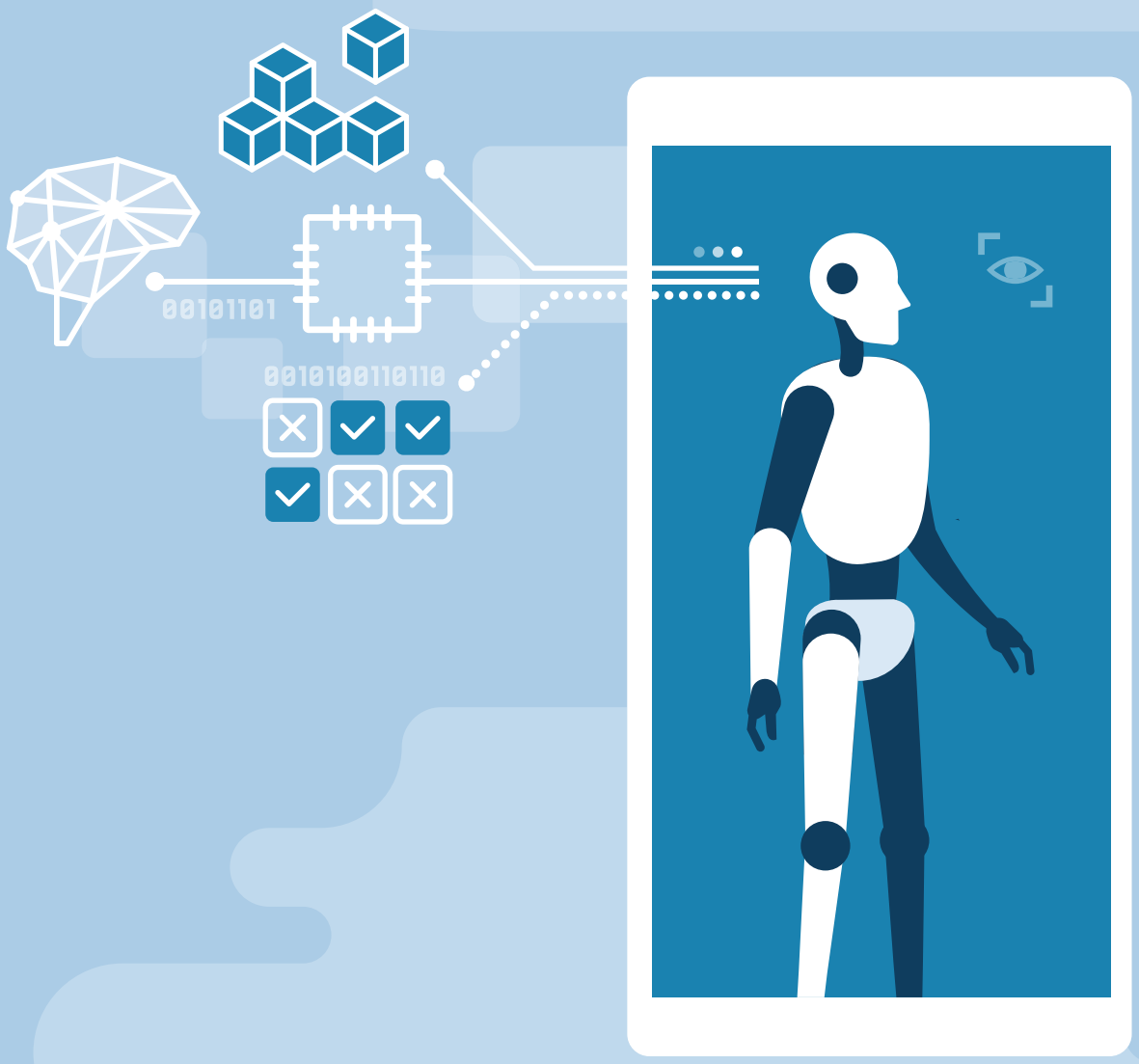
Identifying Bias

in Machine Learning Algorithms:



CLASSIFICATION WITHOUT DISCRIMINATION

"THERE IS A RISK TO EVERY INSTITUTION IN IGNORING THE RISK OF BIAS IN MACHINE LEARNING APPLICATIONS."

