

On Testing for Discrimination Using Causal Models

Hana Chockler,^{1,2} Joseph Y. Halpern³

¹ causaLens

² Department of Informatics, King’s College London

³ Computer Science Department, Cornell University

hana.chockler@kcl.ac.uk, halpern@cs.cornell.edu

Abstract

Consider a bank that uses an AI system to decide which loan applications to approve. We want to ensure that the system is *fair*, that is, it does not discriminate against applicants based on a predefined list of sensitive attributes, such as gender and ethnicity. We expect there to be a regulator whose job it is to certify the bank’s system as fair or unfair. We consider issues that the regulator will have to confront when making such a decision, including the precise definition of fairness, dealing with proxy variables, and dealing with what we call *allowed* variables, that is, variables such as *salary* on which the decision is allowed to depend, despite being correlated with sensitive variables. We show (among other things) that the problem of deciding fairness as we have defined it is co-NP-complete, but then argue that, despite that, in practice the problem should be manageable.

1 Introduction

AI systems are playing a larger and larger role in decision making these days, in applications like deciding who to interview and hire, deciding who gets paroled, and deciding who gets credit. Moreover, AI systems can often make these decisions better than people (Kleinberg et al. 2018a). However, as many have noted, this raises the concern that decisions are made based on sensitive attributes, such as race, gender, or religion.

Given the laws and regulations governing discrimination (i.e., making decisions based on the values of sensitive variables), we consider what we suspect will be an important use case in the future. We assume that there is a regulator that regulates financial institutions, for example, banks, and in particular the decisions made by the banks on whether to grant loans to applicants. (For definiteness, we assume that the system being regulated is a bank’s system for determining who gets a loan. But the points that we make apply without change to all decision-making systems where there are discrimination concerns.) The bank wants to make this decision based on their (possibly proprietary) causal/machine learning model. (We do not distinguish causal models from machine learning models, for reasons that will be clear shortly.) The bank comes to the regulator seeking approval.

The regulator has some variables that she considers sensitive. Intuitively, the bank is not supposed to use these in making its decision (although some uses may be permitted, as we shall see). The bank may view its model as proprietary, so wants to keep as many of the details regarding its model private, while still convincing the regulator that it is not discriminating.

We take the bank’s algorithm to be a “grey box”, where some of its features must be disclosed, but the bank can still keep many of its features proprietary. Specifically, we assume that the bank will need to disclose only which features are inputs and how they are computed, and provide the regulator with black box access to the system, so she can see the decision made given certain inputs. The bank will also request the regulator to have certain variables be explicitly *allowed*. Intuitively, allowed variables are variables that are correlated with sensitive variables that can be used by the bank’s algorithm to make decisions. For example, *gender* may be considered a sensitive variable, but *salary* may be an allowed variable, although it is correlated with gender. (Allowed variables have been called *resolving* variables; see, e.g. (Kilbertus et al. 2017).) The regulator will have to decide whether to agree with the bank’s request regarding allowed variables. This is not an easy decision, and is one that ethicists and society at large may have to resolve. Nevertheless, we believe that there are necessary conditions that must be met for a variable to be allowed. The issues that arise here are essentially those that determine whether *disparate impact* has taken place, according to American law (Primus 2003).

Given the sensitive and allowed variables, our notion of fairness then says, roughly speaking, that the bank’s software is fair (i.e., acceptable to the regulator) provided that, if we fix the values of the allowed variables, changing the values of the sensitive variables has no impact on the outcome. While our definition is very much in the spirit of earlier definitions of fairness that use causal models (in particular, the notion of *counterfactual fairness* introduced by Kusner et al. (2017), path-dependent notions of fairness considered in (Chiappa 2019; Nabi and Shpitser 2018), and the notion implicitly used by Kilbertus et al. (2017)), it differs in one significant way. Whereas the earlier definitions are all statistical, ours is not: it requires that outcomes are the same, not that their probabilities are equal. We argue that, for our set-

ting, this is appropriate. Roughly speaking, we view a system as fair if it is fair for each applicant.

In this setting, we also examine the effect of *proxy variables*. It is often not difficult for an AI system to find a proxy for a sensitive variable and use that instead. For example, if *gender* is a sensitive variable, an AI system may use a highly correlated variable like *favorite clothes* as a proxy for gender. Indeed, not only can an AI system find proxy variables, if it is told that it cannot use sensitive variables in its decision, it will actively seek out proxies. Prince and Schwarcz (2020) point out that while the use of proxy variables is incompatible with (American) anti-discrimination laws, it is likely to increase substantially as more AI systems are used.

Kilbertus et al. (2017) take proxy variables to be nothing more than descendants of sensitive variables in the causal graph. If this were always the case, then dealing with them would be easy. Changing the value of a sensitive variable should change the value of its proxies, and hence the outcome. Our approach would call this unfair.

