scientific reports

Check for updates

OPEN

Addressing bias in bagging and boosting regression models

Juliette Ugirumurera[1]✉, Erik A. Bensen[2], Joseph Severino[1] & Jibonananda Sanyal[1]

As artificial intelligence (AI) becomes widespread, there is increasing attention on investigating bias in machine learning (ML) models. Previous research concentrated on classification problems, with little emphasis on regression models. This paper presents an easy-to-apply and effective methodology for mitigating bias in bagging and boosting regression models, that is also applicable to any model trained through minimizing a differentiable loss function. Our methodology measures bias rigorously and extends the ML model's loss function with a regularization term to penalize high correlations between model errors and protected attributes. We applied our approach to three popular tree-based ensemble models: a random forest model (RF), a gradient-boosted model (GBT), and an extreme gradient boosting model (XGBoost). We implemented our methodology on a case study for predicting road-level traffic volume, where RF, GBT, and XGBoost models were shown to have high accuracy. Despite high accuracy, the ML models were shown to perform poorly on roads in minority-populated areas. Our bias mitigation approach reduced minority-related bias by over 50%.

Keywords Artificial intelligence, Fair machine learning, Bias in machine learning, XGBoost, Random forest, Gradient-boosted trees

Artificial intelligence (AI) models have become ubiquitous in many areas of society, including banking, human resource management, health care, criminal justice, law enforcement, and transportation. Though AI models have enabled innovative progress in many sectors, they have also been shown to introduce and perpetuate biases that can exacerbate existing inequities. For instance, Angwin et al. found that the COMPAS system for predicting the risk of re-offending predicted higher risks for black defendants than their actual risk and higher than the risk for white defendants[1]. Google's advertisement AI algorithm was found to show fewer ads for high-paying jobs to women than men[2]. Object-detection AI models used in autonomous vehicles were also shown to have poor performance when detecting pedestrians with dark skin tones[3].

In spite of growing interest in researching methods for enforcing fairness in AI algorithms, to the best of our knowledge, most works have focused on classification, neglecting regression[4]. In this paper, we present a methodology for addressing bias in tree-based bagging and boosting regression methods, that is also applicable to any machine learning (ML) model trained through minimizing a differentiable loss function. Tree-based methods are some of the most broadly used ML algorithms. They are nonlinear, but still easy to use. Tree-based ensemble methods, such as random forests (RFs)[5] and gradient boosting[6] are efficient and effective for many tasks across domains. RFs, with no hyperparameter tuning, were the best-performing method in a study of over one hundred data sets[7]. A scalable form of gradient boosting, XGBoost[8], dominates in many Kaggle competitions[9].

Related works

Research on AI bias has grown significantly in recent years. AI bias mitigation methods are generally grouped in three categories: pre-processing, in-processing, and post-processing methods. Pre-processing techniques focus on data wrangling to remove discrimination[10]; examples include resampling, and reweighing techniques[11]. Post-processing methods, such as empirical distribution matching approaches[12], adjust the output predictions after model training[13,14]. Though pre-processing and post-processing techniques are straightforward and can be easy to implement, they cannot address bias from ML models themselves[15]. In contrast, in-processing methods tackle bias by integrating fairness considerations directly into the model design process[16,17], thus producing inherently fair models. Our methodology falls in the in-processing method by adding a differentiable regularization term to the objective function of the learning algorithm.

Existing research in in-processing bias mitigation approaches, including for tree-based ensemble methods[14,24,25], have predominantly focused on classification problems[16,17,26–32]. A recent survey on fair classifiers

[1]Computational Science Center, National Renewable Energy Laboratory, 15013 Denver West Parkway, Golden, CO 80401, USA. [2]Department of Statistics and Data Science, Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA 15213, USA. ✉email: jugirumu@nrel.gov

presents over 200 publications on in-processing fairness approaches for classification[33]. In contrast, in-processing techniques have not been studied adequately for regression tasks. Table 1 compares our methodology with existing work in in-processing techniques for fair regression in terms of approach used (constrained optimization vs regularization), the training scheme used, the type of regression task addressed, and the bias metric used. As outlined, our work most closely relates to methods that use regularization, which are easier to apply compared to the constrained optimization approaches. Regularization approaches only require a modification to the objective loss function, while employing fairness constraints typically requires altering the training process by introducing an additional step to address the problem through constrained optimization methods. Previous research in regularization-based fair regression have only focused on linear regression, logistic regression[22] and decision trees[15,23]. In contrast, our approach is suitable for any regression ML model that is trained by minimizing a differentiable loss function.

Table 1 also highlights the variety in bias metrics used in research. Our work uses the overall group accuracy metric, which ensures that the model performs equally well across demographic groups[34]. This is implemented by adding a regularization term that penalizes high correlation between the model's error and a protected attribute of interest. While the appropriateness of a bias metric often depends on the specific use case, overall group accuracy offers distinct advantages: it is straightforward to understand and implement, and emphasizes fairness in model accuracy across different groups. This is unlike other metrics, such statistical parity[4,19,23] or correlation between protected attribute and predictions[20], which only ensure the model predictions are independent of protected attributes with no guarantee on accuracy. Other bias measures, such as equal mean predictions[18], equal treatment[22], or disparate impact[15], also focus on ensuring ML models treats similar individuals or groups equally, yet do not assure a specific level of accuracy.

Another novel area of work within AI bias which has gained traction in recent years is information theoretic bias mitigation techniques[35–38]. These methods focus on removing the information content of protected attributes from the training data so that ML models cannot predict the responses from protected attributes. The inability to predict the response from protected attributes is an important fairness metric for individual level decision making, such as ensuring that hiring decisions are not affected by a protected attribute like race or gender. In our work, we focus on a different notion of fairness where our model is defined as fair if the errors of the model predictions are independent of a demographic attribute. Since information-theoretic approaches focus on adjusting the training data to address bias, they can be classified as pre-processing fairness methods. As previously mentioned, such methods lack the capability to address biases inherent in the machine learning models themselves.

Our work investigated three correlation terms as regularization terms that capture different types of correlations: Pearson's coefficient, which measures linear correlation between two data sets; Kendall's tau, which can determine if two random variables are statistically dependent without assumptions on their underlying distributions and measures non-parametric, ordinal relations between random variables; and distance correlation, which measures both linear and non-linear associations between two random variables. This is because the baseline distribution of ML models' data sets is usually unknown. We then added the correction terms to the loss function of three popular tree-based ensemble regression algorithms: an RF model, a gradient boosted-tree model (GBT) and an extreme gradient boosting (XGBoost) model, and conducted a detailed analysis on the correlation terms' impact on the models' biases. We also present a statistical test for demographic bias using the residuals of an ML model and bootstrap resampling.

To implement our methodology, we extended the highly flexible XGBoost library[39], which can be configured to behave like RF and GBT models. Unlike other popular ML libraries, such as Scikit-Learn, that include tree-based ML models, the XGBoost framework allows the inclusion of custom loss functions. The code implementation of this extension to XGBoost framework, along with implementation of the correlation terms, is publicly available on github[40].

