* **1991**, *19*, 1–67. URL: https://projecteuclid.org/download/pdf_1/euclid.aos/1176347963.
12. Friedman, J.H. Greedy Function Approximation: a Gradient Boosting Machine. *The Annals of Statistics* **2001**, pp. 1189–1232. URL: https://projecteuclid.org/download/pdf_1/euclid.aos/1013203451.
13. Friedman, J.H.; Hastie, T.; Tibshirani, R. ***The Elements of Statistical Learning***; Springer: New York, 2001. URL: https://web.stanford.edu/~hastie/ElemStatLearn/printings/ESLII_print12.pdf.
14. Recht, B.; Re, C.; Wright, S.; Niu, F. HOGWILD: A Lock-free Approach to Parallelizing Stochastic Gradient Descent. Advances in Neural Information Processing Systems (NIPS), 2011, pp. 693–701. URL: https://papers.nips.cc/paper/4390-hogwild-a-lock-free-approach-to-parallelizing-stochastic-gradient-descent.pdf.
15. Hinton, G.E.; Srivastava, N.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R.R. Improving Neural Networks by Preventing Co-adaptation of Feature Detectors. *arXiv preprint arXiv:1207.0580* **2012**. URL: https://arxiv.org/pdf/1207.0580.pdf.
16. Sutskever, I.; Martens, J.; Dahl, G.; Hinton, G. On the Importance of Initialization and Momentum in Deep Learning. International Conference on Machine Learning, 2013, pp. 1139–1147. URL: http://proceedings.mlr.press/v28/sutskever13.pdf.
17. Zeiler, M.D. ADADELTA: An Adaptive Learning Rate Method. *arXiv preprint arXiv:1212.5701* **2012**. URL: https://arxiv.org/pdf/1212.5701.pdf.
18. Aïvodji, U.; Arai, H.; Fortineau, O.; Gambs, S.; Hara, S.; Tapp, A. Fairwashing: The Risk of Rationalization. *arXiv preprint arXiv:1901.09749* **2019**. URL: https://arxiv.org/pdf/1901.09749.pdf.
19. Slack, D.; Hilgard, S.; Jia, E.; Singh, S.; Lakkaraju, H. Fooling LIME and SHAP: Adversarial Attacks on Post-hoc Explanation Methods. *arXiv preprint arXiv:1911.02508* **2019**. URL: https://arxiv.org/pdf/1911.02508.pdf.
20. Vaughan, J.; Sudjianto, A.; Brahimi, E.; Chen, J.; Nair, V.N. Explainable Neural Networks Based on Additive Index Models. *arXiv preprint arXiv:1806.01933* **2018**. URL: https://arxiv.org/pdf/1806.01933.pdf.
21. Yang, Z.; Zhang, A.; Sudjianto, A. Enhancing Explainability of Neural Networks Through Architecture Constraints. *arXiv preprint arXiv:1901.03838* **2019**. URL: https://arxiv.org/pdf/1901.03838.pdf.
22. Goldstein, A.; Kapelner, A.; Bleich, J.; Pitkin, E. Peeking Inside the Black Box: Visualizing Statistical Learning with Plots of Individual Conditional Expectation. *Journal of Computational and Graphical Statistics* **2015**, *24*. URL: https://arxiv.org/pdf/1309.6392.pdf.
23. Lundberg, S.M.; Lee, S.I. A Unified Approach to Interpreting Model Predictions. In *Advances in Neural Information Processing Systems (NIPS)*; Guyon, I.; Luxburg, U.V.; Bengio, S.; Wallach, H.; Fergus, R.; Vishwanathan, S.; Garnett, R., Eds.; Curran Associates, Inc., 2017; pp. 4765–4774. URL: http://papers.nips.cc/paper/7062-a-unified-approach-to-interpreting-model-predictions.pdf.

24. Lundberg, S.M.; Erion, G.G.; Lee, S.I. Consistent Individualized Feature Attribution for Tree Ensembles. In *Proceedings of the 2017 ICML Workshop on Human Interpretability in Machine Learning (WHI 2017)*; Kim, B.; Malioutov, D.M.; Varshney, K.R.; Weller, A., Eds.; ICML WHI 2017, 2017; pp. 15–21. URL: https://openreview.net/pdf?id=ByTKSo-m-.
25. Cohen, J. ***Statistical Power Analysis for the Behavioral Sciences***; Lawrence Erlbaum Associates, 1988. URL: http://www.utstat.toronto.edu/~brunner/oldclass/378f16/readings/CohenPower.pdf.
26. Cohen, J. A Power Primer. *Psychological Bulletin* **1992**, *112*, 155. URL: https://www.ime.usp.br/~abe/lista/pdfn45sGokvRe.pdf.