Unfortunately, it is *not* the case that proxy variables are always descendants of sensitive variables, for (at least) two reasons. The first is that a proxy variable can be correlated with a sensitive variable if it is a descendant of an ancestor of the sensitive variable. For example, if *religious affiliation* is a sensitive variable, one of its parents in the causal graph might be *religious affiliation of parents*. This is clearly a good proxy for *religious affiliation* even though it is not a descendant of it. We solve this problem by requiring that all ancestors of sensitive variables be taken to be sensitive as well. However, there is another, arguably more serious reason that a proxy variable might not be a descendant of a sensitive variable. Suppose that an AI system is often able to determine (perhaps by checking social media) which religious holidays an applicant celebrates (if any). Moreover, it treats this as an input variable. Of course, in an actual causal model of the world, *religious holidays celebrated* is clearly a descendant of *religious affiliation*. However, in the bank’s model, it is not. It is just a variable whose value is determined from social media. The bank’s system will not “understand” that it should be a descendant of religious affiliation, and the bank’s system designers might not even be aware of it being used. While the connection between *religious affiliation* and *religious holidays celebrated* is blatantly clear, the connection between other variables may not be at all clear, and not recognized by the system designers. In any case, *religious holidays celebrated* is *not* a descendant of *religious affiliation*; changing the value of religious affiliation will not affect the media posts observed. We discuss how the regulator can deal with this in Section 3.

To summarize, the main contribution of this paper lies in creating a framework that clearly delineates what a regulator will have to do in order to certify an AI system for fairness. In doing so, we highlight the subtleties involved in dealing with allowed variables and proxy variables, and make the case for a non-statistical definition of fairness. We also examine the complexity of determining whether a system is fair, and show that it is co-NP-complete in the size (i.e., number of variables) of the system, but then argue that this should not be a problem in practice.

2 Causal Models

In this section, we review the definition of causal models introduced by Halpern and Pearl (2005). The material in this section is largely taken from (Halpern 2016).

We assume that the world is described in terms of variables and their values. Some variables may have a causal influence on others. This influence is modeled by a set of *structural equations*. It is conceptually useful to split the variables into two sets: the *exogenous* variables, whose values are determined by factors outside the model, and the *endogenous* variables, whose values are ultimately determined by the exogenous variables. The structural equations describe how these values are determined.

Formally, a *causal model* M is a pair $(\mathcal{S}, \mathcal{F})$, where \mathcal{S} is a *signature*, which explicitly lists the endogenous and exogenous variables and characterizes their possible values, and \mathcal{F} defines a set of (*modifiable*) *structural equations*, relating the values of the variables. A signature \mathcal{S} is a tuple $(\mathcal{U}, \mathcal{V}, \mathcal{R})$, where \mathcal{U} is a set of exogenous variables, \mathcal{V} is a set of endogenous variables, and \mathcal{R} associates with every variable $Y \in \mathcal{U} \cup \mathcal{V}$ a nonempty set $\mathcal{R}(Y)$ of possible values for Y (i.e., the set of values over which Y ranges). For simplicity, we assume here that \mathcal{V} is finite, as is $\mathcal{R}(Y)$ for every endogenous variable $Y \in \mathcal{V}$. \mathcal{F} associates with each endogenous variable $X \in \mathcal{V}$ a function denoted F_X (i.e., $F_X = \mathcal{F}(X)$) such that $F_X : (\times_{U \in \mathcal{U}} \mathcal{R}(U)) \times (\times_{Y \in \mathcal{V} - \{X\}} \mathcal{R}(Y)) \rightarrow \mathcal{R}(X)$. This mathematical notation just makes precise the fact that F_X determines the value of X , given the values of all the other variables in $\mathcal{U} \cup \mathcal{V}$.

The structural equations define what happens in the presence of external interventions. Setting the value of some variable X to x in a causal model $M = (\mathcal{S}, \mathcal{F})$ results in a new causal model, denoted $M_{X \leftarrow x}$, which is identical to M , except that the equation for X in \mathcal{F} is replaced by $X = x$.

We can also consider *probabilistic causal models* if we want to talk about the probability of causality (and, for our purposes, the probability of discrimination). A probabilistic causal model is a tuple $M = (\mathcal{S}, \mathcal{F}, \text{Pr})$, where $(\mathcal{S}, \mathcal{F})$ is a causal model, and Pr is a probability on contexts.

The dependencies between variables in a causal model $M = ((\mathcal{U}, \mathcal{V}, \mathcal{R}), \mathcal{F})$ can be described using a *causal network* (or *causal graph*), whose nodes are labeled by the endogenous and exogenous variables in M , with one node for each variable in $\mathcal{U} \cup \mathcal{V}$. The roots of the graph are (labeled by) the exogenous variables. There is a directed edge from variable X to Y if Y depends on X ; this is the case if there is some setting of all the variables in $\mathcal{U} \cup \mathcal{V}$ other than X and Y such that varying the value of X in that setting results in a variation in the value of Y ; that is, there is a setting \vec{z} of the variables other than X and Y and values x and x' of X such that $F_Y(x, \vec{z}) \neq F_Y(x', \vec{z})$. A causal model M is *recursive* (or *acyclic*) if its causal graph is acyclic. It should be clear that if M is an acyclic causal model, then given a *context*, that is, a setting \vec{u} for the exogenous variables in \mathcal{U} , the values of all the other variables are determined (i.e., there is a unique solution to all the equations). In this paper, following the literature, we restrict to recursive models. We call a pair (M, \vec{u}) consisting of a causal model M and a context \vec{u} a *(causal) setting*.