BY MANISH KUMAR, PH.D., RAHUL ROY, PH.D., AND KEVIN D. ODEN, PH.D.

Machine learning (ML) advanced statistical and mathematical models are used in various partial or fully automated decision-making systems that affect individual lives. Today, these models are not only increasingly used to make important decisions in our financial lives, ranging from retail (closed and open-end products) and wholesale scorecards (application, behavioral, and collection) but also in other aspects such as granting university admission, social benefit assignment, predicting the risk of criminal recidivism (COMPAS model), and part of hiring tools to review job applicants' resumes. In these applications, models are often built using sensitive drivers, also called attributes, such as age, gender, nationality, religion, race, language, culture, marital status, economic condition, zip code, etc. One of the unintended consequences of lax modeling practice is the potential for bias or unfairness in ML models that accentuates our societal stereotypes and contravenes the laws of many jurisdictions as well.

There are a number of recent high-profile incidents reflecting the risk and the difficulty in detecting bias (or unfairness) in ML models. For example, Apple Credit Card offered smaller lines of credit to women than to men due to its ML/AI credit assessment process. Similarly, there was an AI-based talent-hiring tool at Amazon that was biased against women. And possibly the most publicly embarrassing case occurred when Microsoft's racist chatbot plugged into Twitter.

One of the lessons we should all learn from these mistakes is that if this can happen to the best technology firms in the world, there is a risk to every institution in ignoring the risk of bias in ML applications. Furthermore, all institutions should wonder if there are a number of issues in ML models that have gone unnoticed for quite some time. The risk of such bias in ML models is not only a source of major discomfort to people in minority or disadvantaged groups, but also brings reputational risk and potential legal consequences to institutions utilizing them in their decision-making processes.

This bias challenge is inherent in any model development exercise. However, it is accentuated in the ML model development process, which refines, or "trains," a series of algorithms on a large data set by optimizing iteratively as it learns to identify patterns and make predictions for new data. ML essentially imposes much less structure on how data is interpreted compared to conventional approaches in which programmers impose ex-ante rules to make decisions. This leads to difficulty in attributing the results to particular causes or variables.

In most decision-making applications, the quality of a machine learning model is often evaluated based on various performance metrics such as accuracy and hit ratio (the percentage of correct results), its precision (the ability to not label a negative sample as positive), or its recall (the ability to find all the positive samples), ROC and AUC, as well as numerous other metrics. Additionally, ML models are often trained with a loss function, whose objective, among the many examples of objective function criteria, is to minimize the sum of squared errors or maximize the accuracy. The challenge with such optimizations is that they take advantage of information (especially any unfairness or biases that exist in the input data which improves the error metric on the given dataset. If there are underlying patterns of unfairness and discrimination in the data set, such biases will likely be picked up in the training process. Therefore, serious concerns have been raised about the fairness of ML algorithmic decision-making, which not only has a material impact on business decisions but also ethical and moral implications To improve ML models, the accuracy of the ML output should not be the only criteria when developing models. Various fairness measures and metrics must also be taken into account at relevant stages of ML model development to ensure any biases are eliminated or at least minimized. To date, the research on assessment of ML model bias and corrective processes or actions (i.e., implementing model fairness) can be generally divided into three categories:

On Previous Page: Shutterstock.com

These categories and methods have been applied in other areas such as Natural Language Processing (NLP) and Deep Learning as well to minimize unfairness.

Fair Exploratory Data Analysis and Pre-processing

Pre-processing techniques try to transform the data so that the underlying discrimination is removed.

Fair In-processing

In-processing techniques try to modify and change state-of-the-art learning algorithms to remove discrimination during the model training process.

Fair Post-processing

Post-processing is performed on a trained ML model by accessing a holdout set, which was not involved during the training of the model.

The above three categories and the overall workflow are illustrated in Figure 1.

