Algorithmic Bias: On the Implicit Biases of Social Technology

Gabbrielle M Johnson

Abstract

Often machine learning programs inherit social patterns reflected in their training data without any directed effort by programmers to include such biases. Computer scientists call this *algorithmic bias*. This paper explores the relationship between machine bias and human cognitive bias. In it, I argue that they are of the same basic kind and that by adopting a functional model that extends to both cases, we gain two advantages over extant models of bias. First, adopting a functional model captures a heretofore neglected possibility of human cognitive bias: those that influence an individual's beliefs about and actions toward other people, but are, nevertheless, nowhere represented in that individual's cognitive repertoire. Second, adopting a functional account allows for robust predictive and explanatory exchange between the machine and cognitive domains. I end by demonstrating this in the case of mitigation techniques, explaining one reason human implicit biases resist revision: cognitive biases, like machine biases, can rely on proxy attributes.

1 Introduction

On March 23, 2016, Microsoft Corporation released Tay, an artificial intelligence (AI) Twitter chatbot intended to interact with other Twitter users and mimic the language patterns of a 19-year-old American girl. Tay operated by learning from human Twitter users with whom it interacted. Only 16 hours after its launch, Tay was shut down for authoring a number of tweets endorsing Nazi ideology and harassing other Twitter users. Among the inflammatory tweets were those saying "Hitler was right," those endorsing then-Republican-nominee Donald Trump's proposal that "we're going to build a wall," various derogatory remarks about feminists, as well as claims that "9/11 was an inside job." When asked about how Tay developed such a noxious personality, Microsoft responded "As [Tay] learns, some of its responses are inappropriate and indicative of the types of interactions some people are having with it."[1] In other words, Tay's personality was inherited from the individuals with whom it was engaging.

The story of Tay highlights an obstacle facing developers of machine learning programs: implicit *algorithmic* bias. An AI like Tay, which uses machine learning to capitalize on (or "learn" from) statistical regularities in human-generated data-sets, tends to pick up social patterns that manifest in human behavior and that are reflected in the data on which it is trained. In many of these cases, we have reason to suspect that programmers are not explicitly writing biases toward marginalized demographics into their software's code.[2] Instead, it appears the biases in some sense *implicitly emerge* from the algorithms' operating on the data, mimicking the biases reflected in the data themselves.

Because algorithmic biases don't utilize explicitly represented bias rules, cases such as Tay's are not naturally accommodated by extant models of implicit cognitive bias, which either fail to fully consider or mischaracterize biases whose components are not explicitly represented. In contrast, the functional model of implicit bias that I develop accommodates both representational and non-representational biases. In this paper, I argue that my functional model has two advantages over these other accounts.

First, models of social bias (hereafter just 'bias') take human cognitive bias in the form of mental representations, e.g., stereotypes or implicit attitudes, as the paradigmatic case; but, it may turn out that some human cognitive bias is, like Tay's algorithmic bias, non-representational.[3] To motivate this possibility, I begin by presenting various cases in which algorithmic biases mimic patterns of human implicit bias. Next, I argue that a machine learning program need not make use of explicit rules representing stereotypes for the use of that program to result in biased output patterns. Next, I argue that the same may apply to human implicit bias, i.e., some cognitive biases influence an individual's beliefs about and actions toward other people, but are, nevertheless, nowhere represented in that individual's cognitive repertoire. I call these *truly implicit biases*. I then demonstrate how

[1] Price 2016.

[2] See, for example, O'Neil (2016, 154)'s discussion of discriminatory errors in Google's automatic photo-tagging service.

[3] Prominent models that fall into this category include any theory that posits representations at the core of a bias's operation. These include various associative accounts including Gawronski and Bodenhausen (2014)'s Associative-Propositional Model (APE), Gendler (2008)'s treatment of implicit biases as *aliefs*, Holroyd (2016)'s Minimal Model, and Madva and Brownstein (2016)'s intrinsic affect-laden stereotype model; as well as various propositional accounts including Mandelbaum (2015)'s Structure Belief Hypothesis, De Houwer (2014)'s Propositional Account, Levy (2015)'s view that implicit biases are "patchy endorsements", and arguably Fazio (1990)'s Motivations and Opportunity as Determinants Model (MODE). However, I won't argue for the insufficiencies of these accounts here.

a functional model of implicit bias that I develop in previous work can be straightforwardly extended to handle both cases of non-represented biases.

Second, my functional account allows for robust predictive and explanatory exchange between the algorithmic and cognitive domains, independent of whether the biases of these domains are representational.[4] I conclude by using the comparison of these two cases to demonstrate one plausible explanation for why human implicit biases resist revision. In cases of algorithmic biases, programmers have long struggled with the difficulty of eliminating biases that are based on so-called ‘proxy attributes’: seemingly innocuous attribute labels that correlate with socially sensitive attributes, serving as proxies for the socially-sensitive attributes themselves. For example, in the historic cases of discriminatory redlining, zip codes were used as proxies for race.[5] Crucially, the effects of these proxy attributes tend to resist any overt filtering techniques. Eliminating any explicit references to race in a program’s code will not ameliorate the harms it causes since the program can simply substitute a proxy attribute in place of race, resulting in similar discriminatory effects. Likewise, one might think that human implicit biases similarly resist revision since most attempts to revise them focus on overt, socially-sensitive attributes rather than potential proxy attributes. Better understanding of these attempts to identify the operation of proxy attributes in the algorithmic case will plausibly lend to a better understanding of mitigation techniques for human implicit biases. An account of implicit bias on which both algorithmic bias and human cognitive bias are of the same basic kind anticipates and facilitates the fruitfulness of these comparisons.

2 Evidence for Algorithmic Bias

I’ll begin by surveying some of the evidence that suggests machine biases mimic typical bias patterns found in human implicit biases.