3 A Regulatory Framework

In this section we provide more detail about how we expect the regulatory framework to work.

Sensitive variables: We assume that, for each application, the regulator has an initial set of sensitive variables (*race*, *gender*, and so on), typically determined by the law, and a causal graph that includes these sensitive variable and perhaps some ancestors of these variables. The complete set of sensitive variables consists of this initial set and all the ancestors of variables in the initial set, according to the causal graph. The technical reason for closing off the set of sensitive variables under ancestors is, as we mentioned in the introduction, to deal better with proxy variables. Fortunately, in practice, it does seem reasonable to treat ancestors of sensitive variables as sensitive. If *religious affiliation* is sensitive, it seems reasonable to also view *religious affiliation of parents* as sensitive, that is, a variable whose value cannot be used in making the decision.

The bank’s network: The bank can then build its own AI software. Note that we can view a neural network as a causal graph. The inputs to the network can be viewed as exogenous variables; the internal nodes are endogenous variables, whose values are determined from the values of its parents using some function (e.g., a softmax).

While the neural network does not, strictly speaking, have a semantic causal interpretation, we claim that we can still view the network as a causal graph. The causal dependence of the bank’s decision on whether to approve the loan is indeed *caused* by the values of the inputs to the network. Moreover, the neural network can be viewed as describing causal dependencies between variables, and allows us to determine the effect of interventions on its variables; this is exactly what we need for our purposes.

The bank is assumed to have access to the regulator’s causal graph and can use some of the variables in the graph in constructing its causal network. However, it must use the same causal equations for the sensitive variables as the regulator uses. Intuitively, this means that the variables must have the same meaning for the regulator and the bank. We assume that among the output values (i.e., leaves) of the bank’s causal network is the *decision*. All our definitions are with respect to a particular decision. The network can have several decision variables, and it can be fair with respect to some of them and unfair with respect to others.

Allowed variables: After the software is built, the bank may ask the regulator to consider certain variables as allowed. The bank will have to make a case for this; as we suggested in the introduction, we expect the case to have the same form as that currently made to justify a practice having disparate impact in American law. Namely, the bank would have to show that considering these variables is justified by “business necessity”. For example, the bank might argue that, if it is not allowed to take salary into account, the decisions made would be so bad that the bank would just stop making loans altogether. The bank will have to collect data to back this up. But we should note that what counts as appropriate justification of disparate impact standard is

widely disputed. It may be far from obvious what the “right” thing is to do. Consider an example taken from Kleinberg et al. (2018b):

A state government is hiring entry-level budget analysts. It gives a preference to applicants from the prestigious colleges and universities, because these applicants have done best in the past. This has a disproportionate adverse effect on African-American applicants.

Should the variable *university rank* be allowed? A strong business case would have to be presented. This observation suggests that if a system with certain allowed variables is judged to be fair, and some groups feel that it is nonetheless discriminatory, the regulator’s choice of allowed variables might serve as the basis for a legal challenge.¹ Despite the difficulty of doing this, and the potential for lawsuits, we believe that the regulator will ultimately need to decide which variables to treat as allowed (perhaps with inputs from various interested parties).

Proxy variables: As we said, we expect the regulator to treat the bank’s software as a “grey box”. But it will need to be told all the input variables and how they are obtained. The main reason for needing to know the input variables is to test for proxy variables. As we pointed out in the introduction, while we can deal with proxy variables that are descendants of sensitive variables (see also below), we will also have to deal with proxy variables such as *religious holidays celebrated* that the bank uses as inputs (i.e., exogenous variables). We do not see any way of checking this other than by checking, for each sensitive variable, whether some subset of input variables gives inappropriate information about the sensitive variables.

We formalize this below, but before going on, we should stress that the concern about proxy variables used by a system being correlated with sensitive variables is a real one, that has been shown to arise in practice. For example, Datta et al. (2015) showed that the AI system used by Google to decide which job ads to show users makes some discriminatory decisions. When users provided gender information on the Ad Settings page, Datta et al. showed that simulated users who indicated that they were male received ads that promised large salaries more frequently than simulated female users. But Google clearly used as input more than just the Ad Settings to decide which ads to show to each user. The kind of ads shown depended in large part on the web pages visited by the user. Clearly, the web pages visited can be a proxy for gender. For example, the bloggers that the user follows, use of particular keywords in the user’s posts on social media, and the user’s shopping activity can all be used to infer gender. Each variable separately might not have a high correlation with gender, but together they might indicate with a high degree of certainty that the user is female.

¹To give just one real-world example of the difficulty of deciding what should be allowed, as pointed out by Kleinberg et al. (2018b), there are ongoing debates and studies regarding whether, in our language, it is reasonable to take the variable *prior incarceration record* to be allowed. Does it help or hurt willingness to hire black applicants? (See, e.g., (Agan and Starr 2018).)