These categories and the methods developed within them are not just used in ML models, but have been applied in other areas such as Natural Language Processing (NLP) and Deep Learning as well to minimize unfairness.

As just noted, the majority of proposed fair ML approaches focus on fair pre-processing, fair in-processing. or fair post-processing. This article focuses on fair in-processing, as few modelers and validators are familiar with the techniques in this area.

In the jargon of ML, the attributes or variables that can lead to discrimination are called the sensitive attributes. These sensitive attributes can be age, gender, and ethnicity, for example. And within a sensitive attribute, there may be one or more protected classes. For instance, old and young may be protected classes in the age attribute. It is natural to think that by removing these sensitive attributes from the ML model development process you can eliminate unfairness in the model. However, it is well known that discarding, massaging, or transforming the sensitive attributes from the ML-based classification model does not necessarily remove the unfairness and discrimination. Unfair and discriminatory decisions can still be made because of the correlation of sensitive attributes with other model attributes. This effect is called "redlining."

This article is organized as follows: First, we will start with defining "fairness." We note there are many notions of fairness in practice and various measures of fairness. Next, we review industry best-practice approaches for "fair in-processing" in overcoming unfairness. Then, we discuss some of the key challenges in these approaches. And, finally, we conclude the paper with some closing observations.

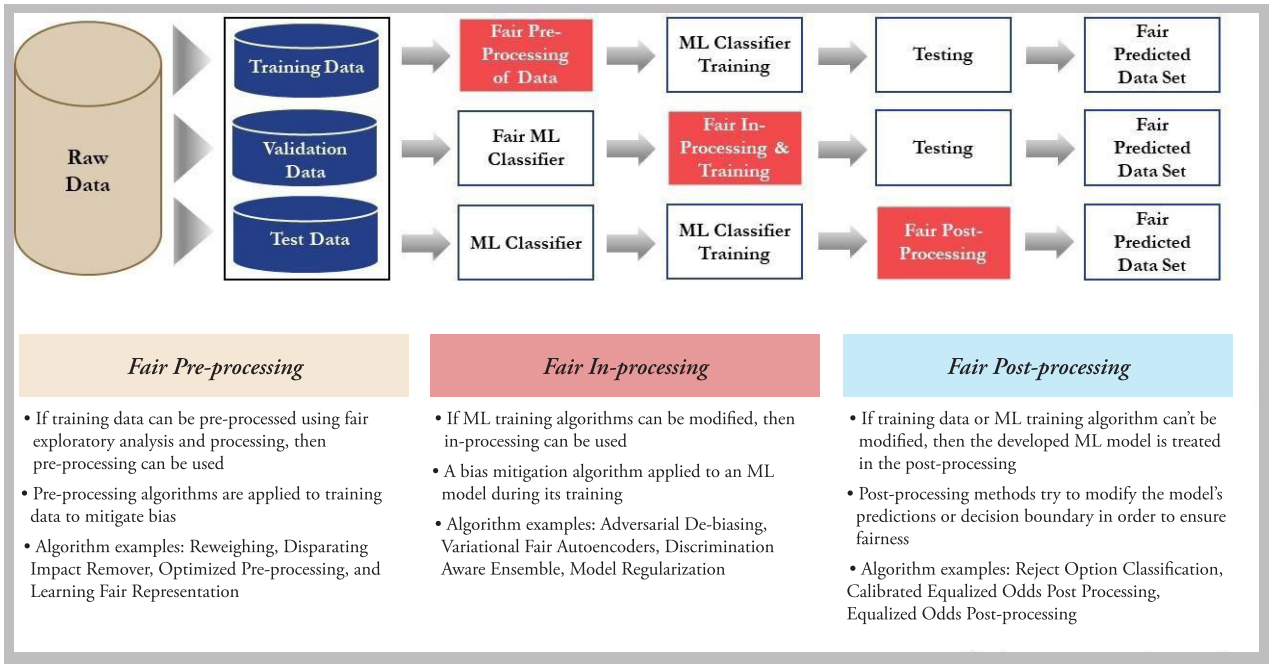


Figure 1: Fairness in ML Algorithm – Workflow

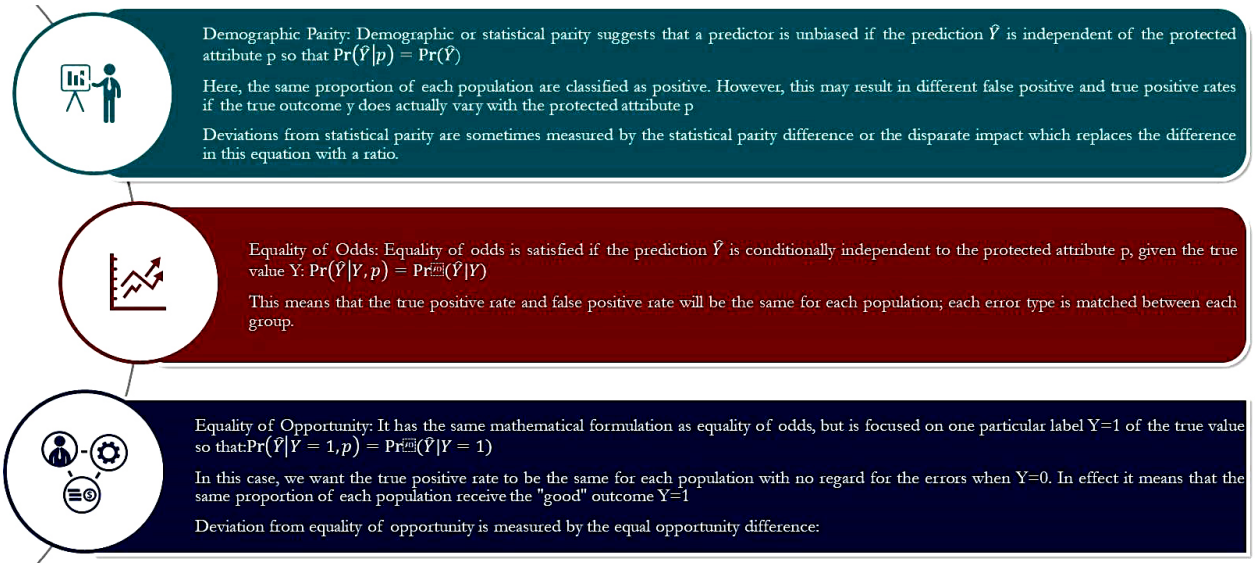


Figure 2: Defining Fairness

Definition of Fairness