Hence, our work extend the state of the art as in fair AI as follows:

In-processing approach	Training scheme	Regression task	Bias metric
Constrained optimization	Controlling attribute effect[18]	Linear regression	Equal mean predictions and residuals across groups
	Supervised learning oracles[19]	Regression with Lipschitz continuous loss functions	Statistical parity and bounded group loss
	Non-convex optimization[20]	Least squares regression and non-linear least square regression	Correlation between protected attributes and predictions
	Counterfactual fairness[21]	Logistic regression or neural network	Counterfactual fairness
	Quadrature approaches[4]	Kernel regression	Statistical parity
Regularization	Convex fair regression[22]	Linear and logistic regression	Equal treatment across individuals and groups
	Fair induction algorithm[23]	Decision trees	Statistical parity
	Mixed integer optimization[15]	Decision trees	Disparate impact
	Penalize correlation between model error and protected attribute [this work]	**Regression with differentiable loss functions**	**Overall group accuracy**

Table 1. Comparison of our work with existing research in in-processing methods for fair regression.

- An easy-to-apply in-processing regularization bias mitigation method for tree-based regression methods, that is usable for any ML model trained through minimizing a differentiable loss function.
- Application of three correlation terms, Pearson's coefficient, Kendall's tau, and distance correlation, as regularization terms for bias correction in three popular tree-based ML models: RF, GBT, and XGBoost.
- A detailed analysis of the effectiveness of the correlations terms to reduce the bias of the tree-based ML models.
- An open-source code extension to the XGBoost library to enable bias correction and custom loss function for RF, GBT, and XGBoost models.

Results

Numerical case study

In this paper, we measure and address bias in tree-based ensemble regression models used to predict network-level traffic volume at the road-link level[41]. Though scarce, quality network-wide traffic volume data at the road level is necessary to measure and understand the state of transportation systems. Traffic volume data also enables the user to accurately model and simulate traffic, to design traffic management policies, to quantify vehicle miles traveled, to calculate mobility-related fuel consumption and greenhouse emissions, and to inform transportation electrification and decarbonization efforts. Researchers at the National Renewable Energy Laboratory have developed an ML-based method that uses tree-based ensemble regression models, an RF, a GBT, and a XGBoost, to predict hourly network-level traffic volume estimates[41]. The ML models take as input TomTom probe count, which is the number of GPS devices recorded[42], weather conditions, road characteristics (speed and road class), and temporal information such as time of day and day of week. The ground truth training data comprise hourly traffic volume data from volume stations, which are traffic count sites to track the usage of roadways. All three models were shown to have high accuracy, with the XGBoost having the highest accuracy.

This ML-based method was applied to estimate the traffic volume in many regions. In this work, we sought to determine if these volume estimation ML models had bias in their performance for particular areas or population groups in spite of having a high overall accuracy. To do this, we considered a case study in which these ML models were used to estimate traffic volume in Hamilton County, in Chattanooga, Tennessee[43]. The overall performance of the models for this county was an R^2 of 0.873 for the XGBoost, 0.868 for the GBT, and 0.833 for the RF. The Hamilton County region was the area of study for the Regional Mobility project[44], which was funded by the Vehicle Technology Office of the U.S. Department of Energy, conducted by the National Renewable Energy Laboratory in conjunction with Oak Ridge National Laboratory, and aimed to improve mobility energy efficiency by deploying adaptive traffic signal control algorithms. The volume prediction ML models were used to identify areas of high traffic congestion to target them for traffic signal operation optimization.

In Table 2, we show descriptive statistics of some of the traffic and weather columns of the case study dataset. The training data consisted of 8,992 entries and the test dataset was 2,250 entries with various attributes related to weather, location, and traffic conditions. The traffic volume column contains the target values. Overall, the traffic data indicated significant variability in volume and probe count statistics, with some entries showing minimal traffic and zero probe counts, while others reported volumes as high as 2,816 vehicles per hour and probe counts up to 294 per hour. The average speed was 38.79 miles per hour, but this ranged widely from 2.29 mph to 78.6 mph. The probe penetration rate, which is the percentage of vehicles equipped with probe devices that actively transmit data to traffic monitoring systems, had an averages of 0.0515% and a maximum of 100%. This suggests that in certain areas, all vehicles were equipped with probes, potentially indicating bias in the probe data collection process.

Temporal aspects showed that the average hour recorded was close to noon (11.70), with data evenly distributed across all hours. the most frequent day of the week recorded was Wednesday, accounting for 30% of the data, indicating a potential weekday bias. The dataset spans the first six months of year 2019. The average temperature was around 58.74°F, with a standard deviation of 14.84°F, indicating a significant variation in temperature. Wind speeds averaged 6.68 mph, with a range from 0.1 mph to 20.4 mph. Precipitation levels were generally low, with a mean of 0.004 inches, and a maximum of 0.27 inches. Notably, the dataset did not record any snow.

Social vulnerability dataset

The Centers for Disease Control and Prevention (CDC) in United States (U.S.) provides social vulnerability data that identifies demographic characteristics that may increase the vulnerability of U.S. communities to natural or

	Traffic volume (vph)	Probe count (vph)	Average speed (mph)	Probe count last week (vph)	Temperature (°F)	Wind speed (mph)
Mean	205.54	15.01	38.79	14.37	58.74	6.68
Std	308.8	28.47	12.94	27.56	14.84	3.24
Min	0.5	0	2.29	0	25.4	0.1
25th percentile	21.5	0	30	0	47	4.4
50th percentile	81	3	37.68	3	59.7	6.1
75th percentile	261	15	47	15	68.4	9
Max	2816.00	294	78.6	338	92.2	20.4

Table 2. Descriptive statistics of traffic volume dataset.

human-made disasters[45]. This data is collected for all U.S. census tracts, which are counties' subdivisions used for census purposes, and include social attributes such as racial and ethnic minority, unemployment level, high poverty, and disability. Table 3 provides a complete list of the tracked demographic information. This data includes the raw values, percentages, and percentile estimates for each demographic attribute. Because our case study's data points are spatially distributed in the state of Tennessee and do not inherently include demographic data, we mapped our case study's data points to the appropriate CDC census tracts in Tennessee and associated them with corresponding demographic features. We then used the CDC's demographic data as protected attributes to measure the bias of the tree-based ensemble ML models.

Bias testing

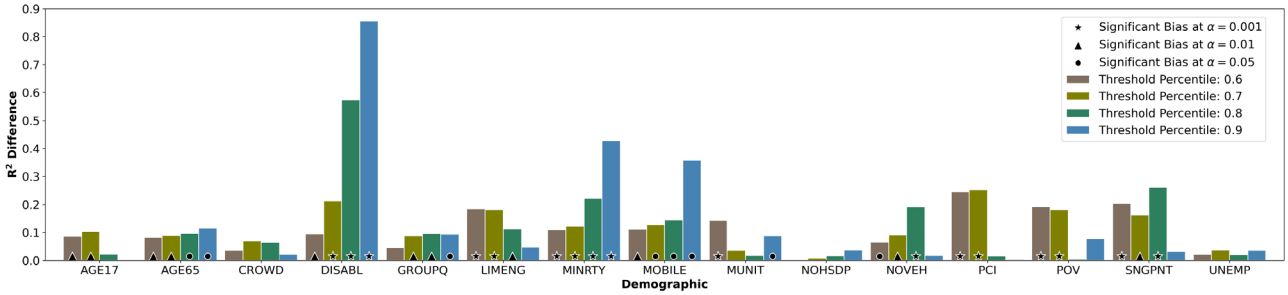
To measure the fairness of the tree-based ensemble regression models for traffic volume prediction, we used the overall accuracy equality fairness metric, which states that the overall prediction accuracy for the protected and not protected groups must be equal[34]. In our case, we wanted to determine if the volume estimation ML models were as accurate in areas with high vulnerability scores as in regions with low vulnerability scores. The CDC defines an area as highly vulnerable with respect to a particular demographic, d, if that demographic's percentile is greater than 0.9. We extended this definition of vulnerability to four cutoff percentiles 0.6, 0.7, 0.8 and 0.9 in order to understand how the model bias changed for different vulnerability levels. The highly vulnerable areas were then considered to be part of the protected group. To determine the accuracy of the ML models for the protected and non-protected groups, we used the coefficient of determination or R^2, which is a value between 0 and 1 and measures how well the observed outcomes are reproduced by a model based on the proportion of the variation in the output-dependent variable that is predictable from the input. We then measured bias as the difference in R^2 accuracy between the protected group (highly vulnerable areas) and non-protected group (low vulnerability areas), in which a positive R^2 difference of value v means that the model's R^2 was v higher for non-protected compared to the protected group. We combined this metric with a bootstrap based significance test to determine model bias, as described in section "Bias testing".