In addition to gender, which is clearly a sensitive attribute and should not influence the job ads shown, Datta et al. also found that ads shown depend on whether the user visits certain webpages associated with substance abuse. Here it is less clear whether this should be illegal, as Google might argue that substance abuse is highly correlated with inability to keep a high-responsibility (and high-paying) job. In the language of this paper, Google might argue that *substance abuse* should be an allowed variable; it is then up to the regulator to approve or deny this request.

There are a number of plausible definitions of what it means for the bank’s input variables give inappropriate information about sensitive variables. We consider two requirements that we believe capture the intuition:

- For some sensitive variable X , the event $X = x$ is independent of $\vec{Y} = \vec{y}$, where \vec{Y} is the set of disallowed variables, for all settings x of X and \vec{y} of \vec{Y} .
- For some sensitive variable $X = x$, X is conditionally independent of $\vec{Y} = \vec{y}$ given $\vec{A} = \vec{a}$, where \vec{Y} is the set of disallowed variables and \vec{A} is the set of allowed variables, for all settings x of X , \vec{y} of \vec{Y} , and \vec{a} of \vec{A} .

The first condition says that knowing the values of disallowed variables does not give any information about the values of sensitive variables. Now, by assumption, the allowed variables do give information about the sensitive variables (e.g., knowing the salary of an applicant gives some information about the applicant’s gender). Thus, the second condition says that knowing the values of disallowed variables does not give any information about the values of sensitive variables beyond what is given by the allowed variables. Note that information is not “additive”. The fact that the bank cannot predict the values of sensitive variables just from disallowed variables does not mean that it cannot predict the values of sensitive variables better using the allowed and disallowed variables than it could from the allowed variables alone. For example, if pet ownership (a disallowed variable) is distributed equally between women and men, but is highly correlated with salary (an allowed variable) for men and not at all for women, then pet ownership alone does not give any information about gender, but together with salary it can determine gender with a higher degree of certainty than salary alone.

While this is the high-level intuition we want to enforce, what does the regulator actually check? That is, what probability distribution is it going to use to determine independence? We believe that, in practice, the regulator will have to use the probability distribution determined by the bank’s applicants. Of course, the distribution determined by this sample may not be a completely accurate description of the distribution of the actual population (e.g., there might be some self-selection about who applies for a loan) and may not have enough data to determine all the relevant independencies. For example, for some setting \vec{y} of \vec{Y} , there may not be enough applicants that have inputs $\vec{Y} = \vec{y}$ to determine whether $X = x$ is independent $\vec{Y} = \vec{y}$. In any case, it seems unreasonable to expect complete independence in the sample; the regulator should have a threshold of acceptability.

Finally, it may be the case that for many settings $\vec{Y} = \vec{y}$ (and $\vec{A} = \vec{a}$), the bank does not have enough data to be able to reliably determine whether the relevant event are (almost) independent. The following definition is a first pass at making precise what we require, where \Pr now represents the sample distribution, $sd(X)$ is the standard deviation of X , and ϵ is some regulator-defined threshold. (The final definition is a slight generalization.)

Definition 1 (*Preliminary version:*) A system has *no disallowed proxy variables (at threshold ϵ)* if the following conditions hold:

- (a) For all sensitive variables X , for all subsets \vec{Y}' of disallowed variables, all settings x of X , and all settings \vec{y}' of \vec{Y}' such that $\Pr(\vec{Y}' = \vec{y}')$ is sufficiently large to determine statistical independence,

$$\frac{|\Pr(X = x) - \Pr(X = x | \vec{Y}' = \vec{y}')|}{sd(X)} < \epsilon.$$

- (b) For all sensitive variables X , for all subsets \vec{Y}' of disallowed variables, for all subsets \vec{A}' of allowed variables, all settings \vec{y}' of \vec{Y} , all settings x of X , and all settings \vec{a}' of \vec{A}' such that $\Pr(\vec{Y}' = \vec{y}' \cap \vec{A}' = \vec{a}')$ is sufficiently large to determine statistical independence,

$$\frac{|\Pr(X = x | \vec{A}' = \vec{a}') - \Pr(X = x | \vec{Y}' = \vec{y}' \cap \vec{A}' = \vec{a}')|}{sd(X)} < \epsilon.$$

■

The standard deviation $sd(X)$ serves as a normalizing factor here; we are computing whether using the disallowed variables gives more than an ϵ fraction of a standard deviation of extra information.

Definition 1 can be visualized as dividing the applicants into “buckets”, where each bucket corresponds to a setting of some disallowed variables, and then checking whether there are buckets that are sufficiently large to be meaningful and have a distribution of sensitive variables that is different from the whole dataset. This check is only meaningful if the bucket is large enough, which might not be the case for very many buckets. We can get a somewhat more general definition by allowing buckets to be combined. Formally, “combining two buckets” simply mean conditioning on their union. That is, rather than just conditioning on $\vec{Y}' = \vec{y}'$ in Definition 1, we consider subsets $\vec{Y}^1, \dots, \vec{Y}^k$ and values $\vec{y}^1, \dots, \vec{y}^k$, and condition on $(\vec{Y}^1 = \vec{y}^1 \cup \dots \cup \vec{Y}^k = \vec{y}^k)$ (or $(\vec{Y}^1 = \vec{y}^1 \cup \dots \cup \vec{Y}^k = \vec{y}^k) \cap \vec{A}' = \vec{a}'$ in part (b)). We take this to be the official definition of having no disallowed proxy variables. Note that an important special case of this is *abstracting values*. For example, if Y is the variable *age*, rather than just conditioning on $age = 37$, we can condition on the range $age \in \{30, \dots, 40\}$ (which is just $age = 30 \cup \dots \cup age = 40$).