Consider a study by Caliskan et al. (2017) on word-embedding machine learning. This study found that parsing software trained on a dataset called “the common crawl”—an

[4] This account borrows important insights from dispositionalist approaches to belief (e.g., those presented by Ryle (1949), Dennett (1981), and Schwitzgebel (2002)) and, regarding models of implicit bias, most resembles the trait approach presented by Machery (2016) and the indeterminate content approach presented by Yumusak (2017). However, this view also differs from each of these approaches since, as I’ll explain, it is committed to very restricted forms of disposition, namely those that are underlain by combinations of states and processes that systematically relate propositional state inputs to propositional state outputs.

[5] Massey and Denton 1993, 51 ff..

assemblage of 840 billion words collected by crawling the internet—resulted in the program producing “human-like semantic biases” that replicated well-known trends in results of indirect measures for human implicit biases. These biases included the tendency to more often pair stereotypical female names with family words than career terms, stereotypical African-American names with unpleasant words rather than pleasant words, and stereotypical male names with science and math terms rather than art terms. A plausible explanation of this phenomenon comes from looking at the patterns within the training data themselves, i.e., patterns in online language use. For example, computer scientist Seth Stephens-Davidowitz’s analyses of Google data trends show people are two and a half times more likely to google “Is my son gifted?” than “Is my daughter gifted?”. This suggests that online texts encode human social biases that often associate males with inherent intelligence.[6] It seems that the word-embedding software picked up on and mimicked these patterns. On this result, co-author Arvind Narayanan writes, “natural language necessarily contains human biases, and the paradigm of training machine learning on language corpora means that AI will inevitably imbibe these biases as well.”[7]

Examples of similar phenomena abound. In 1989, St George’s Hospital Medical School’s Commission for Racial Equality found a computer used for initial screenings of applicants “written after careful analysis of the way in which the staff were making these choices” unfairly rejected women and individuals with non-European sounding names.[8] Similarly, a study by Datta et al. (2015) found that Google’s ad-targeting software resulted in “males [being] shown ads encouraging the seeking of coaching services for high paying jobs more than females.” And finally, a study by Klare et al. (2012) demonstrated that face recognition software, some of which is used by law enforcement agencies across the US, use algorithms that are consistently less accurate on women, African-Americans, and younger people.[9] As the use of machine learning programs proliferates, the social consequences of their biases become increasingly threatening.[10] One particularly jarring example of these consequences was highlighted by a 2016 ProPublica analysis that revealed software being utilized across the country to predict future criminals is biased against black people, often associating African American attributes with crime, a common result found in implicit

[6] Stephens-Davidowitz 2014.

[7] Narayanan 2016.

[8] Lowry and Macpherson 1988.

[9] See also Wu and Zhang 2016.

[10] For a comprehensive overview of the current state of affairs regarding machine learning programs in social technology, see O’Neil 2016.

bias measures.[11] The algorithm was nearly twice as likely to falsely flag black defendants as future criminals than white defendants, while inaccurately labeling white defendants as low risk more often than black defendants.[12]

These examples demonstrate that machine bias exists and that the patterns of such bias mimic well-known implicit bias patterns in humans. However, we needn't from the existence of these biases infer that programmers are writing explicitly racist code. Instead, it's possible that such biases emerge non-representationally out of the operation of seemingly innocuous code paired with statistical regularities of the training data. These cases of algorithmic bias can be demonstrated with a toy model using a simple k-nearest neighbors algorithm, which I turn to next.

2.1 Implicit Algorithmic Bias

I'll now present a simple walkthrough of how machine learning programs operate.[13] Machine learning programs come in two basic forms: supervised learning and unsupervised learning. In what follows, I focus on the simpler cases of supervised learning programs, since their operation is more intuitive and differences between the two aren't important for my purposes. There are two main stages of a supervised learning program's operation: first a training phase, followed by a test phase. During the training phase, the program is trained on pre-labeled data. This affords the program the opportunity to "learn" the relationships between features and labels. The second stage is applying the resulting predictive model to test data, which outputs a classification for each new test datum on the basis of its features.

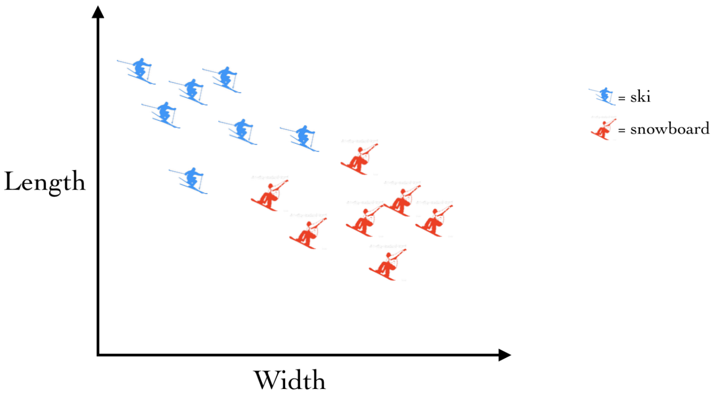
For example, imagine that you're creating a program that you intend to use for the simple classification task of predicting whether an object is a ski or a snowboard. You might begin by identifying features of skis and snowboards that you think are relevant

[11] See, for example, Eberhardt et al. 2004.

[12] Angwin et al. 2016.

