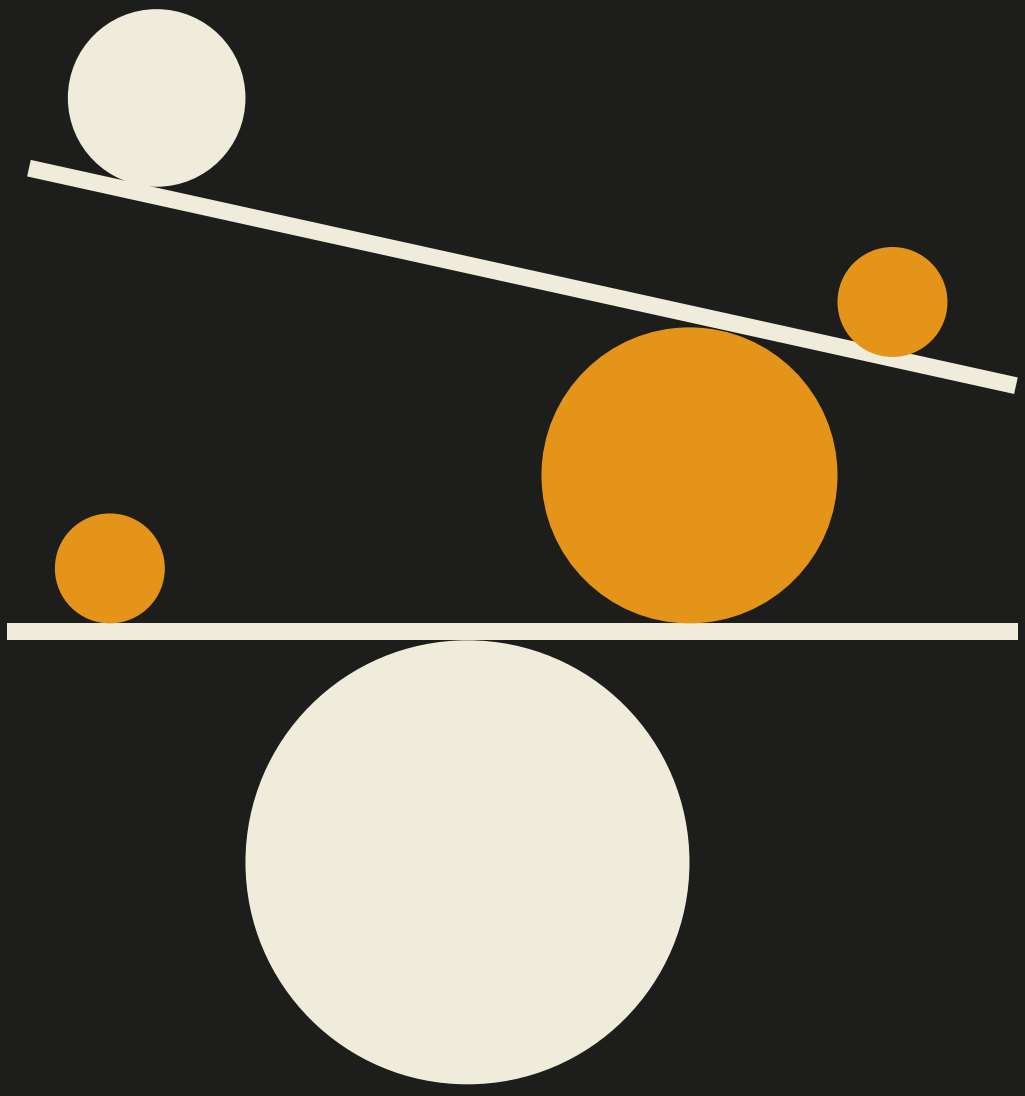
FAIRNESS IN ALGORITHMIC DECISION MAKING



BY SAMPATH KANNAN[1]

1 Sampath Kannan is the Henry Salvatori Professor at the Computer and Information Science Department, University of Pennsylvania.

CPI ANTITRUST CHRONICLE JUNE 2023

WHAT IS ALGORITHMIC BIAS AND WHY ANTITRUST AGENCIES SHOULD CARE?
By Giovanna Massarotto

UNLEASHING THE POWER OF ALGORITHMS IN ANTITRUST ENFORCEMENT: NAVIGATING THE BOUNDARIES OF BIAS AND OPPORTUNITY
By Holli Sargeant & Teodora Groza

ALGORITHMIC PRICING AND COMPETITION
By Robert Clark & Daniel Ershov

CAN WE GET THE BIAS OUT OF OUR AI?
By Paola Cecchi Dimeglio

CAN SELF-PREFERENCING ALGORITHMS BE PRO-COMPETITIVE?
By Emilie Feyler & Veronica Postal

FAIRNESS IN ALGORITHMIC DECISION MAKING
By Sampath Kannan

Visit www.competitionpolicyinternational.com for access to these articles and more!