Fairness is an elusive concept. Broadly, fairness can be defined as the absence of any prejudice or favoritism toward an individual or a group based on their intrinsic or acquired traits in the context of decision-making. Formulating fairness quantitatively in an ML ecosystem typically starts with utilizing definitions of protected classes embedded in anti-discrimination laws. These laws are based on sensitive attributes such as gender, race, color, religion (e.g., Civil Rights Act of 1964, Title VII, Equal Employment Opportunity Act, 1964 etc.,). To improve a machine-learning algorithm to make it "fair," we must formalize definitions of fairness, develop measures to determine the degree of unfairness dependent on our definition, and develop methods to incorporate fairness into the algorithm, for instance via the loss function or constraints or input data manipulation.

We note that there are a number of notions of fairness developed in the academic literature (see e.g., [7]). However, all of these definitions of fairness intuitively aim for predictions, which are equally effective regardless of different demographic groups or membership. For example, a fair ML model is one that makes a prediction of some variable (e.g., default of payment) based on some attributes denoted by X (e.g., credit card history, age, gender) which may be biased or unfair with respect to some sensitive attribute Z (e.g., gender) - unfairness that is corrected in some way to make the ML fair. The three types of fairness often used in the fair ML world are demographic parity, equality of odds, and equality of opportunity. Detailed definitions of these three types are presented in Figure 2.

Approaches to Fair In-Processing

There are numerous approaches to mitigate algorithmic bias or unfairness, and each approach has its own advantages and limitations. Here we give an overview of the algorithmic approaches followed in the industry to mitigate bias, which we believe could be useful for practitioners, be they users, model developers, or model validators.