[13] Ideally, I would present a walkthrough of the operation of one of the algorithms discussed above. Unfortunately, providing a detailed analysis of one of these algorithms is difficult, if not impossible, since information relating to the operation of commercial machine learning programs is often intentionally inaccessible for the sake of competitive commercial advantage or client security purposes. Even if these algorithms were made available for public scrutiny, many likely instantiate so-called 'black-box' algorithms, i.e., those where it's difficult if not impossible for human programmers to understand or explain why any particular outcome occurs. This lack of transparency with respect to their operation creates a myriad of concerning questions about fairness, objectivity, and accuracy. However, these issues are beyond the scope of this discussion.

for determining an object's classification into either category. In theory, you can choose an indefinite number of relevant features.[14] For our purposes, we'll focus on just two: length and width.[15] We begin stage one by training our program on pre-labeled data called 'training data'. These include many instances of already-categorized objects. We can represent the relationships between the relevant features and classifications for the known data by plotting them on a two-dimensional features space as follows:

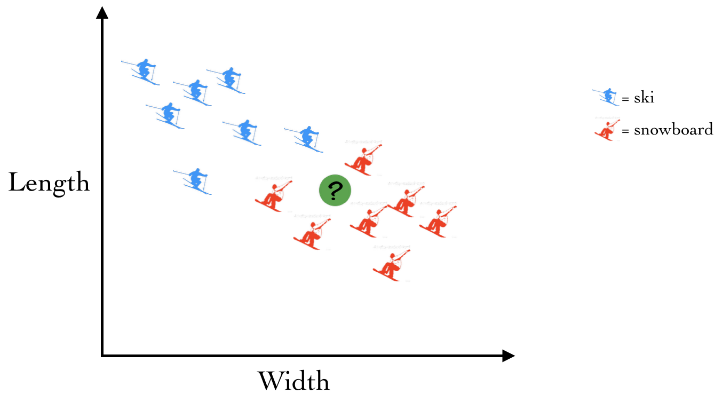


Each data-point specifies two things: the *feature values*, i.e., values corresponding to its length and width; and a *class label*, i.e., a label corresponding to its classification as a ski or snowboard. Per the diagram, the objects in the test data that are longer but not as wide tend to be skis, while the objects that are wider but not as long tend to be snowboards.

In the next phase of the program, the algorithm gets applied to new, unclassified data called 'test data', and the aim of the program is to classify each datum as either a ski or a snowboard on the basis of its feature values. One way for the program to do this is to classify new instances on the basis of their proximity in the feature space to known classifications. For example, say we had a new object that we didn't know was a ski or a snowboard, but we did know had a certain length and width. The program could then plot this new instance on the feature space based on its length and width as follows:

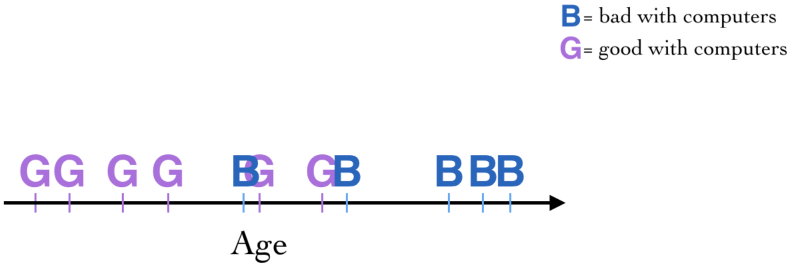
[14] The problem of which features are most relevant for some classification test is an interesting and complicated question from a computer science perspective that is unfortunately beyond the purview of this paper.

[15] This case is discussed in much more detail by Daum III (2015, 30-32).



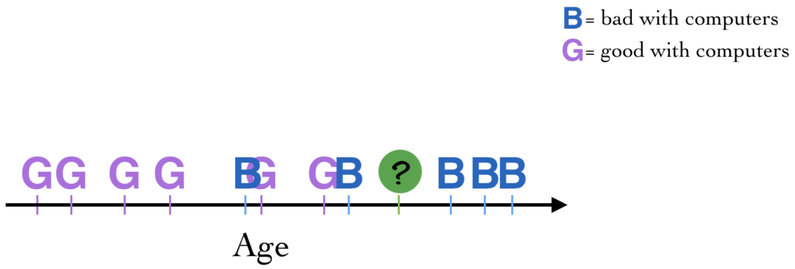
The program then classifies it based on its relationship to the other data points. Many different methods are used to do this, but one intuitive method is to simply classify based on a majority vote of its k-nearest neighbors. If we set k to 5, then the program will decide on the basis of the five nearest neighbors in the feature space. In this case, the five nearest neighbors comprise one ski and four snowboards. Since the majority vote in this case results in snowboards, the program classifies the test instance as a snowboard.

This same method can be used for any number of classification tasks based on relevant attributes, including those overlapping with stereotypical judgements based on members of marginalized social groups. For example, imagine an engineer creating a program that classifies individuals as good or bad with computers. Let's say she thinks one relevant property for determining this classification is a person's age. Thus, she trains the program on many instances of individuals she labels as a certain age and as either good or bad with computers. The following is an example of how these instances might end up plotted on a one-dimensional feature space:



Per the diagram, the data in this example are skewed: those individuals that are bad with computers are clustered near the end of the line, while those good with computers near the beginning. There are many ways test data can come to be patterned like this. For example, imagine that the programmer pulled training data from a local library on the same day that that library was hosting an after-school charity event during which high-schoolers provide social media training to residents of a local assisted living facility. In this case, we would expect her sample to be disproportionately filled with tech-savvy young people and elderly individuals who struggle with computer technology (as compared to the general population). Thus, our training data can be skewed by a failure to collect a representative sample of the population.[16] Although seemingly contrived, this example demonstrates an important lesson: a machine learning program is only as good as the data on which it is trained, giving rise to the oft-cited motto "garbage in, garbage out." If the data going into the training period are biased, then we can expect the generalizations the program makes based on those data to be biased as well.

This is precisely what we see when we again apply our k-nearest neighbors algorithm:



In this case, the five nearest neighbors to the target will include a majority of individuals who are bad with computers. Thus, the new individual will, on the basis of their age, be labeled as bad with computers.