Figure 1 presents the results of our bias testing method and illustrates the measured bias for the different demographic features collected by the CDC, as described in Table 3. The brown color bars show the models' R^2 differences between areas with a percentile less than 0.6 for a particular demographic feature and areas with percentiles greater or equal to 0.6. The green-yellow color bars identifies the measured R^2 difference between regions with a demographic characteristic percentile that is $\geq$ 0.7% and regions where the percentile is $\leq$ 0.7%. The green color bars are for the R^2 difference when areas with 0.8 percentile or higher are considered as part of the protected group, while the blue bars represent the R^2 difference when locations with 0.9 or higher percentiles for demographics are considered part of the protected group. The star, triangle and circle symbols at the bottom of the bars indicate if the R^2 difference was statistically significant at the 0.001, 0.01 or 0.05 α-level respectively. From Fig. 1, we observed that the disability (DISABL) and minority (MINRTY) features showed the highest bias between the protected and non-protected group for all three tree-based models, while the no high school diploma (NOHSDP) and unemployed (UNEMP) attributes had the lowest.

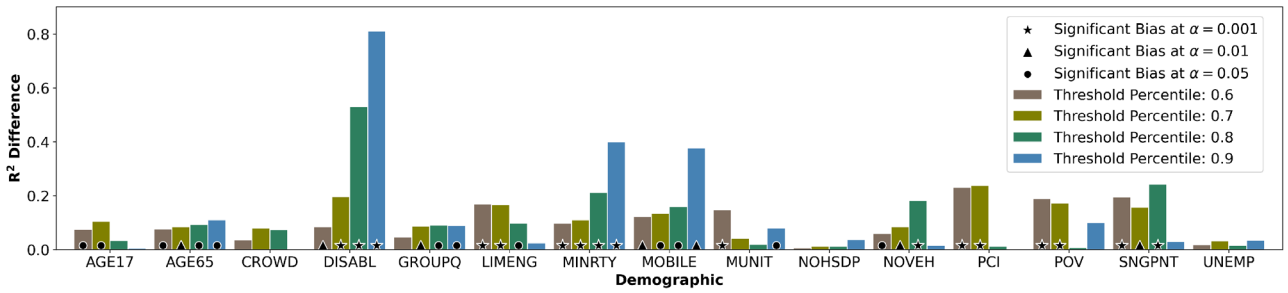
In Table 4, we present a statistical summary of the four demographic groups from the CDC dataset that demonstrate the highest bias across all thresholds. The data for individuals with disabilities (DISABL) is the most centrally distributed, with a mean value of 0.48. Conversely, the data for mobility-impaired (MOBILE) and single-parent households (SNGPNT) exhibit a skew towards lower values. The minority group (MINRTY), on the other hand, shows a higher concentration at the upper end of the distribution, indicating that the dataset contains more areas with high minority percentiles. In contrast, it contains fewer areas with high concentrations of mobility-impaired and single-parent households, while the distribution of individuals with disabilities remains relatively even.

CDC Variable Name	Description
AGE17	Persons aged 17 and younger
AGE65	Persons aged 65 and older
CROWD	At household level (occupied housing units), more people than rooms
DISABL	Civilian non-institutionalized population with a disability
GROUPQ	Persons in group quarters
LIMENG	Persons (age 5+) who speak English "less than well"
MINRTY	Minority (all persons except white, non-Hispanic)
MOBILE	Mobile homes
MUNIT	Housing in structures with 10 or more units estimate
NOHSDP	Persons (age 25+) with no high school diploma
NOVEH	Households with no vehicle available
PCI	Per capita income
POV	Persons below poverty
SNGPNT	Single parent household with children under 18
UNEMP	Civilian (age 16+) unemployed

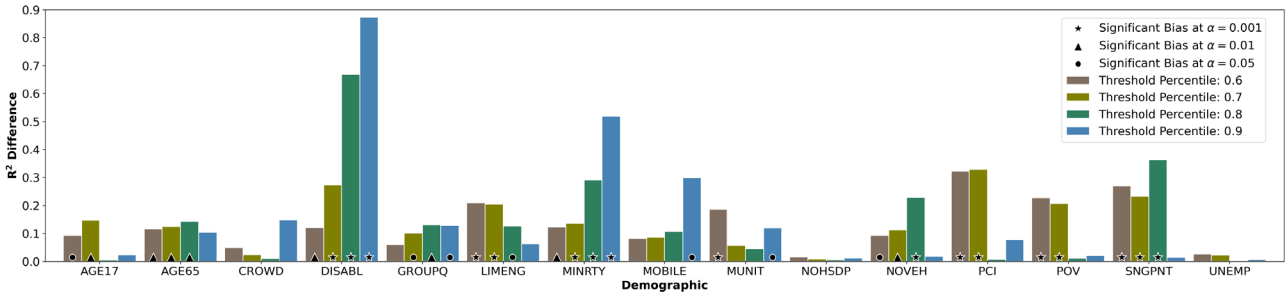
Table 3. Descriptions for the 15 demographic variables provided by the CDC.



(a) Bias testing for XGBoost model



(b) Bias testing for GBT model



(c) Bias testing for RF model

Figure 1. Bar graph of the demographic bias tests for the 15 CDC demographic variables for the XGB model (**a**), GBT model (**b**), and RF model (**c**). The height of each bar represents the difference between R^2 for areas where a specific demographic d's percentile is greater than p and areas where the percentile is less than p, with $p \in 0.6, 0.7, 0.8, 0.9$. The symbol at the bottom of the bar represents the alpha-level for which the model showed significant bias in that demographic variable and percentile cutoff: the star symbol indicates the highest statistical significance, the triangle represents medium significance, and the circle denotes the lowest significance.

	DISABL	MINRTY	MOBILE	SNGPNT
Mean	0.48	0.58	0.30	0.37
Std	0.24	0.26	0.28	0.28
Min	0.04	0.02	0.00	0.04
25th percentile	0.30	0.37	0.00	0.12
50th percentile	0.48	0.61	0.31	0.30
75th percentile	0.65	0.77	0.46	0.56
Max	0.96	0.95	0.97	0.97

Table 4. Demographic descriptive statistics.