Adversarial De-biasing

The adversarial de-biasing methodology is one of the most popularly used approaches for mitigating unfairness/bias in ML algorithms. This methodology relies on adversarial training to mitigate bias from latent representations learned by the model.

As an example, let us assume Z is a sensitive feature the model developer wants the ML model to impede from discriminating on (e.g., age, nationality, gender, or race). Since most of the data has a discriminatory or redlining effect, simply removing Z from the training data is not advisable, as it may be highly correlated with other features such as zip code and income, for example. Given this, the model developer wants to prevent the ML model from learning a representation of the input attributes that relies on Z in any substantial way. Adversarial de-biasing helps us train such models, reducing or preventing the influence of Z and its influence on other variables.

The intuition behind adversarial de-biasing is as follows: In the first step, the original ML model produces a representation of all the attributes, denoted by X, that primarily encrypt information about the sensitive attribute Z (e.g., race). Next, an adversarial model recovers and predicts Z using that representation. If the adversarial models fail to find any evidence about Z, then it is clear the model has learned

a representation of the input attributes that are not substantially dependent on any discriminatory features.

In practice, we can use the adversarial de-biasing framework as a separate neural network model or a multi-head neural network where one head is used for predicting Y (the target, e.g., default) and another head is used for predicting Z. Essentially, the output of the first neural network model is used as the input of the adversarial network.

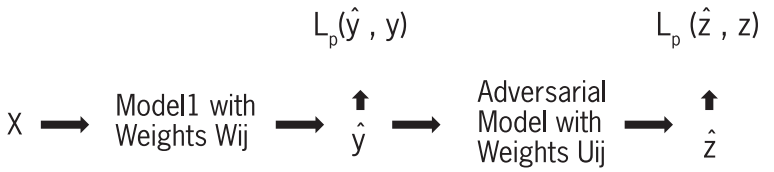
However, training adversarial type networks is challenging. If we are updating the weights in the first model based on improving the ability to predict Y and then updating the weights in the adversarial model, we won't necessarily achieve a fair and unbiased ML model. To achieve fairness, we need to incorporate the first model's weights into the update function, so that the updated model has a worse chance of predicting *Z* at each update. The way one can achieve this is analogous to how Generative Adversarial Networks (GANs) train their generators. [6]

Naïve Bayes Models

Naïve Bayes approaches are popular classification techniques because they are intuitive and easy to implement. In the Naïve Bayes in-processing approach, we divide the data set into two separate sets: one for S+, and one for S– where S+ and S– are one of the discriminating or sensitive attributes. For example, if gender is the sensitive attribute, we could divide the data set into one for males and one for females. The model M+ is trained using only the tuples from the S+ dataset, while another model M– is trained using data set S–.

The overall classifier chooses either model M+ or M– depending on the value of S and uses that model's classification. The overall classifier will need to be calibrated to ensure the overall distribution of the target variable fits with the firm's modeling and risk objectives. This may be a target approval rate for example.

Thus, in a retail scorecard approach, this Naïve Bayes Model could have two different models: One for males, and one for females, underpinned by the same methodology and calibrated to have a target approval rate overall. Such an approach would be more suitable since gender unfairness occurs when males and females are treated differently, whereas here the methodologies are agnostic to the two groups since M+ and M– share the same Naive Bayes structure. See [1],[2] for details and more examples utilizing Naïve Bayes methods for de-biasing ML models.



Illustrated above, the adversarial de-biasing approach can be directly translated into two networks. The first one is used to predict Y using *X* as input. The second model (the adversarial or update model) attempts to use the predicted value of Y to predict *Z*.

Discrimination Aware Ensemble

The discrimination aware ensemble (DAE) model approach is based on the assumption that discrimination mostly impacts instances close to the decision boundary. As an example, consider two candidates with all other attributes roughly equal and near the decision boundary. The candidate in the S– group is typically negatively impacted or discriminated against. The basic hypothesis behind this DAE model is that various classification models disagree more on the instances that are close to the decision boundary or hyperplane (in case of Support Vector Machines), and that disagreement allows the model developer to identify those instances that may be misclassified due to discrimination.