Certifying a system as fair: To certify a system as fair, the regulator must conduct a number of checks. The first few involve the bank’s input variables. We already discussed an important check above: checking that the system has no

disallowed proxy variables. The regulator must also check that the system input variables are being used as the bank claimed that they were. Recall that we require the bank to reveal how the input variables are obtained. The regulator will need to check whether all inputs are obtained appropriately. For example, is the bank allowed to scrape social media posts? The regulator should be able to check fairly easily that the information can be obtained just as the bank claims, and then check that all the input variables are indeed computed as they should be. To understand why this is critical, consider the following example.

Example 1 Since it gets salary information in many different currencies, the bank convinces the regulator that, not only should *salary* be allowed, but it should be able to convert all information regarding salary to internal units of currency (according to agreed-upon conversion rates). But in doing the conversion, the bank slightly modifies the salary, replacing the low-order number by either 0 or 1, depending on whether the applicant is male or female. For example, a salary of 87,325 (in the bank’s internal units) would become either 87,320 or 87,321, depending on whether the applicant is male or female. This means that the bank can completely base its decision on gender. This is precisely why the regulator needs to know how all the input variables in the bank’s system are calculated from data. If the regulator knows this, she should be able to spot the discrepancy above. But this will clearly require an alert regulator! ■

Finally, the regulator must check that there are no inputs being used other than those listed by the bank and that the endogenous sensitive variables are computed correctly (according to the equations specified by the regulator). As we said, we assume that the regulator has access to the input data for all applicants. (It actually suffices that she can get data for a reasonably large random subset of applications.) To ensure that she is testing *all* the relevant variables in the tests discussed above, the regulator can test that setting the inputs appropriately gives the decision taken by the bank. Similarly, she can check that she gets the expected answers for endogenous sensitive variables. The fact that the bank will be monitored in this way should suffice to prevent it from using undeclared inputs or altering the equations of endogenous variables.

With all these tests of the input variables out of the way, the regulator can now check that there is no discrimination in the more standard sense, namely, checking whether changing the values of sensitive variables has any impact on the decision, once we fix the allowed attributes. This is a way of making precise a claim like “gender has no impact on the decision, beyond its impact on allowed variables (such as *salary*)”. Since we assumed that the set of sensitive variables is closed under ancestors, it suffices to ensure that changing the values of exogenous sensitive variables has no impact on the decision. (Indeed, we may not want to simultaneously change the values of two sensitive values, one of which is a child of another, if doing so would result in an inconsistent situation.) Consider, for example, a model in which there is an endogenous sensitive variable *goes to church regularly*, with range {Y,N}, and its exogenous parent *religious affiliation*,

with range corresponding to the common religious affiliations. According to our observation, the *religious affiliation* variable is also sensitive, and changing both at the same time does not make sense, as it might lead to inconsistent situations, such as *religious affiliation=Judaism* and *goes to church regularly=Y*.

Definition 2 A model M and a causal variable D of M are *fair* with respect to a set \vec{Y} of allowed variables and a set \vec{X} of exogenous sensitive variables if, for all contexts \vec{u} , changing the values of the sensitive variables has no effect on the value of D if the allowed variables are fixed to their values in \vec{u} .² ■

Our definition differs from other causal definitions of fairness (e.g., (Kilbertus et al. 2017; Kusner et al. 2017; Loftus et al. 2018)) in one significant respect. Other definitions of fairness are statistical. They require only that the probability of the decision D having a certain value is the same for all settings of the sensitive variables. This difference is mainly due to our application. We assume that the values of all the exogenous variables are known (since they represent inputs to the bank’s system); in the other papers, it is assumed that all that is known about the contexts is their probability. Given that we take the values of exogenous variables to be known, we believe that our choice is appropriate for our application.

Dealing with complaints: Suppose that the bank’s system is certified as fair, yet someone brings a complaint of discrimination. The bank should be able to provide all the inputs used for that person. The regulator can verify that all input variables were computed appropriately, that all the endogenous sensitive variables get the appropriate values, and that the bank’s software really does produce the result claimed by the bank for these values. If, despite this, the regulator finds that the complaint has merit, she can then see the affect of disallowing some allowed variables, to try to pinpoint what is causing a perhaps undesirable result. We anticipate that complaints may result in pressure to disallow some allowed variables.