With those individuals who are older patterning roughly with those bad with computers (and vice versa for young individuals and good with computers), we could think of this entire data-set as "reflecting" the social generalization (or, loosely, the stereotype) that

[16] The data can come to reflect biases in a variety of other ways too, e.g., if they were labeled by individuals with personal biases. If the person labeling the training data in our case was—due to her own biases and prejudices—more likely to label elderly individuals as bad with computers even when they weren't, then we should still expect a mismatch between the training data and the real world. Moreover, even randomly selected data might reflect social biases if those biases are ubiquitous in the environment. For example, if it turns out that in general, the elderly are statistically more likely to be bad with computers (a trend that might itself be the result of structural and societal discrimination against them), then even carefully collected data might reflect this bias. This begins to highlight the dynamic relationship between machine bias, human cognitive bias, and structural bias.

elderly people are bad with computers, though that content is never explicitly represented as a rule in the program's code.[17] Instead, the algorithm operates on a simple Euclidean calculation involving the proximity of the test instance to the training instances in the feature space. Rather than the stereotype being explicitly represented as a rule, it *implicitly emerges* out of the distribution of training instances in the feature space.

One might worry that the content *elderly people are bad with computers* is arguably represented by the data set, say as the decision boundary for the algorithm, despite there not being any syntactic elements in the program's code that correspond precisely to that content. A proper rebuttal to this point would require more space than is available here. Instead, I'll merely gesture at two possible reasons for thinking it's not. First, the toy model I've presented here is (for the ease of exposition and understanding) much simpler than real-world machine learning programs. In the range of more complicated cases that make up actual machine learning programs, e.g., those relying on high-dimensional feature spaces that encode collections of a great number of feature values, it seems less plausible that the system is explicitly representing simple stereotype-like rules that might, nevertheless, be apt in describing its operation. Secondly, as a more intuitive point, notice that if we were to include in the program's code an explicit rule with content *it's not the case that elderly people are bad with computers*, there isn't *obviously* a contradiction anywhere in the program, e.g., as there might be if we also programmed as a rule *elderly people are bad with computers.*[18]

Most important, even in cases where the stereotype isn't represented in program in the same way stereotypes that are directly written into a program's code would be, the program still operates roughly *as if* it represented such a rule: it still classifies individuals as being bad with computers on the basis of their age, with elderly individuals being more likely to be labeled as bad-with-computers. I'll call biases that operate in this way *truly implicit biases*, and algorithmic biases—such as the one that emerges from the example

[17] The notion of stereotype that I utilize in this paper is purposely neutral with respect to certain features that are commonly assumed in discussions of stereotypes. For example, my notion doesn't assume that stereotypes are pernicious or morally problematic, nor that they are false or inaccurate. Here, I follow Barnes (2016, 11-12) in thinking that a metaphysical account of some social phenomenon—in her case, disability; in mine, bias—ought not prejudge the normative issues surrounding the phenomenon. For a more detailed discussion of the notion of 'stereotype' and how it gets employed in psychological and philosophical discussions, as well as arguments in favor and against including normative and accuracy conditions in its definition, see Beeghly 2015 and footnote 23.

[18] These two considerations suggest that, at the very least, there exists an important distinction in representational status between the two ways a generalization might be present in the program.

above—are one form such biases can take.

In what follows, I'll draw an analogy of this case to the case of implicit human cognitive biases and argue that, as with algorithmic bias, cognitive biases may be truly implicit. Given this possibility, it will be critical to construct a model that recognizes both cases of bias.

2.2 Implicit Cognitive Bias

Before constructing such a model, it helps to get clear on the various components of a cognitive bias's operation, provide some terminology for those components, and show how those components correspond to the components of the algorithmic cases above. I start with a case of explicit bias, where the inner-workings of the bias are available to introspection. This affords us the opportunity to scrutinize and examine the role of each component.

Imagine a fellow academic (call him 'E') attempts to help his colleague Jan join a Skype interview. When asked why he did this, E explains that it is because he believes that Jan is bad with computers. When asked why he believes *this*, he explains that it's because he believes that Jan is elderly and that elderly people are bad with computers. Here, the inference E is making is straightforward:

(i) Jan is elderly.
(ii) Elderly people are bad with computers.
$\therefore$ (iii) Jan is bad with computers.

This case clearly involves an explicit bias: E is completely aware that he is drawing conclusions about Jan based on his beliefs about the elderly and Jan's belonging to that group.

Confusingly, the term 'bias' is often ambiguous between a stereotype belief, the conclusion of some inference involving a stereotype belief, and the behavior based on the conclusion of some inference involving a stereotype belief.[19] To keep these components distinct, I use different terms for each. First, I call the belief about a particular person on the basis of which a discriminatory judgment is formed, in this case, E's belief that Jan is elderly, *the bias-input*. Next, I call the collection of states and processes that—in tandem with the bias-input—cause a discriminatory judgment *the bias-construct*; the bias-construct in E's case is his stereotype belief that elderly people are bad with computers (together with whatever inferential processes are necessary to derive the conclusion). I call the discriminatory judgment that bias-constructs and bias-inputs together cause—like E's belief that Jan

[19] See Holroyd and Sweetman 2016 for discussion and examples.

is bad with computers—*the bias-output*. Finally, I call actions that are performed on the basis of bias-outputs—like E's trying to help Jan with the Skype interview—*bias-actions*. Using this terminology, I interpret those who say that an individual harbors a bias as claiming—at minimum—that that individual harbors a bias-construct. This is why it's appropriate to regard an individual as having a bias even in cases where they don't draw particular conclusions or act on that bias.