In its simplest form, we consider a family or ensemble of independent ML models. If all independent ML classification models predict the same class/label, that agreed class label is assigned. However, if one or more ML classification models shows disagreement in the model outputs, the ML Ensemble compensates the instances belonging to the deprived group by assigning them the *C*+ label and penalizes the instances belonging to the favored group by giving the *C*– label.

Let us assume that in a retail application scorecard approach, one ML classifier is biased against female applicants and it rejects their applications. Most likely, the ML-based classifier is only able to affect those applicants who are close to the decision boundary because the highly qualified (in terms of income, qualification, experience, and other factors) female applicants cannot be rejected due to their overall highly significant attributes. Since the DAE has more than one ML classifier, the likelihood of the candidate being approved by one of the models is high, thus neutralizing the discriminatory effect. [4],[5] provide a good introduction to the topic and applications.

Fairness Regularized Logistic Regression

This regularization strategy is most appropriate for the logistic regression family of models. In this regularization approach, a fairness regularization term is added in the objective function that penalizes the mutual information between the sensitive

feature and the classifier outcomes. The problem can be formulated mathematically as follows: Find the model parameter $\emptyset$, which minimizes the following objective function:

Where:

y: sample objective (or target) variables

x: sample of non-sensitive features

s: sample of sensitive features

$\emptyset$: model parameter

r: regularization parameter (larger the value, more the fairness)

$R(\{\{y, x, s\}, \emptyset)$: regularizes for prejudice removal, the smaller value more strongly constrains the independence between s and y.

$Pr(\{\{y, x, s\}, \emptyset)$: the probability of the target class (e.g., approval) based on x, s and $\emptyset$.

The last term is a regularizer to prevent overfitting of the model.

The above equation quantifies the degree of prejudice based on the measure of mutual information (MI) and adds this as a "regularizer," R, in a logistic regression framework. MI is one of many factors that measures how much one random variable tells us about another. It is a dimensionless factor and can be thought of as the reduction in uncertainty about one random variable given knowledge of another. High MI indicates a large reduction in uncertainty; low MI indicates a small reduction; and zero MI between two random variables means the variables are independent. The MI plays a critical role and makes Y, the target variable (e.g., good credit, college admission, etc.), completely independent of sensitive features. Thus, this regularization strategy treats the MI as the unfairness proxy, as opposed to covariance. This technique needs the sensitive feature information at decision time, hence cannot remove disparate treatment. See [3] for further details and references as well as a comparison to the Naïve Bayes approaches.

Overcoming Algorithmic Unfairness – Industry Best Practices

In the previous section, we discussed some of the in-processing methodologies to overcome algorithmic unfairness. Although the use of fair-pre-processing is expanding in every industry that utilizes ML, relatively few modelers and validators have a good grasp of the processes and principles that should be followed when developing such methodologies. Therefore, we have also identified some of the recommended best practices to mitigate unfairness and discrimination in ML. Though some of these are obvious and apply to all model development and validation situations, they become even more relevant in this still-nascent field of ML and are therefore worth reiterating. Whether conducting a model development or a model validation, the analyst should understand the best practices shown in Figure 3.

Key Challenges

Although the use of Fair ML model development and validation is expanding within financial institutions, there are still some challenges that become barriers to fair ML model development and validation. Based on the authors' collective experience with various regional and global financial institutions, we have listed some of the key challenges in Figure 4.