FAIRNESS IN ALGORITHMIC DECISION MAKING

By Sampath Kannan

This article explains how machine learning algorithms work and how they are being used to make critical decisions. Eliminating bias in these algorithms and making them fair to groups and individuals is vitally important. We review a few common definitions of fairness of algorithms, and point a way out of the seeming impasse of choosing between mutually incompatible criteria in some scenarios. We identify two things that algorithmic decision-making systems can learn from human decision-making systems – rules-of-evidence-type limitations on the kinds of data that may be used, and exercising great forbearance in making highly preemptive decisions. Finally, we describe what machine learning models are and describe the need for transparent and accountable models..

Machine learning algorithms are being used to classify individuals in increasingly consequential situations. Such algorithms are used for example, in hiring, college admissions, bank loan decisions, and in the criminal justice system for predictive policing and predicting recidivism risk. It has been observed that while algorithmic decision-making may appear *prima facie* to be more objective than human decision-making, it inherits some of the problems of human decision-making, and presents new ones.

How do machine learning algorithms make decisions? What are their desiderata and what yardsticks do we use to measure whether they are met? What new challenges do algorithms present, compared to human decision-makers? What failings in human decision-making do they ameliorate? These are some of the questions we will examine in this brief article.

While machine learning comes in many forms, most deployed systems to date use a form of learning called *supervised learning*. Taking bank loan applications as our running example, in supervised learning, the algorithm is given many examples of individuals who applied for a bank loan along with the decisions made by a human decision-maker in each case. An application might have fields containing demographic information, as well as financial data such as income, assets, obligations, etc. For the algorithm, an applicant is just this *vector of features*, where each feature is a quantity capturing one of these items of information. The decisions by the human decision-maker --- YES / NO, coded as 1 / 0, are the *labels* associated with these examples. If the human decision maker gives more informative scores to each applicant, the learning algorithm could presumably also use this. Supervised learning is characterized by the fact that the algorithm is given such labeled examples, and must then figure out a "rule" to apply to future applicants.

This rule has to be "simple": If it follows all the "idiosyncratic complexities" of the human decision maker, it is merely "memorizing," not generalizing, and therefore not learning. This statement can be made mathematically precise. A consequence of this requirement of simplicity is that machine learning algorithms invariably make mistakes, unless the classification task at hand is inherently simple.

While classification errors are inevitable, one could ask that these mistakes do not result in discrimination against certain individuals, and systemic discrimination against groups based on gender, race, or other protected attributes. To reason about this, one needs mathematical definition(s) of what it means for an algorithm to be fair, and for it to discriminate. It is beyond the scope of this short article to even mention the many intellectual strands from philosophy, legal theory, political science, and other areas that inform the kinds of definitions that have been made. We simply describe some of the most common definitions in use. Notions of *individual fairness* seek to prevent discrimination against individuals, while notions of *group fairness* seek to ensure that protected groups (as defined in law) are treated similarly under different statistical notions of parity.

The simplest notion of group fairness is *demographic parity*. Is the same fraction of people in each protected group given a benefit or caused harm by a decision? Is each group "represented" in proportion to its share of the population? Demographic parity might not be appropriate if different groups have different qualifications or interests. In the bank loan example, the algorithm classifies people as positive, i.e. worthy of a loan, or negative, i.e. not worthy of a loan. The *false negative* rate for a group is the fraction of individuals in that group who are classified as negative, even though they are truly loan worthy (or positive). A natural goal is to equalize the false negative rate amongst groups. This accords with the well-studied principle of "equality of opportunity," since a positive classification can be seen as an opportunity. We might go further and ask that both the false negative rate, and the false positive rate (which has a similar definition) be equalized across groups. This stronger notion of fairness is referred to as "equalized odds."

In order to apply these notions of fairness, we need to know who the true positives and true negatives are in each group. This typically may come from training data where human experts have labeled people as positive or negative. Since in this setting we take the labels assigned by humans as the ground truth, a classifier that exactly mimics this human labeling would have 0 false positive and false negative rates on all groups, and thus achieve equalized odds. However, recall that a good machine learning algorithm has to produce simple classifiers, in order to be useful in classifying new data. Hence it will invariably not mimic these labels exactly and there will be non-zero false positive and negative rates. These metrics can be criticized because they regard the training data as ground truth. However, this is unavoidable, because all machine learning involves learning from data, and algorithms designed in this way can only be as good as the data on which they are trained. These metrics are in line with meritocratic, equity-based notions of distributive justice.