When testing for bias and selecting a demographic attribute for focused analysis, we considered both the magnitude of the bias and its statistical significance. Although the disability (DISABL) attribute displayed the highest values for bias, the statistical significance, shown in Fig. 1 using a star, triangle or circle at the bottom of each bar, of the DISABL attribute was reduced for the 0.6 percentile for all three models. In contrast, the minority feature presented the second highest bias values, which were also shown to have the highest statistical significance in most cases across all three models, except for the RF model at the 0.6 percentile. Thus, our numerical case study concentrated on addressing the models' bias in relation to the minority attribute. For simplicity of presentation, our results also concentrated on areas with minority population percentile ≥ 0.8 and those with minority percentage ≥ 0.9, because, as shown in Fig. 1, they exhibited the highest bias as measured by R^2 difference. However, the methodology we developed can be applied to address bias for any other demographic feature or percentile cutoff.

Bias mitigation

To make the models' performances fair, we modified their loss function by adding a regularization term that correlates the model's error to the minority attributes. We considered three correlations terms: Pearson correlation coefficient, Kendal's Tau, and distance correlation. We also included a parameter, γ, with values between 0 and 1 as the coefficient in front of the regularization expression to indicate the significance given to addressing bias during training. That is, when $\gamma = 0$, the regularization term was ignored and the models were trained to maximize accuracy, and when $\gamma = 1$, the models' training focused on just minimizing the bias in the model. Section "Bias mitigation" describes the bias mitigation approach in detail.

Figure 2 shows the impact of the three correlation terms, Pearson's coefficient, Kendall's tau and distance correlation, on the bias of the RF, GBT, and XGBoost models as γ increases from 0 to 1, and for 0.8 and 0.9 minority attribute percentile cutoffs. Of all the three models, the RF showed the highest bias per percentile limit (Fig. 2a,b), followed by the XGB (Fig. 2c,d), while the GBT models had the lowest R^2 difference (Fig. 2e,f). In addition, Kendall's tau correction term had a higher impact on the RF model compared to the other models, and started reducing RF's bias at $\gamma \geq 0.5$. The Pearson's coefficient and distance correlation had similar performance across models and significantly reduced the R^2 difference when $\gamma \geq 0.9$.

Figure 3 shows a zoomed-in version of Fig. 2, where we focus on the results for $0.9 \leq \gamma \leq 1$. As depicted, for the RF model, the Kendall's tau correction was able to reduce the bias from 0.5 to about 0.25 for areas with a minority percentile of 0.9 or higher (Fig. 3b) and from 0.3 to about 0.1 for the 0.8 minority percentile or higher (Fig. 3a). For the GBT and XGB models, Pearson's coefficient and distance correlation performed better than Kendall's tau for $\gamma > 0.99$. As shown in Fig. 3c–f, Pearson's coefficient and distance correlation reduced GBT and XGB bias from 0.22 to a little over 0.1 for the percentile limit of 0.8, and from 0.42 to a little over 0.2 for a percentile cutoff of 0.9.

Figure 4 represents the models' R^2 test performances when their loss functions were modified with correlation terms and as γ varied from 0 to 1. From this figure's sub-graphs (a), (b), and (c), we observed that the R^2 scores of the XGB and GBT models were higher than the RF's performance for all correction terms and for all γ's, and

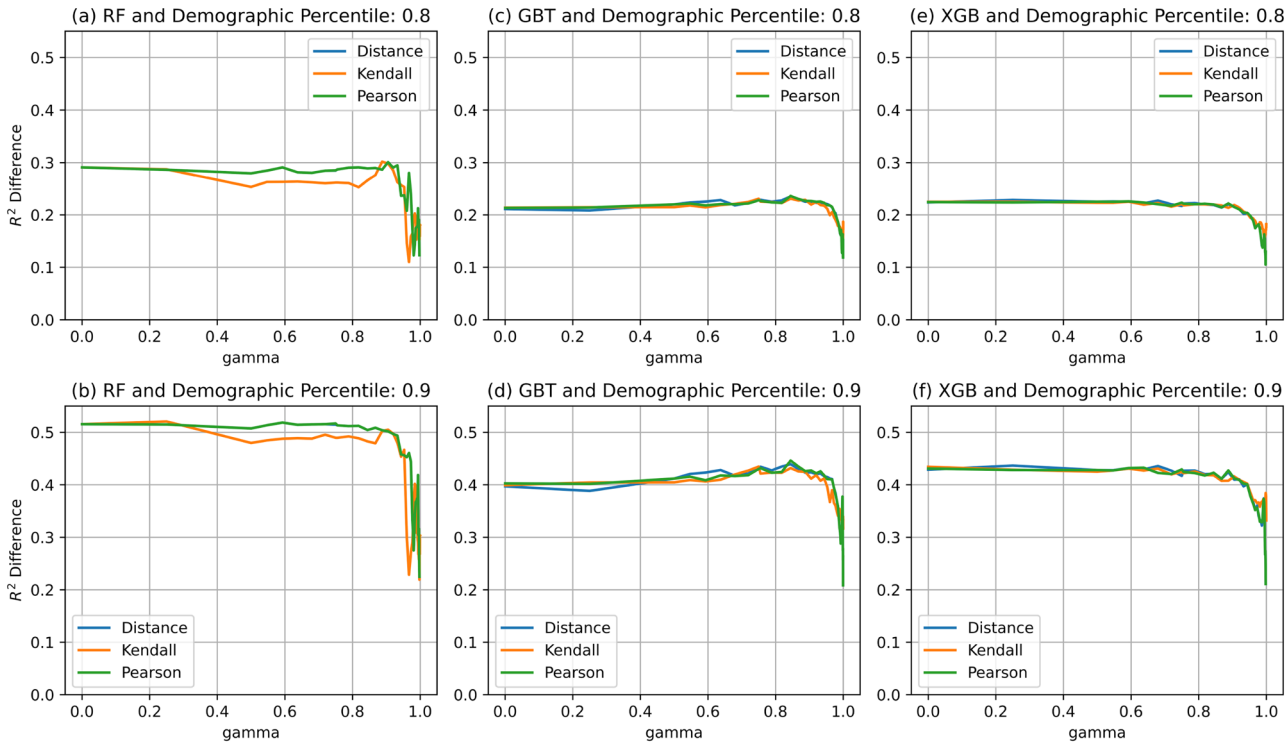


Figure 2. Bias mitigation with Pearson's coefficient, Kendall's tau, and distance correlation for RF, GBT, and XGBoost models for the minority attribute at different percentile levels.