Changing the status of variables: While the AI system is created and maintained by the bank, variables are defined as sensitive or allowed by the regulator; their status may change over time. For example, the Equal Credit Opportunity Act (ECOA) of 1974 prohibited creditors from discrimination on the basis of race, color, religion, national origin, sex, marital status, or age, thus making these attributes sensitive variables. Not all cases of such changes require re-certification, but some do. It is fairly straightforward to see that declaring a previously non-sensitive variable sensitive can render a previously fair system unfair. Indeed, this probably happened with many bank systems in 1974. It is also easy to see that if a previously sensitive variable is declared non-sensitive, then a system that was previously fair continues to be fair (and a system that was unfair may become fair).

²This can be expressed formally in the logic of causality (Halpern 2016) as: there is a value d in the range of D such that for all settings \vec{x} of \vec{X} , we have $(M, \vec{u}) \models [\vec{X} \leftarrow \vec{x}; \vec{Y} \leftarrow \vec{y}] (D = d)$, where \vec{y} are the values of \vec{Y} in context \vec{u} ; that is, $(M, \vec{u}) \models \vec{Y} = \vec{y}$.

The effect of changing the status of allowed variables is somewhat less obvious. In fact, both changing the status of a previously disallowed variable to allowed and making a previously allowed variable disallowed can change the status of the system from fair to unfair or the other way around. Consider a loan application system with two binary exogenous variables: a sensitive variable *gender*, with values $\{M, F\}$, and a non-sensitive variable (loan application) *amount*, with values $\{\text{low}, \text{high}\}$ (we make both variables binary for ease of exposition). The endogenous variable *salary* has, again, two values, $\{\text{low}, \text{high}\}$, and the decision is “yes” if *amount=low* or *salary=high*. Now we assume that the equation for *salary* sets *salary* to *low* if *gender=F* and to *high* otherwise. If *salary* is not an allowed variable, the system is clearly unfair: in the context of a woman applying for a high loan amount, the decision is “no”, yet toggling the gender to *M* changes the decision to “yes”. Changing the status of the *salary* variable to allowed, however, makes the system fair, as the gender only affects the decision via salary.

Perhaps a more surprising observation is that making a previously disallowed variable allowed can make a previously fair system unfair. Suppose that we add a new endogenous binary variable *impulsivity* to the system above, which is *low* if *gender=F* and *high* otherwise, and change the equation for the decision to be “yes” if either *salary=high* or *impulsivity=low*. It is easy to see that the system approves all loan applications, and if there are no allowed variables, it is fair. If *salary* now becomes an allowed variable, the system stops being fair: if *gender=F* and we toggle *gender* while keeping *salary* fixed to *low*, *impulsivity* becomes *high*, and the loan is not approved.

4 Complexity

Clearly, for the regulator to certify a system, she will have to be able carry out all the checks in a reasonable amount of time. We assume that the regulator can run the bank’s software on a specific input (i.e., for a particular applicant) to see what the outcome would be, and do so in polynomial time. The following results seems to suggest that checking for fairness will be difficult. Importantly, they hold even if there are relatively few sensitive variables (which is likely to be the case in practice).

Theorem 4.1 *Deciding if a system M is fair with respect to sensitive variables \vec{X} and allowed variables \vec{Y} is co-NP-complete. (More precisely, if L_{fair} is the language consisting of all tuples $\langle M, D, \vec{X}, \vec{Y} \rangle$ such that the model M and a causal variable D of M are fair with respect to the set \vec{X} of sensitive variables and set \vec{Y} of allowed variables, then L_{fair} is co-NP-complete.) This is true even if the number of settings of exogenous sensitive variables is bounded.*

Proof. To see that checking for fairness is in co-NP, it suffices to check that the complementary problem, is in NP. To check for unfairness we simply have to guess a setting of the exogenous variables (which amounts to guessing the features of an applicant), and guess two settings of the exogenous variables that give different values for D .

To show that checking fairness is co-NP hard, we reduce the problem of checking whether a propositional formula ϕ is valid to the problem of checking fairness. Given a propositional formula ϕ whose primitive propositions are X_1, \dots, X_n , construct a causal model M_ϕ where the non-sensitive binary input variables are X_1, \dots, X_n , there is only one sensitive exogenous binary variable, X_0 , only one endogenous variable, D , and no allowed variables. The equation for D is $D = 1$ if $X_0 = 0$, and $D = \phi$ if $X_0 = 1$. Since there are no allowed variables, this system is fair iff $\phi = 1$ (i.e., ϕ is true) for every setting of the variables X_1, \dots, X_n . But this is the case iff ϕ is valid. ■

As complaints would typically originate from one perceived case of discrimination, the regulator might have an easier task checking a complaint than certifying the whole system. Checking fairness with respect to a specific context can have lower complexity than checking fairness of the system in general. In order to reason about this complexity formally, we introduce the following definition of fairness with respect to a specific case (context).

Definition 3 A model M and a causal variable D of M are *fair* with respect to a set \vec{Y} of allowed variables and a set \vec{X} of exogenous sensitive variables in a context \vec{u} if changing the values of the exogenous sensitive variables has no effect on the value of D when the allowed variables are fixed to their values in \vec{u} . ■

Here there is some good news. Although the problem continues to be co-NP-complete, the co-NP-completeness stems completely from the number of possible settings of the sensitive variables (since we have to check that the value of the decision variable is unaffected if we change the values of the sensitive variables). If we assume, as will almost certainly be the case in practice, that there are relatively few sensitive variables and that they have relatively few values, we can do a brute force check in polynomial time.