Conclusion

Biased machine learning models may

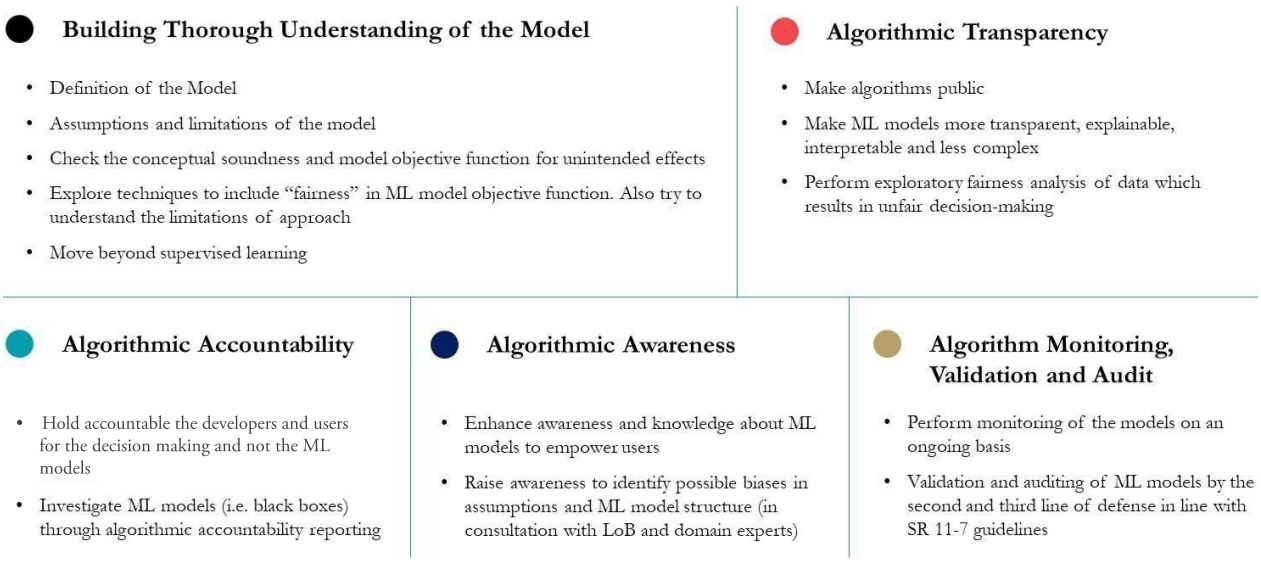


Figure 3: Overcoming Algorithmic Bias – Best Practices

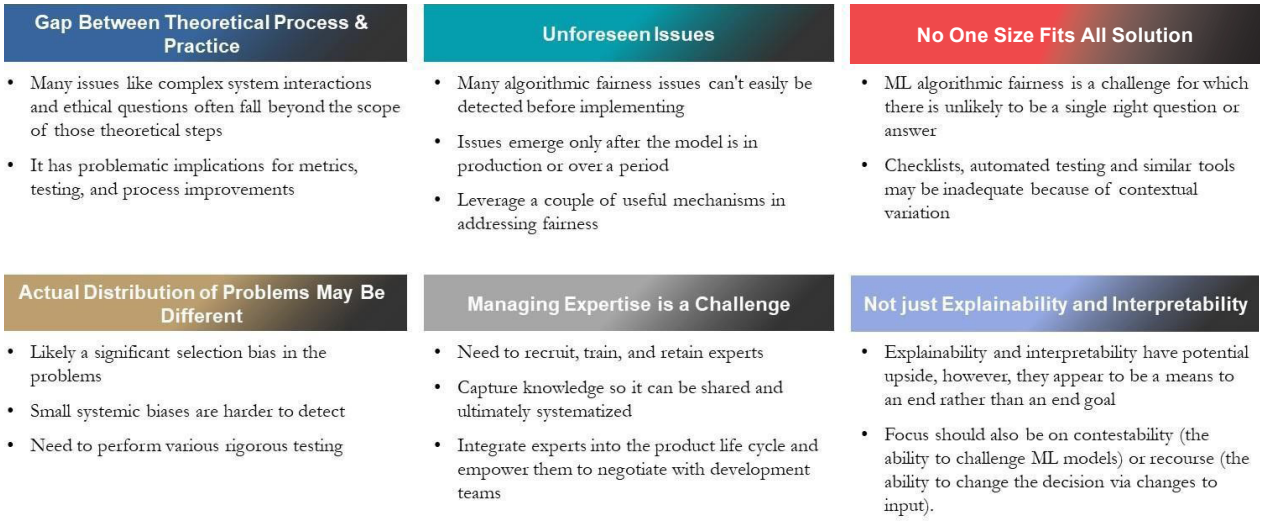


Figure 4: Overcoming ML Algorithmic Bias – Key Challenges

result in making unfair or biased decisions that not only negatively impact the discriminated individuals or groups, but could result in financial and legal loss as well as reputational damage that could impact the firm for years to come or, in the worst-case scenario, push a firm to insolvency. There could even be direct consequences to the developers of these models and the validators who approved the use of these models. Thus, the opportunities to have a positive impact through fairer and more transparent decisions are paramount. There exist three primary strategies to mitigate bias and ensure fairness in ML models: pre-processing, in-processing, and post-processing. Each method has its own pros and cons.

The right solution for a business will depend upon a number of factors, but certain foundations need to be in place. In this article, we have provided:

- An in-processing fairness approach to avoid bias in ML models
- Industry best-practices for avoiding bias in ML model through fair in-processing approaches
- Current key challenges institutions need to overcome to build a strong ML ecosystem.

To conclude, the model developers, validators, and auditors can apply the solutions proposed in this paper to ensure the ML-based classification models they build and validate are fair and discrimination-free by developing an appropriate development plan that identifies, measures, and mitigates potential unfairness in the development process; has an ongoing validation and monitoring approach to identify problems; and appropriate policy and procedure to quickly correct any identified problems. ®

Notes

1. Toon Calders and Sicco Verwer. 2010. Three Naive Bayes Approaches for Discrimination-Free Classification. Data Mining and Knowledge Discovery 21, 2 (2010), 277–292.