There is also another class of metrics that are used to judge the fairness of machine learning algorithms. To understand them, let's think of an algorithm that predicts the weather each day for a period of time. Let's be even more specific and say that for some fixed location, say New York, on each day, it predicts the probability of rain the next day. How do we judge the quality of such an algorithm? Even if it says that there is 90 percent chance of rain, and it does not rain, can't the algorithm defend itself because it left open some possibility of no rain? A good metric for this scenario is *calibration*, which captures the idea that the numerical probabilities that the algorithm outputs should mean something, just

as the numerical reading of length from a ruler means something. An algorithm such as this one is said to be calibrated, if it rained exactly on 20 percent of the days on which it predicted a 20 percent chance of rain. (Of course, it must similarly satisfy the condition that it rained exactly on 30 percent of the days on which it predicted a 30 percent chance of rain, etc.) Calibration can also be used as a fairness notion.

Turning to a scenario where humans are being assigned a score by an algorithm, we can use calibration as a group fairness notion by requiring an algorithm to be calibrated separately on each protected group. Thus, for example, among the people in one group for whom the algorithm rated loan-worthiness as 80 percent, exactly 80 percent should be loan-worthy. Similar to calibration, there are notions known as positive predictive value and negative predictive value, which can be applied to algorithms that simply classify people and don't assign them scores.

None of these notions of fairness deal with corrective justice, and in fact, not many mathematical definitions capture this important notion. While calibration for each group, and equalized odds all sound like good notions of group fairness, unfortunately, except under the most perfect of circumstances with respect to how the groups are endowed with qualifications. Thus, in designing these algorithms we are forced to choose whether they will achieve equalized odds or be calibrated.

The argument about whether equality of opportunity/equalized odds or calibration is the better measure of fairness is not purely academic. The company Northpointe produced a recidivism prediction software package called COMPAS that critics at ProPublica found not to meet the fairness goal of equal false positive and false negative rates between the Caucasian and African-American populations it made predictions on. In their defense, Northpointe pointed out that the predictions for the same data were calibrated for both groups.

Let us briefly examine some arguments in favor of each of these notions of fairness. An algorithm that “really understands” the members of a group can hope to be calibrated by using that understanding in its predictions, just as a really good weather prediction model can hope to be calibrated in its task. However, one can be calibrated with just rudimentary statistical knowledge about a group. For example, when predicting the weather in Phoenix for the month of October, suppose we know from years of observation that it rains 5 percent of the time in October. Then we could predict a 5 percent chance of rain every day without having too accurate a weather model for the actual probability of rain each day, and we would probably be pretty close to calibrated. On the other hand, if we knew the weather patterns in New York intimately, we could make more varied and informative predictions for each day, and be calibrated as well. Thus, being calibrated may not tell us anything about how accurate predictions are to subsets of individuals in each group.

As a point in favor of calibration, it has been argued that an algorithm that best uses all the information it has at hand to produce the most accurate classification, will produce a calibrated classifier, and therefore this is the right kind of classifier to produce.

A system where the people being classified adjust their behavior and the data they generate based on how the classifier works is called an endogenous system. For example, imagine that individuals in different groups choose to commit a crime or not based on a risk/benefit analysis, which in turn depends on the probability with which the classifier will find a person guilty when they commit a crime and the probability with which it will find that person guilty when they do not. An individual's incentive to commit a crime is governed by the difference in these probabilities --- the larger the difference, the more dissuaded s/he is, to commit a crime. As the analysis shows, an adjudicator wanting to minimize the number of crimes overall should equalize the false positive and false negative rates across, even though equalizing calibration leads to a more accurate classifier! Here accuracy is less important than incentivizing individuals away from crime. This point naturally leads to the second part of the paper where we ask what machine learning systems could learn from more traditional systems.

In cases tried in courts there are rules of evidence that vary across jurisdictions and across types of cases that restrict what kinds of evidence may be presented. Should there be a similar restriction in the kind of data a machine learning algorithm is allowed to use? No such restrictions exist on today's learning algorithms. The justification for this might be the thinking that more data is never bad – If some features are uncorrelated or even negatively correlated with the desired output of the algorithm, the algorithm will learn this and disregard or lower the weight it gives to these features. However, if the algorithm is optimizing for accuracy it may well use some features, which would have been best left unused. In the example in the previous paragraph, an adjudicator wanting to minimize crime rate should ignore which group an individual belongs to, even if the most accurate classifier demands that this information be used. A further study of rules of evidence, the reasons they were put in place, and how machine learning algorithms might adopt them would be invaluable.