I use the notion of a mental *construct* to pick out an open-ended collection of mental states and processes.[20] Social psychologists and philosophers investigating bias-constructs regard them as involving one of two core mental states: *stereotypes* or prejudices. The definition of 'stereotype', however, is not a straightforward matter. For our purposes, we can stick with a standard textbook definition: a stereotype is "a set of cognitive generalizations (e.g., beliefs, expectations) about the qualities and characteristics of the members of a group or social category."[21] On this view, stereotypes are beliefs about members of social groups that take the form of generalizations, e.g., elderly individuals are bad with computers.[22] Following standard representationalist views about belief, I regard them as propositionally structured mental states. *Prejudices*, on the other hand, are the *feelings* associated with a stereotype, e.g., affective responses toward a group of people. Since my main focus is on representational mental states, and these representational components are most clearly evident in the case of stereotypes, I set aside affective aspects of bias states (such as prejudices).[23]

Finally, it will be helpful to introduce the notion of a *contrast social group*: a group

[20] I take it this is roughly what Greenwald and Nosek (2008) and Machery (2017) have in mind with their uses of 'construct'. Importantly, a mental construct in this sense need not be constituted by mental representations, although in the case of explicit bias, it is.

[21] VandenBos 2015, 1031.

[22] This view of stereotypes as generalizations raises interesting questions about its relationship with a wide range of other philosophical theories including those about generics, the cluster-concept construal of stereotypes popularized by Putnam (1975), and theories of concept formation including Rosch (1978)'s notion of a *prototype*, to name a few. Though deserving of lengthy discussion, I largely set these issues aside for the remainder of the paper.

[23] I'm forced to oversimplify a vast literature on these topics. The distinction between stereotypes and prejudices was made prominent by Allport (1979)'s landmark book *The Nature of Prejudice.* Many have followed this general distinction in social psychology for notions such as *stereotype*, *attitude*, and *prejudice.* In short, the difference is usually between evaluative and affective mental states on the one hand and mental states that attribute properties to persons or groups on the other. I adopt the philosophical notion of *attitude* and regard biases as involving *either* likings or dislikings toward members of certain social groups *or* attributions to them of stereotypical properties, covering both *attitudes* and *stereotypes* in the social psychological sense. See Banaji and Greenwald 1994, Greenwald and Banaji 1995, Banaji and Hardin 1996, Madva and Brownstein 2016, and particularly Machery 2016, 105-110 for discussion and examples.

distinct from the relevant target social group, i.e., the group referenced in the stereotype, but that relates to it along some salient dimension. In the case above, E wouldn't have a bias against old people if it turned out that he believed that the young, the elderly, and everyone in between were bad with computers, and on the basis of this concluded that everyone he interacted with typically needed help with their computers. Instead, E needs to treat the relevant social group (the elderly) differently from how he treats the contrast social group (young people). This notion of *differential* treatment (in thought or behavior) is crucial to understanding the operation of social bias.[24]

Notice that in E's inference, he takes as a premise that Jan is elderly, or Ej in predicate logic, and he comes to the conclusion that Jan is bad with computers, Bj. What allows us to make a valid inference from the first to the second isn't merely the premise that elderly people are bad with computers, $\forall x(Ex \rightarrow Bx)$, but rather that premise together with the universal instantiation rule and *modus ponens* in predicate logic.[25] Since the bias-construct is *whatever* bridges the gap between the two, the bias-construct in this case is a combination of both the generalization and the inference rules.

Largely, extant theories of social bias regard *implicit* biases as involving bias-constructs that we are not aware or conscious of and *explicit* biases as involving those that we are. This is a claim that implicit and explicit biases differ with respect to their **conscious accessibility**.[26] However, there exists another dimension along which implicit biases and explicit biases might differ that has been largely neglected by extant theories: the bias's **representational status**. Bias-constructs that are unconscious might be, on the one hand, stored and represented like other mental states; or, more curiously, they might be "merely encoded" with no representational basis at all. In this case, the states and processes

[24] The importance of this notion of differential treatment can be seen in the structure of tests for bias: both direct and indirect measures are designed to reveal differential treatment, e.g., direct reports involve a difference in responses about questions regarding target and social groups, and the IAT involves a difference in performing sorting tasks about target and contrast social groups. The emphasis on this notion also allows for important structural similarities between my model of bias and models of discrimination in ethics and law (e.g., Lippert-Rasmussen (2014, 15-16)'s model of discrimination as "essentially comparative with respect to individuals, i.e., a matter of how an agent treats some people compared to others").

[25] Strictly speaking, it might be more appropriate to model this content as a generic rather than a universal quantification. Doing so might additionally prove useful in explaining much of the real-world "messiness" that biases tend to exhibit since generics are prone to similar variability, e.g., being context-sensitive, having statistical variation, or having modal import. For a discussion of the range of characteristics generics display, see Nickel 2016 and Leslie 2017; Leslie, Sarah-Jane 2015. In what follows, I'll avoid these complications.

[26] Recently theorists including Gawronski et al. (2006) and Hahn et al. (2014) have disputed that implicit biases are in fact unconscious. I don't take up the dispute in detail here because it is largely irrelevant for the main contention of my paper regarding the *representational nature* of bais-constructs.

underlying them are analogous with respect to their representational status to algorithmic biases—they are *truly implicit*.

To see this point, notice first that distinct bias-constructs can systematically relate bias-inputs and bias-outputs in similar ways. Consider a case similar to E's. In this new version, imagine that a different colleague, T, also considers Jan elderly. Like E, T assumes that Jan needs assistance joining a Skype interview, and she does not make this assumption about younger colleagues. However, unlike E, T appears to lack any conscious stereotype that elderly people are bad with computers. In fact, if you asked T, she would deny the claim and assert instead that elderly individuals are just as good with computers as anyone else is.