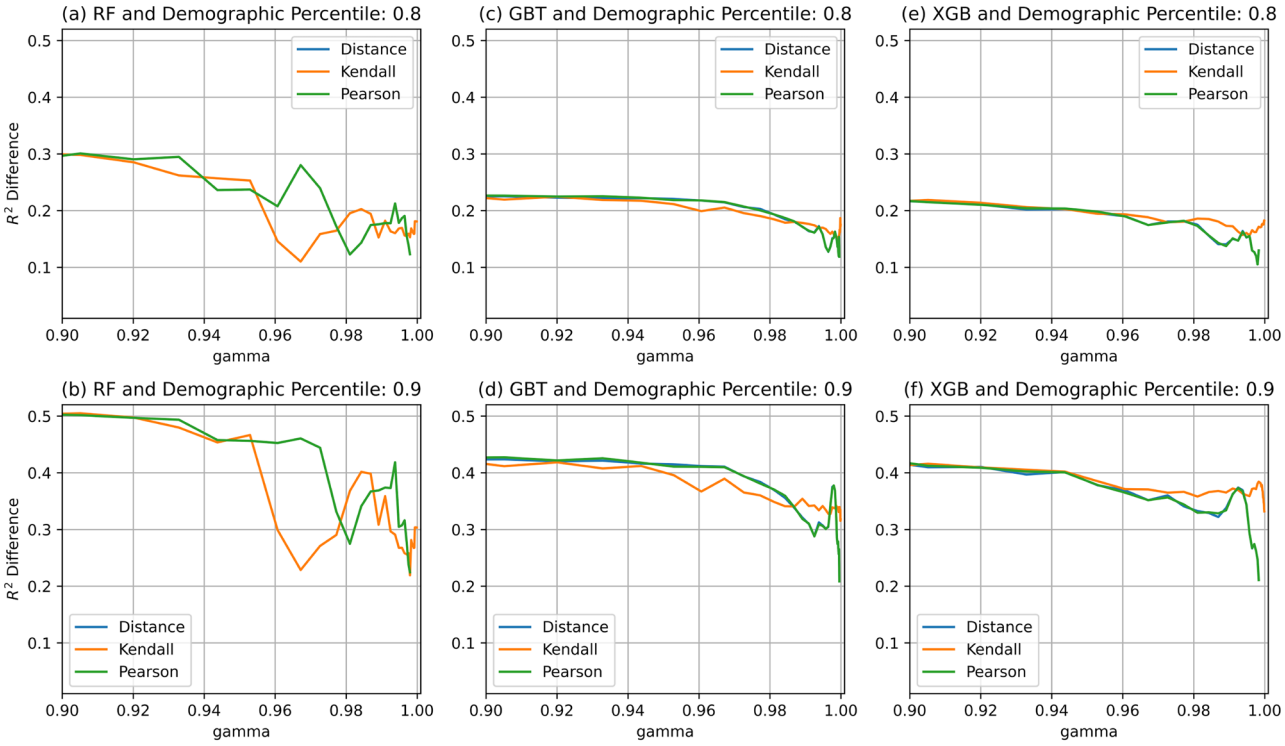


Figure 3. Zoomed-in bias mitigation with Pearson's coefficient, Kendall's tau, and distance correlation for RF, GBT, and XGBoost models for the minority attribute at different percentile levels.

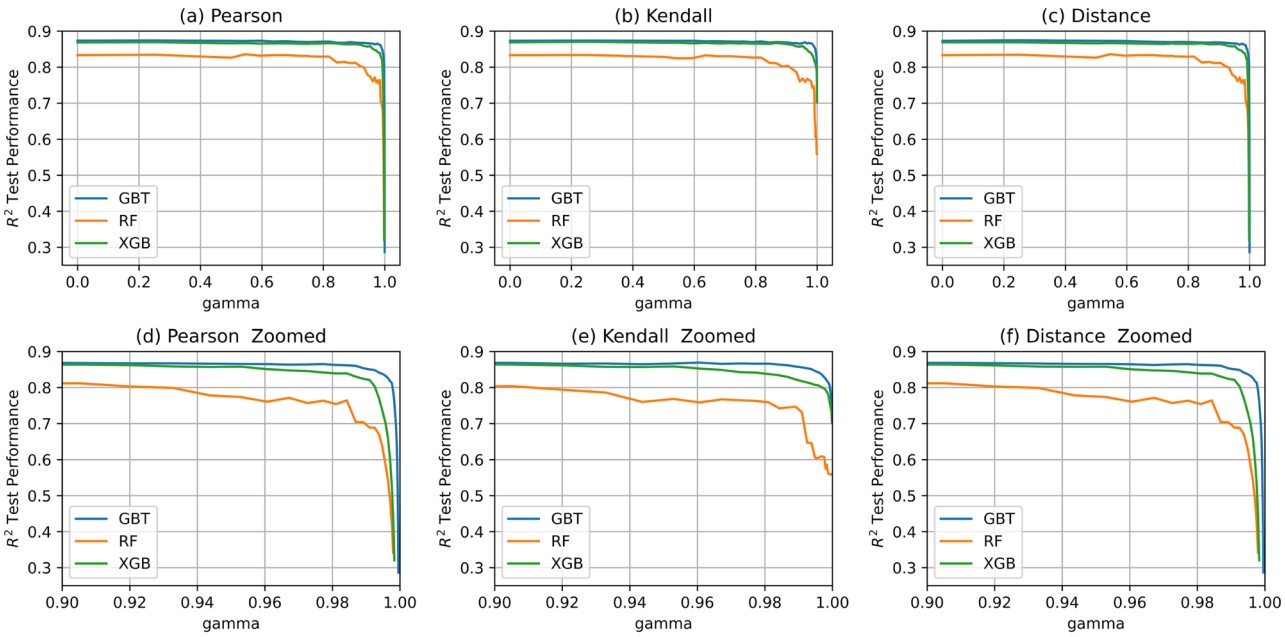


Figure 4. Test R^2 performance for Pearson's coefficient, Kendall's tau, and distance correlation for RF, GBT, and XGBoost models for the minority attribute at different percentile levels.

that R^2 scores for all models remained relatively constant up until γ was close to 1. The RF's R^2 accuracy started reducing at $\gamma \geq 0.8$, while for the XGBoost and GBT models, the R^2 score decreased when $\gamma \geq 0.9$. Figure 4d–f shows the zoomed-in behavior of the R^2 for all models and all correction terms when $0.9 \leq \gamma \leq 1$. As depicted, the R^2 score for all models decreased to about 0.3 (30%) as γ approached 1 when their loss functions were combined with Pearson's coefficient and distance correlation correction terms, while Kendall's tau maintained the models' performances greater than 0.55 (55%).

From Fig. 4, we see that there is a trade-off between the models' accuracy in terms of test R^2 score, reduction in performance bias, and model type. That is, as the γ parameter put more weight on reducing bias, the overall accuracy reduced at different rates depending on the model and on the correction term. Hence, depending on the application, an end user will need to decide how much model accuracy they are willing to lose to achieve a desired bias reduction. This in turn would dictate which model to pick, what correlation term to select, and what γ value to use. As an example, for our case study, we found that for percentile cutoff of 0.8: for the RF model, Kendall's tau correction achieved the minimum bias that equaled 0.11, which matched a model accuracy of $R^2 = 0.767$; the Pearson's coefficient reduced bias the most for XGB model, with the smallest bias equaled 0.137, and an $R^2 = 0.824$; and for the GBT model, the distance correction term performed the best, with a minimum bias of 0.127 and a $R^2 = 0.826$. If a user wanted to minimize the bias the most, they would choose the RF model with the Kendall's tau correction and $\gamma = 0.967$. However, if they wanted to reduce bias while maintaining an accuracy of $R^2 \geq 0.8$, they could choose the XGB with $\gamma = 0.989$ or the GBT with $\gamma = 0.996$.

Discussion

The previous section shows that our approach is effective at considerably reducing the performance bias for tree-based ensemble regression models while maintaining high model accuracy. For the RF model, the R^2 bias was halved for all minority percentage cutoffs at $\gamma \geq 0.96$. For the XGBoost and GBT models, the difference in R^2 reduced by 50% at $\gamma \geq 0.98$. XGBoost and GBT models were less sensitive to the correction terms than the RF model, but also exhibited higher accuracy and lower bias compared to RF.