27. Zafar, M.B.; Valera, I.; Gomez Rodriguez, M.; Gummadi, K.P. Fairness Beyond Disparate Treatment & Disparate Impact: Learning Classification Without Disparate Mistreatment. Proceedings of the 26th International Conference on World Wide Web. International World Wide Web Conferences Steering Committee, 2017, pp. 1171–1180. URL: https://arxiv.org/pdf/1610.08452.pdf.
28. Lou, Y.; Caruana, R.; Gehrke, J.; Hooker, G. Accurate Intelligible Models with Pairwise Interactions. Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 2013, pp. 623–631. URL: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.352.7682&rep=rep1&type=pdf.
29. Apley, D.W. Visualizing the Effects of Predictor Variables in Black Box Supervised Learning Models. *arXiv preprint arXiv:1612.08468* **2016**. URL: https://arxiv.org/pdf/1612.08468.pdf.
30. Shapley, L.S.; Roth, A.E.; others. ***The Shapley value: Essays in Honor of Lloyd S. Shapley***; Cambridge University Press, 1988. URL: http://www.library.fa.ru/files/Roth2.pdf.
31. Hall, P. On the Art and Science of Machine Learning Explanations. KDD '19 XAI Workshop, 2019. URL: https://arxiv.org/pdf/1810.02909.pdf.
32. Molnar, C. ***Interpretable Machine Learning***; christophm.github.io, 2018. URL: https://christophm.github.io/interpretable-ml-book/.
33. Hu, X.; Rudin, C.; Seltzer, M. Optimal Sparse Decision Trees. *arXiv preprint arXiv:1904.12847* **2019**. URL: https://arxiv.org/pdf/1904.12847.pdf.
34. Friedman, J.H.; Popescu, B.E.; others. Predictive Learning Via Rule Ensembles. *The Annals of Applied Statistics* **2008**, *2*, 916–954. URL: https://projecteuclid.org/download/pdfview_1/euclid.aoas/1223908046.
35. Gupta, M.; Cotter, A.; Pfeifer, J.; Voevodski, K.; Canini, K.; Mangylov, A.; Moczydlowski, W.; Van Esbroeck, A. Monotonic Calibrated Interpolated Lookup Tables. *The Journal of Machine Learning Research* **2016**, *17*, 3790–3836. URL: http://www.jmlr.org/papers/volume17/15-243/15-243.pdf.
36. Chen, C.; Li, O.; Barnett, A.; Su, J.; Rudin, C. This Looks Like That: Deep Learning for Interpretable Image Recognition. Proceedings of Neural Information Processing Systems (NeurIPS), 2019. URL: https://arxiv.org/pdf/1806.10574.pdf.
37. Wilkinson, L. Visualizing Big Data Outliers through Distributed Aggregation. *IEEE Transactions on Visualization & Computer Graphics* **2018**. URL: https://www.cs.uic.edu/~wilkinson/Publications/outliers.pdf.
38. Udell, M.; Horn, C.; Zadeh, R.; Boyd, S.; others. Generalized Low Rank Models. *Foundations and Trends® in Machine Learning* **2016**, *9*, 1–118. URL: https://www.nowpublishers.com/article/Details/MAL-055.
39. Holohan, N.; Braghin, S.; Mac Aonghusa, P.; Levacher, K. Diffprivlib: The IBM Differential Privacy Library. *arXiv preprint arXiv:1907.02444* **2019**. URL: https://arxiv.org/pdf/1907.02444.pdf.
40. Ji, Z.; Lipton, Z.C.; Elkan, C. Differential Privacy and Machine Learning: A Survey and Review. *arXiv preprint arXiv:1412.7584* **2014**. URL: https://arxiv.org/pdf/1412.7584.pdf.
41. Papernot, N.; Song, S.; Mironov, I.; Raghunathan, A.; Talwar, K.; Erlingsson, Ú. Scalable Private Learning with PATE. *arXiv preprint arXiv:1802.08908* **2018**. URL: https://arxiv.org/pdf/1802.08908.pdf.
42. Abadi, M.; Chu, A.; Goodfellow, I.; McMahan, H.B.; Mironov, I.; Talwar, K.; Zhang, L. Deep Learning with Differential Privacy. Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security. ACM, 2016, pp. 308–318. URL: https://arxiv.org/pdf/1607.00133.pdf.
43. Pearl, J.; Mackenzie, D. *The Book of Why: The New Science of Cause and Effect*; Basic Books, 2018. URL: http://cdar.berkeley.edu/wp-content/uploads/2017/04/Lisa-Goldberg-reviews-The-Book-of-Why.pdf.
44. Wachter, S.; Mittelstadt, B.; Russell, C. Counterfactual Explanations without Opening the Black Box: Automated Decisions and the GPDR. *Harv. JL & Tech.* **2017**, *31*, 841. URL: https://arxiv.org/pdf/1711.00399.pdf.