Given the difference between the two cases, it becomes difficult to explain what prompted the assumption that Jan needs help with a Skype interview. T seems to share the following beliefs with E:

(*i*) Jan is elderly
(*iii*) Jan is bad with computers

It seems that in order to explain T's behavior, we must posit the existence of a mental entity, which is non-obvious to T herself, and that plays the same role for her as (*ii*) does for E, i.e., mapping (*i*) onto (*iii*). But, of course, we can't on the basis of this alone assume that T believes (*i*) elderly people are bad with computers. As in the case of algorithmic bias, the transition from (*ii*) to (*iii*) could instead be the result of a truly implicit bias.

My claim that truly implicit cognitive biases are plausible is the claim that the principles governing the transitions from the belief that Jan is elderly to the belief that she's bad with computers are plausibly merely encoded in just the same way. Consider another example of a small child successfully running to catch an overshot ball that's been thrown to them. In this case, we needn't attribute to that child any understanding of the complex geometric principles that govern the rate and angle at which the child had to be moving to make the catch successful. These rules merely describe the mathematical function we think best explains the motion of the child in relation to the goal of catching the ball, but these motions can be wholly constituted by non-representational cognitive processes and motor activities of the child. Theoretically, the same could apply to T. That is, we needn't attribute to T explicit representation of the rules that describe the mental transition from one belief to the other any more than we need attribute any knowledge about complex geometric principles to the child or explicit stereotype rules to the code of a machine with an algorithmic bias.[27]

Let's now take stock of the similarities between the implicit cognitive case and the algorithmic case. The bias-input in the case of T's cognitive bias is a belief with the content *Ej*. The bias-input in the case of the machine learning program is a combination of the object, call it *j*, and some relevant feature attribute label, e.g., *E* for *elderly*, which results in the same content as the aforementioned belief: *Ej*.[28] The bias-output in the case of T's cognitive bias is a belief with the content *Bj*. The bias-output in the case of the machine learning program is the combination of the test instance, again *j* and a new classification attribute label, *B* for *bad-with-computers*. This again results in the same content as the relevant belief: *Bj*. Finally, the bias-construct in both cases is *whatever plays the role* of systematically relating variable inputs to variable outputs. In the cognitive case, this will be some combination of states and processes that mimic the stereotype that elderly people are bad with computers and the inferential processes that lead from that belief together with the bias-input to the bias-output. In the algorithmic case, the bias-construct comprises Euclidean distance calculations on the data in the feature space that result in the bias-output.

3 The Functional Account

The possible existence of truly implicit biases in both the cognitive and the machine learning domains calls for a theory of bias that allows for "internal" diversity—i.e., that prescinds from which states and processes underlie different types of bias—while maintaining "external" uniformity—i.e., that conveys what's common in the operation of these types of bias. Such criteria are best met by a functional approach to bias. Thus, I propose a functional model of social bias *tout court*, which covers both cases of algorithmic and cognitive social biases, and where each bias-construct has a bias-input and a bias-output.

[27] The example of a child catching a ball is based roughly on an example presented by Devitt (2006, 50). Moreover, the fact that some rules are encoded in this way is not new to philosophy. Carroll (1895)'s famous discussion of what Achilles said to the Tortoise demonstrates that some rules of inference *must* be merely encoded, otherwise we run the risk of an infinite regress. Three other examples include moral inference (Horgan and Timmons 2007), hypothesized 'Bare Inferential Transitions' (Quilty-Dunn and Mandelbaum 2017), and internalized generative grammar rules (Stabler 1983; Chomsky 1965).

[28] Here, I switch from the previous gradient feature attribute label of a particular age to the polar attribute 'E' for elderly to illustrate the similarities between the two cases. This shift is trivial. The important point is about structure: both the belief and the algorithmic input combine some object and feature label, resulting in propositional structure.

In all cases, the input and output are propositional states (represented in the algorithmic case as combinations of objects, feature-values, and class labels) and the bias-constructs are some combination of states and processes that systematically relate the bias-inputs to their characteristic bias-outputs. Crucially, this model affords the advantage over other models of bias of being able to account for truly implicit biases.

Putting all of these functional components together, we get a formal definition of what it means for some particular object P (person or program) to *have a bias*:

Definition of Having a Bias:

For any social group label G and attribute label T, some object P *has the bias* $\boxed{G{\cdot}T}$ iff
there exists a contrast group label H such that for any object x
(1) the input Gx reliably causes P to output Tx
(i.e., classify x as T), and
(2) the input Hx does not reliably cause P to output Tx
(i.e., classify x as T)

Applying this definition to the cases of E, T, and the machine learning program outlined above, we can see that all three instantiate the bias $\boxed{E{\cdot}B}$ since the inputs with content *Ej* reliably cause the outputs with content *Bj*, and inputs with contents *Yj* do not reliably result in *Bj*.

This definition of *having a bias* is useful because it incorporates the characteristic feature of bias—namely, the operation of systematically relating bias-inputs to bias-outputs—while remaining agnostic about certain phenomenal and representational features of the bias-construct itself. However, it's important to note that this definition is too simple to account for many real-world instances of bias. Cases of intersectionality and familiarity can affect the predication of an attribute to an individual despite that individual's perceived belonging to a social group. For this reason, comprehensive accounts of having a bias will need to incorporate ways of weighing certain attributes against others.[29] This model is an attempt to describe the most basic case of an individual having a bias; that is, cases where an individual has a bias and there are no "defeaters" present.

According to this functional account of bias, machine biases and cognitive biases are of the same basic (functional) kind. Having a model that extends to these two species of

[29] Ultimately, these weights and the complex relations between social groups they're meant to encode are within the purview of social theory and social psychology to discover and outline. I'm happy to have my functional theory defer to those fields of study with respect to how these relationships ought be characterized.

bias helps in identifying important practical lessons that comparisons of them can provide, irrespective of their representational statuses. In the next section, I provide the example that biases of both kinds can operate using proxy attributes.