To understand why the tuning parameter γ had to be high to influence the models' bias (at least ≥ 0.5 for RF and ≥ 0.9 for GBT and XGBoost), we did an analysis where we tracked the mean squared error (MSE) and correction term values for each training round for all the models. We found that the correction term values were much smaller (generally less than 1) in magnitude compared to the MSE values (generally in the thousands), and thus require more weight from γ to impact the bias in the model. However, we also note that the tuning of γ is dependent on the a model's dataset. If the training data has very low MSE values, then low γ values will likely start reducing the model's bias.

We also note our method, which only involves modifying the loss function, is applicable to any models, such as neural networks and Gaussian processes, that are trained by minimizing a differentiable loss function. This makes our method easy to incorporate into many existing ML models that are currently in use.

Methods

Problem formulation

We considered a standard regression problem, in which, given n input instances $x_i \in \mathbb{R}^d$ with d features, and n target variables $y_i \in \mathbb{R}$, we sought to learn a function to predict the target variables. This was done by minimizing a loss function $L(y_i, \hat{y}_i)$, where $\hat{y}_i \in \mathbb{R}$ was the predicted variable. The MSE is typically used as the loss function and is defined as:

$$\text{MSE}(y_i, \hat{y}_i) = \frac{1}{2}\sum_i (y_i - \hat{y}_i)^2 \tag{1}$$

To account for possible bias in the model, we also considered a set of protected demographic attributes z_i, where z_i is the demographic attribute for input x_i and is a continuous random variable. We modified the loss function by adding a regularization term that measured how the model's error correlated to the protected attribute:

$$L_c(y_i, \hat{y}_i) = (1-\gamma)L_\theta(y_i, \hat{y}_i) + n\gamma\, c(e, z)^2 \tag{2}$$

In Eq. (2), z is the variable for the value of the demographic of interest, $\gamma \in [0, 1]$ is a tunable regularization parameter, and c is correlation measurement function. Finally, e is some measure of the prediction error, such as: $e = \hat{y}_i - y_i$, $e = |\hat{y}_i - y_i|$ or $e = (\hat{y}_i - y_i)^2$.

Bias testing

The CDC defines a community to have a high level of vulnerability vis-a-vis a particular demographic if the community's percentile for that demographic is greater or equal to 0.9[46]. We use "group fairness" to measure the fairness of our models' predictions. We define unbiased predictions based on the model accuracy of predictions within the high-vulnerability group and the non-vulnerable group. Following this definition, we define an unbiased model based on the model accuracy above and below a demographic percentile cutoff.

Definition: Unbiased Model 1

Let R_0^2 be the model R^2 on observations with demographic percentage $< c$ for some cutoff percentile c and R_1^2 be the same with demographic percentage $\geq c$. Then a model is defined to be unbiased at cutoff c if and only if $R_0^2 = R_1^2$.

Using this definition, we now define a test statistic, s, shown in Eq. (3).

$$s = \begin{cases} R_1^2/R_0^2 & R_1^2 > R_0^2 \\ R_0^2/R_1^2 & R_1^2 \leq R_0^2 \end{cases} \tag{3}$$

We define s to be piece-wise so that it can by used for a one-sided hypothesis test, because, according to our definition of an unbiased model, we can now define a model as unbiased if and only if $s = 1$. Thus, the natural hypothesis test for bias becomes Eq. (4).

$$H_0 : s = 1 \qquad H_1 : s > 1 \tag{4}$$

Because we do not know the sampling distribution of s, we look at a second definition of an unbiased model.

Definition: Unbiased Model 2

Let e be errors in model predictions and let d be the demographic percentile variable. Then we define a model to be unbiased if and only if $e \perp\!\!\!\perp d$.

Using the second definition, we reformulate the Hypothesis test as a Bayesian hypothesis test. To do this, let s_d be the s statistic calculated using the demographic cutoff and define p according to Eq. (5). Then we reject $H_0 : e \perp\!\!\!\perp d$ if $p < \alpha$ for some chosen level alpha.

$$p = \mathbb{P}(s \geq s_d | e \perp\!\!\!\perp d) \tag{5}$$

To compute this probability, we use Bootstrap resampling to approximate the sampling distribution of s. To do this, let n_0 be the number of low-vulnerability points, let n_1 be the number of high-vulnerability points, and let n_B be the number of Bootstrap iterations.

1. Sample n_0 prediction/ target pairs with replacement from the entire dataset and calculate $R^2_{0,i}$.
2. Sample n_1 prediction/ target pairs with replacement from the entire dataset and calculate $R^2_{1,i}$.
3. Calculate s_i from $R^2_{0,i}$ and $R^2_{1,i}$.
4. Repeat n_B times.

Because we are sampling from the entire dataset independent of the demographic variable, this will result in a set $\mathcal{S}$ of s_i values from bootstrapping that approximates the sampling distribution of s under the null hypothesis. Then we calculate p according to Eq. (6).

$$p = \frac{\#\{s_i | s_i \in \mathcal{S}, s_i \geq s_d\}}{n_B} \tag{6}$$

Finally, because we are testing bias in 15 demographics at 4 cutoff levels, we correct the p-values using the Benjamini and Hochberg FDR correction for multiple testing[47].

Bias mitigation

In order to mitigate the bias present, we augment the loss function with a correlation penalty term according to Eq. (2).

In this work, we only consider the MSE initial loss function; however, this method can be extended to any arbitrary initial loss function. For the correlation function, we investigate three measures of correlation: Pearson's r, Kendall's τ, and distance correlation.

Pearson's r measures linear correlation and is defined by Eq. (9).

$$s_x = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x - \overline{x})^2} \tag{7}$$

$$s_y = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(y - \overline{y})^2} \tag{8}$$

$$r(x, y) = \frac{1}{n-1}\sum_{i=1}^{n}\left(\frac{x_i - \overline{x}}{s_x}\right)\left(\frac{y_i - \overline{y}}{s_y}\right) \tag{9}$$

Kendall's τ measures rank correlation and is defined by Eq. (10), where sgn is the sign function. Because the sign function is not differential at 0 and trivially differentiable elsewhere, we use a sigmoid approximation, $\widetilde{\tau}$, for our augmented loss function defined in Eq. (11) where $\sigma(x)$ is the sigmoid function.

$$\tau = \frac{2}{n(n-1)}\sum_{i=1}^{n}\sum_{j=i}^{n}\text{sgn}(x_i - x_j)\text{sgn}(y_i - y_j) \tag{10}$$

$$\widetilde{\tau}(e, d) = \frac{2}{n(n-1)}\sum_{i=1}^{n}\sum_{j=i}^{n}\sigma(e_i - e_j)\text{sgn}(d_i - d_j) \tag{11}$$

Distance correlation, dCor, measures the statistical dependence of two variables and is defined according to Eq. (12).

$$
\begin{aligned}
a_{jk} &= \left\|x_j - x_k\right\| \quad b_{jk} = \left\|y_j - y_k\right\| \quad \forall j,k = 1,\ldots,n \\
A_{jk} &= a_{jk} - \bar{a}_{j\cdot} - \bar{a}_{\cdot k} + \bar{a}_{\cdot\cdot} \qquad B_{jk} = b_{jk} - \bar{b}_{j\cdot} - \bar{b}_{\cdot k} + \bar{b}_{\cdot\cdot} \\
\mathrm{dCov}(x,y) &= \frac{1}{n^2}\sum_{j=1}^{n}\sum_{k=1}^{n} A_{jk}B_{jk} \qquad \mathrm{dVar}(x) = \mathrm{dCov(x,x)} \\
\mathrm{dCor}^2(x,y) &= \frac{\mathrm{dCov}^2(x,y)}{\sqrt{\mathrm{dVar}(x)\mathrm{dVar}(y)}}
\end{aligned}
\tag{12}
$$

Bias mitigation framework implementation

To implement our approach, we selected the XGBoost library[39], as it offers a highly flexible tree-based model that can be restricted to behave similarly to traditional RF and GBT models. Additionally, unlike other popular ML libraries like Scikit-Learn[48], XGBoost allows the user to input custom loss functions so that we could test our bias correction regularization terms.