Another potential in traditional systems one is wary of making “preemptive” decisions, lest they impinge on individual liberties. But it is difficult to define what constitutes a preemptive decision. On the one hand, a decision that is entirely a function of existing data seems to be too deterministic . On the other hand, all decisions are made based on existing data. Nevertheless, we tend to think of a restraining order on an individual, or the denial of certain types of internet or social network access as examples of preemptive actions, while a sentence including jail

time for a crime for which one has been convicted does not seem so preemptive. Preemptive judgments have to be made carefully balancing the need to protect individual liberties against the potential harm to society and the individual. How can we define when an algorithmic decision is preemptive? Or perhaps, define the degree of preemption involved in a decision? Do we forbid preemptive decisions in certain spheres and allow them in others? And how do we redesign machine learning algorithms to be only as preemptive as warranted? There has been no work along these lines because even defining what preemption means is challenging.

Algorithmic decision-making is nucleated on decisions made by human decision-makers. This is both a feature and a bug. On the negative side, this means that bias and discrimination in human decision-making become systematized and broadly applied. However, one could hope to mitigate this bias by choosing training data from a diverse set of human decision-makers. One of the major issues in algorithmic decision-making is representational fairness in the training data, both in the set of people the algorithm is trained on, as well as in the set of features it uses.

However there is hope. Unlike a human decision maker, an algorithm is eminently auditable in all aspects: The particular machine model used, how it was trained, and what results it produces on test data can all be fully recorded and analyzed. In fact, for this reason, these algorithms should be monitored and audited constantly. There is a concern about the auditability of really complex algorithms.

To understand this concern, it is first necessary to understand what machine learning models are. When designing a machine learning classifier for a task (such as determining who gets a bank loan) one first chooses the type of classifier one wants. For example, a classifier might have the rule, “If 5*income + assets - liabilities - loan amount > 0, then award the loan.” Of course this is a highly simplified, unrealistic example of this kind of rule. This kind of rule computes a weighted score out of the features of the application and grants a loan if this score is above a threshold. Before looking at any data this model (the type of rule the algorithm will use) is already chosen; during training the weights (5 for income, +1 for assets, -1 for liabilities, and -1 for loan amount) are learnt.

Another popular kind of model is a decision tree. Here the algorithm asks a series of (typically yes/no) questions, and based on the answers, decides how to classify. In the same highly simplified setting above, the first question might be, “Is income > 3*loan amount?” If the answer is “yes,” a second question might be, “Is credit rating > 600?” and if “yes” again, perhaps the loan is granted. However, if the answer to the first question is “no,” then perhaps a second question might be, "Is assets - liabilities > loan amount?," and again if the answer is “yes” the loan might be granted. So each possible sequence of answers to these questions leads one down a path, at the end of which lies a decision - to grant the loan or not. Here the model is a decision tree. But what question to ask first, what subsequent questions to ask for each possible answer to the first question, etc., are learnt from the training data during the training phase.

These are just two of the simplest and most common models in use.

Lately, a form of learning called deep learning has proved remarkably effective in natural language understanding, computer vision, and many other tasks. Here models are incredibly complex, loosely based on our understanding of how neurons connect and work in biological organisms. Again, the rough architecture of the neural net is the model chosen *a priori*, and the training data helps determine the parameters of this model.

As may be evident from the three examples of models above, they can vary greatly in their understandability. Notions such as transparency, interpretability, accountability have become increasingly important desiderata as algorithms in complex models are making decisions nearly as inscrutable as some decisions made by humans.

As remarked earlier, machine learning algorithms in critical decision making tasks should be monitored continually and audited from time to time. It also seems clear that these monitoring and auditing tasks should not be done solely by mathematicians and engineers. To engage policy makers, social scientists, etc., it is important that these models be transparent and interpretable.

In summary, algorithmic decision making offers the prospect of reducing bias in critical decisions. However, in order to succeed such algorithms should mimic some aspects of human decision making that have evolved over centuries of experience. Because algorithms often learn from flawed data, and because their design might not be perfectly aligned with societal goals, they should be continually monitored, and modified as necessary. In any particular application, choosing between competing, and sometimes incompatible notions of fairness will remain a major challenge, that must be solved jointly by algorithm designers, policy makers, and society at large.

CPI COMPETITION POLICY® INTERNATIONAL

CPI Subscriptions

CPI reaches more than 35,000 readers in over 150 countries every day. Our online library houses over 23,000 papers, articles and interviews.

Visit competitionpolicyinternational.com today to see our available plans and join CPI's global community of antitrust experts.