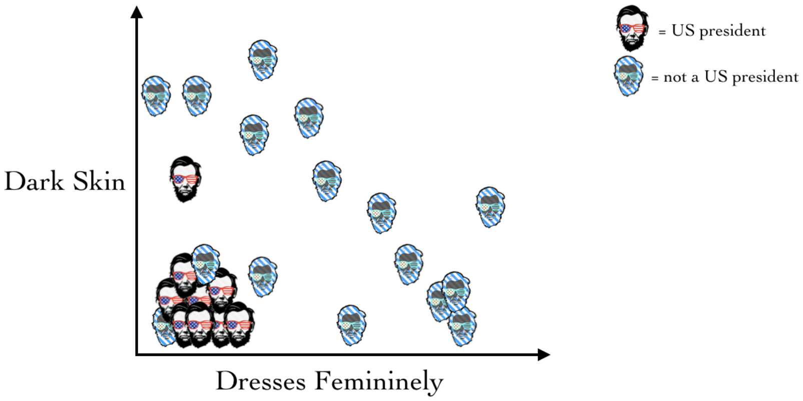
4 Proxy Attributes

Machine learning programmers have long struggled with eliminating biases that are based on so-called 'proxy attributes'.[30] Often, engineers attempt to protect disadvantaged social groups by preventing classification algorithms from performing tasks on the basis of socially-sensitive features in cases where using these features would be discriminatory. For example, it would be both legally and morally problematic to allow a program to categorize candidates as either eligible or ineligible for a mortgage loan based on those candidates' races. As a result, programmers attempt to prevent any explicit reliance on race by including filters that block the program from labeling individuals on the basis of race. However, often these filters fail to prevent the program from adopting so-called 'proxy attributes' that correlate with the socially-sensitive attributes. For example, rather than labeling some individual as 'African American', the program might label them based on their zip codes. Since neighborhood demographics are often racially homogeneous, a person's zip code will often correlate with their race. Thus, an algorithm utilizing the former might operate very similar to it utilizing the latter.[31]

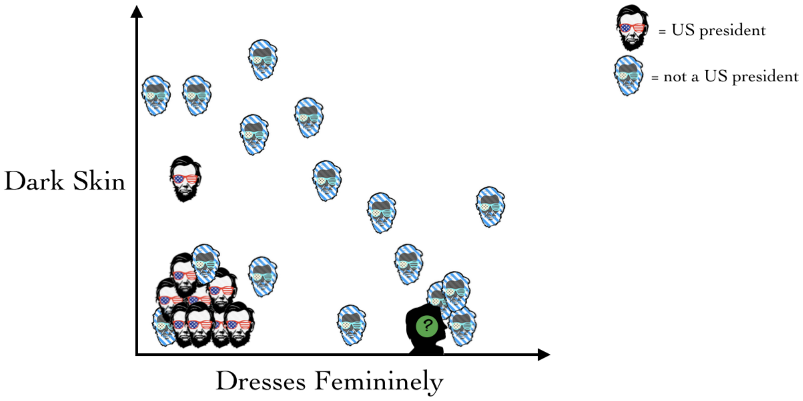
Consider yet another classification task. This time, imagine the classification task of categorizing individuals as presidential or non-presidential and that the chosen relevant features the program relies on are skin tone and the degree to which a person dresses in a stereotypically feminine manner—imperfect proxies for race and gender, respectively. Just as in the training phases above, this program utilizes known instances, in this case past U.S. Presidents. Based on historical discriminatory trends, we can imagine the feature space looks very roughly like the following:

[30] See, for example, Adler et al. (2016)'s discussion of how to audit so-called 'black box' algorithms that appear to rely on proxy attributes in lieu of target attributes.

[31] Often, programmers are forced to rely on proxies. See, for example, Eubanks 2018, 143-145, 168. Moreover, the notion of a proxy discrimination is familiar in discussions of discrimination in ethics and law. See, for example, Alexander 1992, 167-173.



With these training data, we can predict what is likely to happen in the test phase with an individual who, for the most part, dresses in a stereotypically feminine manner:



Such an individual will, on the basis of their feminine presentation, be categorized as not a U.S. President. Crucially, the program makes no explicit reference to race or gender, but the results of its operation correlate with the results from a program that did rely on such features.

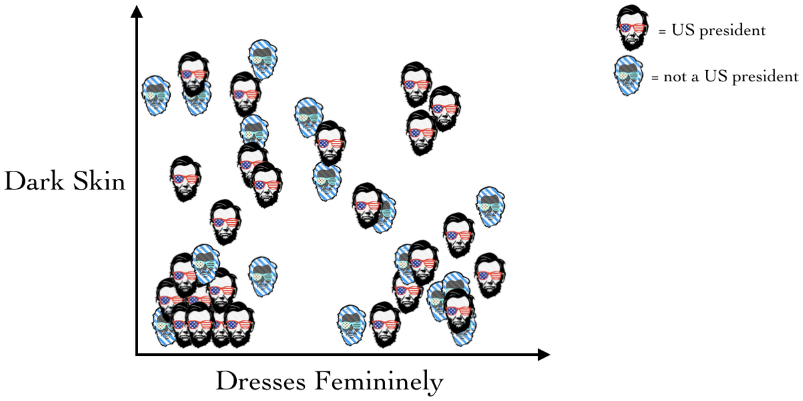
There's no reason in principle to rule out that individuals might similarly rely on proxy attributes to make stereotypical judgements about other individuals.[32] Someone voting in a presidential election, for example, might rely on similar proxies and known historical trends for categorizing an individual as a viable candidate. If so, their inferences might

[32] The felt plausibility of this point likely depends on one's theory of social kind concepts.

pattern roughly as if they were relying on each candidate's membership in the relevant social categories. In these cases, it seems that that individual still has a bias against members of the social groups, even in cases where they don't recognize that the relevant features are collections of proxy attributes.