We provide two wrappers around the XGBoost model that restrict its flexibility to behave like an RF model and GBT model[40]. The RF model removes all explicit regularization parameters, prohibits boosting by setting the number of boosting rounds and learning rate to 1, and asserts that there is data subsampling and column sampling either by node, by level, or by tree . The GBT model removes L^1 and L^2 regularization, removes pruning, allows column sampling by tree only, and fixes the grow policy to depth-wise and sampling method to uniform. All remaining parameters were set to the defaults used in Scikit-Learn's RandomForestRegressor and GradientBoostingRegressor, respectively. Table 5 shows the tuning parameter ranges for each model type.

We also provide functions to create our correlation regularized loss functions[40]. These functions compute the gradient and hessian of the correction term and combine them into the augmented loss function shown in Eq. (2). In order to speed up the run time, particularly for the distance correlation and Kendall's τ metrics, we implemented the correlation term calculations in C++ and interfaced them with Python using the ctypes library. The gradient and hessian calculations for each correlation metric are shown in Appendix A.

Conclusion

In summary, we present an effective and easy to apply methodology for addressing bias in tree-based bagging and boosting ML models for regression, that is also suitable for any ML models trained through minimization and differentiation. This is to fill a gap in literature, where most work in AI bias mitigation has mainly focused on classification problems. Our bias correction approach involves adding a regularization term to an ML model's loss function that penalizes correlation between the model's error and membership in a protected/vulnerable group.

We implemented three regularization terms: Pearson's coefficient, Kendall's tau, and distance correlation. We demonstrated our technique by applying it to three popular tree-based ensemble regression models: RF, GBT, and XGboost. We leveraged the XGBoost library flexibility to use custom loss function and to represent the RF and GBT models. To exemplify our approach, we applied it to RF, GBT, and XGboost models trained to predict traffic volume for roads in Hamilton County, Tennessee. Through rigorous statistical testing, we established that the tree-based models exhibited high performance bias in regards to the minority attribute. Our numerical results demonstrated that our bias mitigation methodology could reduce the models' bias toward areas with the highest minority density by as much as 50%.

Most bias mitigation research focuses on ML models that make predictions based on protected characteristics. Our traffic volume prediction case study shows that ML models can still be biased even without direct inclusion of protected attributes. This emphasizes that ML models can exhibit unfairness even when protected attributes are not included in the training process. Therefore, our bias testing and mitigation methodology is also suitable for regression models where protected attributes are aggregate quantities indirectly related to regression.

Future work includes investigating the combination of pre-processing methods, such as up-sampling, with our methodology to further reduce bias in regression models. Another future research topic will be studying how

Parameter	XGB range	RF range	GBT range
Boost rounds	100–1000	1	100–1000
Parallel trees	1	100–1000	1
Max depth	2–10	2–10	2–10
Learning rate	0.0001–1	1	0.0001–1
Min split loss	0.001–1	0.001–1	0
Min child weight	1–10	1–10	1–10
Subsample	0.5–1	0.5–0.999	0.5–1
Colsample by tree	0.5–1	0.5–0.999	0.5–1
L2 Regularization	1	0.00001	0

Table 5. Hyperparameter ranges for each model type.

to address bias for multiple demographics simultaneously, as reducing a model's bias for one protected group could increase the model's bias toward another protected group that was not considered.

Data availability

The CDC data used to support this study is available online on the link in[45]. The traffic data used to train the machine learning models cannot be shared publicly as it belongs to the Chattanooga Department of Transportation and requires signing a data agreement to access it.