Proposition 4.2 *Let $L_{\text{fair}}^{\vec{u}}$ be the language of all tuples $\langle M, D, \vec{X}, \vec{Y} \rangle$ such that the model M and a causal variable D of M are fair with respect to the set of sensitive variables \vec{X} and the set of allowed variables \vec{Y} in context \vec{u} . For a fixed context \vec{u} , the decision problem for $L_{\text{fair}}^{\vec{u}}$ (i.e., determining whether the system is fair for the applicant represented by \vec{u}) is co-NP-complete, but is polynomial in the number of settings of the exogenous sensitive variables.*

Proof. Given a context, we can consider all settings of the exogenous sensitive variables, and ensure that the decision value D has the same value in all contexts resulting from these settings. This is clearly polynomial in the number of settings of the exogenous sensitive variables.

If there is no bound on the number of sensitive variables, then the problem is still clearly in co-NP (this is a special case Theorem 4.1). To show co-NP hardness, we again reduce the validity problem to the problem of checking fairness. Given a propositional formula ϕ whose primitive propositions are X_1, \dots, X_n , we construct a causal model M_ϕ , where the only exogenous variables are the sensitive variables X_0, \dots, X_n , where X_1, \dots, X_n are the variables

of ϕ and X_0 is a fresh variable, and there is one endogenous variable D , whose equation is $X_0 \vee \phi$. This system is fair iff D has the same value for all settings of the exogenous variables. If $X_0 = 1$, then $D = 1$, so we must also have $D = 1$ if $X_0 = 0$. But this means that for all sets of X_1, \dots, X_n , $D = 1$. This is the case iff ϕ is valid. ■

Proposition 4.2 already suggests why in practice, the co-NP-completeness of checking fairness will not be a big problem in practice, assuming that the number of settings of sensitive variables is small. Checking fairness for a particular individual can be done quickly. Thus, the regulator can easily sample a relatively large number of applicants and verify that fairness holds for all of them (more precisely, she would verify that fairness holds of the context determined by each applicant’s inputs). Why is this compatible with Theorem 4.1? To verify that the formula is valid, we must check *all* possible settings of the primitive propositions in the formula. If the bank’s system uses, say, 1000 input variables, even if they are binary, there are 2^{1000} settings of these variables, far more than the number of applicants. We care only about the settings that actually arise for applicants.

There is another check that the regulator must perform: checking that there are no disallowed proxy variables. Again, we believe that this will not be a problem in practice. Note that whether there are disallowed proxy variables depends on features of the applicants; that is, it is not an intrinsic property of the causal graph, but a property of the data. We believe that, given n applicants, or, more precisely, given all the input variables, including exogenous sensitive variables, for n applicants, we should be able to check in time polynomial in n whether there are disallowed proxy variables. We believe that, specifically, we should be able to write a machine learning program to see if the disallowed input variables give information about the sensitive variables, or if the disallowed and allowed input variables give information beyond that given just by the allowed variables (above threshold ϵ).

There is a concern that the bank might have a better machine learning program than the regulator, so that the regulator might not detect any correlation between the disallowed variables and the sensitive variables, but the bank’s program can. This is clearly a topic that requires further investigation.

We remark that the other checks that the regulator needs to carry out, checking that the input variables are computed correctly, that there are no input variables beyond those explicitly given by the bank, and that the endogenous sensitive variables are computed correctly, can clearly be carried out in polynomial time. For the second check, we need to assume that the regulator can check, for a random subset of applicants, that if she sets the input variables that she was told about appropriately, then she gets the same outcome as the bank did. (If there were additional random variables that had a nontrivial effect on the outcome, then this would not be the case.)

On what dataset should the regulator run the checks we describe in the paper? We expect there to be a wealth of historical data that is used by the bank to train its AI system. The regulator can request the same training set as the bank uses and run the initial checks on that set. The regula-

tor should then request all the applicant data after the bank starts running its system, and do periodic checks on (a sample of) that data. Note that the bank can try to fool the regulator initially, by omitting applicants from the dataset that would demonstrate that there are disallowed proxy variables. But as long as the regulator has access to all the applicants, that problem should be spotted relatively quickly. And if the bank does not share all the applicant data, this will be discovered when someone complains. We assume that there is a system of fines and sanctions that would discourage this type of “cheating”. Of course, it is possible that the bank’s initial dataset is not representative of later data for legitimate reasons. For example, there may be changes in the legal process for applying for a loan application. But then we would expect the bank to have to (and be able to) justify why the initial dataset is not representative.

As these results suggest, regulators should be able to certify a bank’s system in a reasonable amount of time, despite the initially discouraging complexity results, although more work needs to be done to develop algorithms for verifying that a system has no proxy variables.