2. Calders T, Kamiran F, Pechenizkiy M (2009) Building classifiers with independency constraints. In: IEEE ICDM workshop on domain driven data mining. IEEE press
3. Toshihiro Kamishima, Shotaro Akaho, Hideki Asoh, and Jun Sakuma. Fairness-aware classifier with prejudice remover regularizer. In Peter A. Flach, Tijl De Bie, and Nello Cristianini, editors, earning and Knowledge Discovery in Databases, pages 35–50, Berlin, Heidelberg, 2012.
4. Verwer S., Zhang Y. (2017) Learning Decision Trees with Flexible Constraints and Objectives Using Integer Optimization. In: Salvagnin D., Lombardi M. (eds) Integration of AI and OR Techniques in Constraint Programming. CPAIOR 2017. Lecture Notes in Computer Science, vol 10335. Springer, Cham
5. F. Kamiran, A. Karim and X. Zhang, "Decision Theory for Discrimination-Aware Classification," 2012 IEEE 12th International Conference on Data Mining, Brussels, 2012, pp. 924-929, doi: 10.1109/ICDM.2012.45.
6. D. Xu, S. Yuan, L. Zhang and X. Wu, "FairGAN: Fairness-aware Generative Adversarial Networks," 2018 IEEE International Conference on Big Data (Big Data), Seattle, WA, USA, 2018, pp. 570-575, doi: 10.1109/BigData.2018.86225
7. Alexandra Chouldechova, Aaron Roth. "A Snapshot of the Frontiers of Fairness in Machine Learning", Communications of the ACM, May 2020, Vol. 63 No. 5, Pages 82-89

MANISH KUMAR, PH.D. is principal at Solytics Partners, a global analytical company where he leads the Machine Learning and Automation Group, and is also a senior consultant for RMA's Model Validation Consortium. He can be reached at manish.kumar@solytics-partners.com.

RAHUL ROY, PH.D. is senior quantitative analyst at Solytics Partners, a global analytical company, where he works on automation in the risk and analytics space. He is also a consultant for RMA's Model Validation Consortium. He can be reached at rahul.roy@solytics-partners.com.

KEVIN D. ODEN is the founder and managing partner of the risk management consulting firm Kevin D. Oden & Associates, and is the managing director of RMA's Model Validation Consortium. He can be reached at kevin.oden@kdoden.com.