Received: 8 May 2024; Accepted: 29 July 2024
Published online: 08 August 2024

References

1. Angwin, J., Larson, J., Mattu, S. & Kirchner, L. Machine bias. In *Ethics of Data and Analytics* 254–264 (Auerbach Publications, 2016).
2. Datta, A., Tschantz, M. C. & Datta, A. Automated experiments on ad privacy settings: A tale of opacity, choice, and discrimination. *arXiv preprint*arXiv:1408.6491 *(2014).*
3. Wilson, B., Hoffman, J. & Morgenstern, J. Predictive inequity in object detection. *arXiv preprint*arXiv:1902.11097 *(2019).*
4. Fitzsimons, J., Al Ali, A. R., Osborne, M. & Roberts, S. A general framework for fair regression. *Entropy* **21**(8), 741 (2019).
5. Breiman, L. Random forests. *Mach. Learn.* **45**, 5–32 (2001).
6. Friedman, J. H. Stochastic gradient boosting. *Comput. Stat. Data Anal.* **38**(4), 367–378 (2002).
7. Fernández-Delgado, M., Cernadas, E., Barro, S. & Amorim, D. Do we need hundreds of classifiers to solve real world classification problems?. *J. Mach. Learn. Res.* **15**(1), 3133–3181 (2014).
8. Chen, T. & Guestrin, C. XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 785–794 (2016).
9. Becker, D. XGBoost. https://www.kaggle.com/code/dansbecker/xgboost (2016).
10. Luong, B. T., Ruggieri, S. & Turini, F. k-NN as an implementation of situation testing for discrimination discovery and prevention. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 502–510 (2011).
11. Kamiran, F. & Calders, T. Data preprocessing techniques for classification without discrimination. *Knowl. Inf. Syst.* **33**(1), 1–33 (2012).
12. Belitz, K. & Stackelberg, P. E. Evaluation of six methods for correcting bias in estimates from ensemble tree machine learning regression models. *Environ. Model. Softw.* **139**, 105006 (2021).
13. Kamiran, F., Calders, T. & Pechenizkiy, M. Discrimination aware decision tree learning. In *2010 IEEE International Conference on Data Mining*, pp. 869–874 (IEEE, 2010).
14. Abebe, S. A., Lucchese, C. & Orlando, S. EIFFFeL: Enforcing fairness in forests by flipping leaves. In *Proceedings of the 37th ACM/SIGAPP Symposium on Applied Computing*, pp. 429–436 (2022).
15. Aghaei, S., Azizi, M. J., & Vayanos, P. Learning optimal and fair decision trees for non-discriminative decision-making. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33, pp. 1418–1426 (2019).
16. Zafar, M. B., Valera, I., Rodriguez, M. G. & Gummadi, K. P. Fairness beyond disparate treatment and disparate impact: Learning classification without disparate mistreatment. In *Proceedings of the 26th International Conference on World Wide Web*, pp. 1171–1180 (2017).
17. Kamishima, T., Akaho, S. & Sakuma, J. Fairness-aware learning through regularization approach. In *2011 IEEE 11th International Conference on Data Mining Workshops*, pp. 643–650 (IEEE, 2011).
18. Calders, T., Karim, A., Kamiran, F., Ali, W. & Zhang, X. Controlling attribute effect in linear regression. In *2013 IEEE 13th International Conference on Data Mining*, pp. 71–80 (IEEE, 2013).
19. Agarwal, A., Dudík, M. & Wu, Z. S. Fair regression: Quantitative definitions and reduction-based algorithms. In *International Conference on Machine Learning*, pp. 120–129 (PMLR, 2019).
20. Komiyama, J., Takeda, A., Honda, J. & Shimao, H. Nonconvex optimization for regression with fairness constraints. In *International Conference on Machine Learning*, pp. 2737–2746 (PMLR, 2018).
21. Kusner, M. J., Loftus, J., Russell, C. & Silva, R. Counterfactual fairness. *Adv. Neural Inf. Process. Syst.***30** (2017).
22. Berk, R., Heidari, H., Jabbari, S., Joseph, M., Kearns, M., Morgenstern, J., Neel, S. & Roth, A. A convex framework for fair regression. *arXiv preprint*arXiv:1706.02409 *(2017).*
23. Raff, E., Sylvester, J. & Mills, S. Fair forests: Regularized tree induction to minimize model bias. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, pp. 243–250 (2018).
24. Iosifidis, V., Fetahu, B. & Ntoutsi, E. Fae: A fairness-aware ensemble framework. In *2019 IEEE International Conference on Big Data (Big Data)*, pp. 1375–1380 (IEEE, 2019).
25. Bhargava, V., Couceiro, M. & Napoli, A. Limeout: An ensemble approach to improve process fairness. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 475–491 (Springer, 2020).
26. Beutel, A., Chen, J., Doshi, T., Qian, H., Woodruff, A., Luu, C., Kreitmann, P., Bischof, J. & Chi, E. H. Putting fairness principles into practice: Challenges, metrics, and improvements. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, pp. 453–459 (2019).
27. Jiang, R., Pacchiano, A., Stepleton, T., Jiang, H. & Chiappa, S. Wasserstein fair classification. In *Uncertainty in Artificial Intelligence*, pp. 862–872 (PMLR, 2020).
28. Zafar, M. B., Valera, I., Gomez-Rodriguez, M. & Gummadi, K. P. Fairness constraints: A flexible approach for fair classification. *J. Mach. Learn. Res.* **20**(75), 1–42 (2019).
29. Di Stefano, P. G., Hickey, J. M. & Vasileiou, V. Counterfactual fairness: Removing direct effects through regularization. *arXiv preprint*arXiv:2002.10774 *(2020).*
30. Dwork, C., Hardt, M., Pitassi, T., Reingold, O. & Zemel, R. Fairness through awareness. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*, pp. 214–226 (2012).
31. Garg, S., Perot, V., Limtiaco, N., Taly, A., Chi, E. H. & Beutel, A. Counterfactual fairness in text classification through robustness. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, pp. 219–226 (2019).
32. Kearns, M., Neel, S., Roth, A. & Wu, Z. S. Preventing fairness gerrymandering: Auditing and learning for subgroup fairness. In *International Conference on Machine Learning*, pp. 2564–2572 (PMLR, 2018).
33. Hort, M., Chen, Z., Zhang, J. M., Harman, M. & Sarro, F. Bias mitigation for machine learning classifiers: A comprehensive survey. *ACM J. Responsible Comput.* (2023).
34. Verma, S. & Rubin, J. Fairness definitions explained. In *2018 IEEE/ACM International Workshop on Software Fairness (Fairware)*, pp. 1–7 (IEEE, 2018).
35. Kim, J.-Y. & Cho, S.-B. An information theoretic approach to reducing algorithmic bias for machine learning. *Neurocomputing* **500**, 26–38 (2022).

36. Ghassami, A. E., Khodadadian, S. & Kiyavash, N. Fairness in Supervised Learning: An Information Theoretic Approach (2018). arXiv:1801.04378 [cs, math, stat].
37. Madrid, V. F. An Information Theoretic Approach for Fair Machine Learning.
38. Rathore, S. & Brown, S. M. Information Theoretic Framework For Evaluation of Task Level Fairness (2022).
39. Chen, T. & Guestrin, C. XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, pp. 785–794 (ACM, 2016).
40. Bensen, E., Severino, J. & Ugirumurera, J. Fair bagging and boosting models. https://github.com/NREL/Fair_Bagging_Boosting_Models (2024).
41. Hou, Y., Young, S. E., Dimri, A. & Cohn, N. Network scale ubiquitous volume estimation using tree-based ensemble learning methods. Technical report, National Renewable Energy Lab. (NREL) (2018).
42. TomTom. Traffic stats (2022). https://www.tomtom.com/products/traffic-stats/, Last accessed on 2022-07-14.
43. Severino, J. *et al.* Real-time highly resolved spatial-temporal vehicle energy consumption estimation using machine learning and probe data. *Transp. Res. Rec.* **2676**(2), 213–226 (2022).
44. Sanyal, J. Regional mobility project meeting (2020). https://www.energy.gov/sites/default/files/2020/06/f75/eems061_sanyal_2020_o_4.27.20_453PM_JL.pdf. Accessed: 2023-09-29.
45. Cdc's social vulnerability index (svi) (2021).
46. Social vulnerability index documentation (2020).
47. Benjamini, Y. & Hochberg, Y. Controlling the false discovery rate: A practical and powerful approach to multiple testing. *J. R. Stat. Soc. Ser. B (Methodol.)* **57**(1), 289–300 (1995).
48. Pedregosa, F. *et al.* Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.* **12**, 2825–2830 (2011).

Acknowledgements

This work was authored by the National Renewable Energy Laboratory, operated by Alliance for Sustainable Energy, LLC, for the U.S. Department of Energy (DOE) under Contract No. DE-AC36-08GO28308. Funding provided by U.S. Department of Energy Vehicle Technology Office. The views expressed in the article do not necessarily represent the views of the DOE or the U.S. Government. The U.S. Government retains and the publisher, by accepting the article for publication, acknowledges that the U.S. Government retains a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this work, or allow others to do so, for U.S. Government purposes. This work was made possible through the close cooperation of the City of Chattanooga Department of Transportation. The authors would also like to acknowledge Jen King for her inspiring encouragement in the exploration of this topic.

Author contributions

The authors confirm contribution to the paper as follows: study conception and design: Juliette Ugirumurera and Jibonananda Sanyal; data collection: Erik A. Bensen, Joseph Severino and Juliette Ugirumurera; model development: Erik A. Bensen, and Joseph Severino; numerical analysis: Joseph Severino, Erik A. Bensen, and Juliette Ugirumurera; interpretation of results: Joseph Severino, Erik A. Bensen, Juliette Ugirumurera, and Jibonananda Sanyal; draft manuscript preparation: Erik A. Bensen, Juliette Ugirumurera, Joseph Severino, and Jibonananda Sanyal. All authors reviewed the results and approved the final version of the manuscript.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1038/s41598-024-68907-5.

Correspondence and requests for materials should be addressed to J.U.