5 Conclusions

Assuring fairness of AI algorithm is a relatively recent subject, but it has already attracted a lot of attention, due to the ever-increasing use of AI to make decisions. We believe that there will be pressure to regulate this activity. Companies may even welcome this regulation, to avoid getting sued for their practices. Indeed, a number of large companies recently released packages to detect certain types of unfairness in the form of bias or under-representation (e.g. IBM’s AI Fairness 360 (IBM 2018) and Facebook Fairness Flow (Facebook 2021)). These packages are not a general attempt to provide a regulatory framework; they have tailor-made routines to check for particular types of discrimination. But this does demonstrate that industry is aware of the problem and is taking preliminary steps. There has also been work on discovering discrimination against individuals (Zhang, Wu, and Wu 2016; Bonchi et al. 2017; Kilbertus et al. 2017; Zhang, Wu, and Wu 2019). (Recall that in our setting, this can be checked easily.)

In this paper, we make a first attempt to define such a regulatory framework, with definitions and criteria that can be verified and supported by evidence. While the worst-case complexity of certifying fairness may appear high, in practice, we expect that the certification process will be quite fast and efficient. Of course, not everyone will agree with all the choices we have made here, and some may feel that more (or less) regulation should be required. We welcome discussion of these issues. We believe that it is important for the AI community to take the lead here, and help guide policy-makers in coming up with ways to certify software as acceptable. We hope that our work provides a useful first step in this direction.

6 Acknowledgments

Chockler was supported in part by the UKRI Trust-worthy Autonomous Systems Hub (EP/V00784X/1) and the UKRI Strategic Priorities Fund to the UKRI Research Node on Trustworthy Autonomous Systems Governance and Regulation (EP/V026607/1). Halpern was supported in part by NSF grants IIS-178108 and IIS-1703846 and MURI grant W911NF-19-1-0217.

References

- Agan, A.; and Starr, S. 2018. Ban the box, criminal records, and racial discrimination: a field experiment. *Quarterly Journal of Economics*, 133(1): 191–235.
- Bonchi, F.; Hajian, S.; Mishra, B.; and Ramazzotti, D. 2017. Exposing the probabilistic causal structure of discrimination. *International Journal of Data Science and Analytics*, 3(1): 1–21.
- Chiappa, S. 2019. Path-specific counterfactual fairness. In *Proc. Thirty-Third AAAI Conference on Artificial Intelligence (AAAI-19)*, 7801–7808.
- Datta, A.; Tschantz, M. C.; and Datta, A. 2015. Automated Experiments on Ad Privacy Settings. *Proceedings on Privacy Enhancing Technologies*, 2015(1): 92–112.
- Facebook. 2021. <https://ai.facebook.com/blog/how-were-using-fairness-flow-to-help-build-ai-that-works-better-for-everyone/>.
- Halpern, J. Y. 2016. *Actual Causality*. Cambridge, MA: MIT Press.
- Halpern, J. Y.; and Pearl, J. 2005. Causes and explanations: a structural-model approach. Part I: Causes. *British Journal for Philosophy of Science*, 56(4): 843–887.
- IBM. 2018. <https://www.ibm.com/blogs/research/2018/09/ai-fairness-360/>.
- Kilbertus, N.; Rojas-Carulla, M.; Parascandolo, G.; Hardt, M.; Janzing, D.; and Schölkopf, B. 2017. Avoiding discrimination through causal reasoning. In *Advances in Neural Information Processing Systems (NIPS '2017)*, 656–666.
- Kleinberg, J.; Lakkaraju, H.; Leskovec, J.; Ludwig, J.; and Mullainathan, S. 2018a. Human decisions and machine prediction. *The Quarterly Journal of Economics*, 133(1): 237–293.
- Kleinberg, J.; Ludwig, J.; Mullainathan, S.; and Sunstein, C. 2018b. Discrimination in the Age of Algorithms. *Journal of Legal Analysis*, 10: 113–174.
- Kusner, M. J.; Loftus, J. R.; Russell, C.; and Silva, R. 2017. Counterfactual fairness. In *Advances in Neural Information Processing Systems 30*, 4069–4079.
- Loftus, J. R.; Russell, C.; Kusner, M. J.; and Silva, R. 2018. Causal reasoning for algorithmic fairness. Available at <https://arxiv.org/pdf/1805.05859.pdf>.
- Nabi, R.; and Shpitser, I. 2018. Fiar inference on outcomes. In *Proc. Thirty-Second AAAI Conference on Artificial Intelligence (AAAI-18)*.
- Primus, R. A. 2003. Equal protoection and disparate impact: round three. *Harvard Law Review*, 117(2): 494–587.
- Prince, A. E. R.; and Schwarcz, D. 2020. Proxy discrimination in the age of Artificial Intelligence and big data. *Iowa Law Review* 1257, 105: 1257–1318.
- Zhang, L.; Wu, Y.; and Wu, X. 2016. Situation Testing-Based Discrimination Discovery: A Causal Inference Approach. In Kambhampati, S., ed., *Proceedings of the 25th International Joint Conference on Artificial Intelligence, IJCAI*, 2718–2724. IJCAI/AAAI Press.
- Zhang, L.; Wu, Y.; and Wu, X. 2019. Causal Modeling-Based Discrimination Discovery and Removal: Criteria, Bounds, and Algorithms. *IEEE Trans. Knowl. Data Eng.*, 31(11): 2035–2050.