45. Ancona, M.; Ceolini, E.; Oztireli, C.; Gross, M. Towards Better Understanding of Gradient-based Attribution Methods for Deep Neural Networks. 6th International Conference on Learning Representations (ICLR 2018), 2018. URL: https://www.research-collection.ethz.ch/bitstream/handle/20.500.11850/249929/Flow_ICLR_2018.pdf.
46. Wallace, E.; Tuyls, J.; Wang, J.; Subramanian, S.; Gardner, M.; Singh, S. AllenNLP Interpret: A Framework for Explaining Predictions of NLP Models. *arXiv preprint arXiv:1909.09251* **2019**. URL: https://arxiv.org/pdf/1909.09251.pdf.
47. Kamiran, F.; Calders, T. Data Preprocessing Techniques for Classification Without Discrimination. *Knowledge and Information Systems* **2012**, *33*, 1–33. URL: https://bit.ly/2lH95lQ.
48. Zhang, B.H.; Lemoine, B.; Mitchell, M. Mitigating Unwanted Biases with Adversarial Learning. Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society. ACM, 2018, pp. 335–340. URL: https://arxiv.org/pdf/1801.07593.pdf.
49. Zemel, R.; Wu, Y.; Swersky, K.; Pitassi, T.; Dwork, C. Learning Fair Representations. International Conference on Machine Learning, 2013, pp. 325–333. URL: http://proceedings.mlr.press/v28/zemel13.pdf.
50. Kamiran, F.; Karim, A.; Zhang, X. Decision Theory for Discrimination-aware Classification. 2012 IEEE 12th International Conference on Data Mining. IEEE, 2012, pp. 924–929. URL: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.722.3030&rep=rep1&type=pdf.
51. Rauber, J.; Brendel, W.; Bethge, M. Foolbox: A Python Toolbox to Benchmark the Robustness of Machine Learning Models. *arXiv preprint arXiv:1707.04131* **2017**. URL: https://arxiv.org/pdf/1707.04131.pdf.
52. Papernot, N.; Faghri, F.; Carlini, N.; Goodfellow, I.; Feinman, R.; Kurakin, A.; Xie, C.; Sharma, Y.; Brown, T.; Roy, A.; Matyasko, A.; Behzadan, V.; Hambardzumyan, K.; Zhang, Z.; Juang, Y.L.; Li, Z.; Sheatsley, R.; Garg, A.; Uesato, J.; Gierke, W.; Dong, Y.; Berthelot, D.; Hendricks, P.; Rauber, J.; Long, R. Technical Report on the CleverHans v2.1.0 Adversarial Examples Library. *arXiv preprint arXiv:1610.00768* **2018**. URL: https://arxiv.org/pdf/1610.00768.pdf.
53. Amershi, S.; Chickering, M.; Drucker, S.M.; Lee, B.; Simard, P.; Suh, J. Modeltracker: Redesigning Performance Analysis Tools for Machine Learning. Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems. ACM, 2015, pp. 337–346. URL: https://www.microsoft.com/en-us/research/wp-content/uploads/2016/02/amershi.CHI2015.ModelTracker.pdf.
54. Papernot, N. A Marauder's Map of Security and Privacy in Machine Learning: An overview of current and future research directions for making machine learning secure and private. Proceedings of the 11th ACM Workshop on Artificial Intelligence and Security. ACM, 2018. URL: https://arxiv.org/pdf/1811.01134.pdf.
55. Mitchell, M.; Wu, S.; Zaldivar, A.; Barnes, P.; Vasserman, L.; Hutchinson, B.; Spitzer, E.; Raji, I.D.; Gebru, T. Model Cards for Model Reporting. Proceedings of the Conference on Fairness, Accountability, and Transparency. ACM, 2019, pp. 220–229. URL: https://arxiv.org/pdf/1810.03993.pdf.
56. Bracke, P.; Datta, A.; Jung, C.; Sen, S. Machine Learning Explainability in Finance: An Application to Default Risk Analysis **2019**. URL: https://www.bankofengland.co.uk/-/media/boe/files/working-paper/2019/machine-learning-explainability-in-finance-an-application-to-default-risk-analysis.pdf.
57. Friedler, S.A.; Scheidegger, C.; Venkatasubramanian, S.; Choudhary, S.; Hamilton, E.P.; Roth, D. A Comparative Study of Fairness-enhancing Interventions in Machine Learning. Proceedings of the Conference on Fairness, Accountability, and Transparency. ACM, 2019, pp. 329–338. URL: https://arxiv.org/pdf/1802.04422.pdf.
58. Schmidt, N.; Stephens, B. An Introduction to Artificial Intelligence and Solutions to the Problems of Algorithmic Discrimination. *Conference on Consumer Finance Law Quarterly Report* **2019**, *73*, 130–144. URL: https://arxiv.org/pdf/1911.05755.pdf.
59. Hoare, C.A.R. The 1980 ACM Turing Award Lecture. *Communications* **1981**. URL: http://www.cs.fsu.edu/~engelen/courses/COP4610/hoare.pdf.