Incorporating the notion of proxy attributes into our functional model of *having a bias* will require a few minor amendments. In these cases, we might just say that 'For any social group G (or relevant proxy attribute J)...' and define the proxy attributes as those that significantly correlate in the population with a member's belonging to that social group. This would entail that an individual whose belief with the content Jx reliably causes a belief with the content Bx has a bias against the elderly, despite their never tokening a belief with the content Ex. Similar accommodations can be made for the notion of a contrast social group, where we would also include the idea of a contrast proxy attribute that correlates with an individual's membership in the contrast group.

Crucially, the insight that cognitive biases might, like machine biases, operate using proxy attributes has important implications for mitigation techniques in both domains. Programmers have long struggled with proxy attributes since programs that rely on them tend to resist any overt filtering techniques. Eliminating any explicit references to race in the program's code will be ineffective, as the program can simply substitute a proxy attribute in place of race, resulting in similar discriminatory effects. One might think that human implicit biases similarly resist revision. That is, if a mitigation technique operates on overt, socially-sensitive attributes, and the relevant cognitive biases instead rely on proxy attributes, then that mitigation technique will fail. If so, we might borrow mitigation techniques from machine learning. One programming strategy is to curate the training data in such a way that the problematic features bear no straightforward relationship to the relevant categories, making reliance on them for categorization ineffective. If we wanted to likewise frustrate the reliance on the proxies in the example above, one place to start is to introduce more instances of counter-stereotypical exemplars, as follows:



In this case, relying on whether a candidate dresses in a stereotypically feminine manner or not will not be a reliable guide in categorizing them as presidential. This example appears to naively advocate that we combat biases by changing the overarching social patterns that are likely themselves the result of the biases we're attempting to ameliorate, i.e., it appears to advocate that we *simply* elect more women and people of color. However, in its more modest form, it serves to concretely demonstrate and bolster the critical insight from equality advocates that *representation matters*.

5 Are Machines Biased?

In this paper, I've suggested we regard computers and humans as harboring biases that are of the same functional kind. Some might be averse to this claim for the reason that it implies humans and computers are capable of similar psychological capacities. For the same reason we resist saying that a computer has any beliefs, desires, or emotions, we might also want to resist attributing to them biases proper.

Although I think that the functional characterization I've provided helpfully subsumes both sorts of biases, my account leaves open that there might still be philosophical reasons for distinguishing between them. For example, theories of content in philosophy of language might individuate the representations involved in the input-output profile of human biases from the contents involved in algorithmic biases. Similarly, theories of moral responsibility and blame in ethics and value theory might provide philosophically important reasons for regarding the biases that operate in human agents as entirely distinct from the biases that operate within machine learning programs. My point is not to deny that these other theories might eventually provide reasons for distinguishing between the two, but rather to resist at the onset foreclosing the possibility that these philosophical investigations might go the other way, or that there are other explanatory projects for which the identification is apt.

Finally, although I see on the horizon tempting philosophical reasons to avoid saying computers are biased in the same way humans are, I also feel there are pragmatic considerations that motivate this comparison, as it clearly highlights valuable avenues of philosophical inquiry explored in one area that have been unexplored in the other. My example of proxy attributes is one such avenue. Along similar lines, it seems to me there exist interesting questions regarding what constitutes a bias in the first place and whether it's even possible to entirely eliminate biases within a person or program.

Consider, for example, recent work by Kleinberg et al. (2016) and Chouldechova (2016). In this work, researchers identify three intuitive conditions some risk-assessment program must achieve in order to be fair and unbiased. These criteria include first, that the algorithm is *well-calibrated*, i.e., if it identifies a set of people as having a probability z of constituting positive instances, then approximately a z fraction of this set should indeed be positive instances; second, that it *balance the positive class*, i.e., the average score received by people constituting positive instances should be the same in each group; and third, that it *balance the negative class*, i.e., the average score received by people constituting the negative instances should be the same in each group.[33] Strikingly, Kleinberg et al. (2016) demonstrate that in cases where base rates differ and our programs are not perfect predictors—which subsumes most cases—these three conditions necessarily trade off from one another. This means most (if not all) programs used in real-world scenarios will fail to satisfy all three fairness conditions. There is no such thing as an unbiased program in this sense. More strikingly, researchers regard this so-called 'impossibility result' to generalize to all predictors, whether it be an algorithm or a human decision-maker.[34]

In summary, this account is not intended to argue that algorithmic and cognitive biases are similar in all respects; rather, it's intended as a starting point for the comparison between the two. This starting point might eventually enable us to see important differences between the two cases, or it might enable us to see the similarities; in both cases, however, it will lend to a better understanding of bias.

[33] Kleinberg et al. 2016, 2.

[34] Kleinberg et al. 2016, 6 and Miconi 2017, 4.

6 Conclusion

Inquiries regarding the nature of bias and the utility of comparing its existence in both machine and cognitive domains are still in their infancy. However, preliminary results showing the usefulness of these comparisons have been positive. This paper aimed to highlight just a few of these positive results, while gesturing at the many avenues such comparisons open for further, fruitful philosophical engagement. Given the potential benefits of comparisons, it seems obvious that we need a common model of bias in both domains that will serve to inform and structure the comparisons themselves. The account presented in this paper provides one such model that could serve this role.

7 Bibliography

Adler, P., Falk, C., Friedler, S. A., Rybeck, G., Scheidegger, C., Smith, B., and Venkatasubramanian, S. (2016). Auditing black-box models for indirect influence. In *Data Mining (ICDM), 2016 IEEE 16th International Conference on*, pages 1–10. IEEE.

Alexander, L. (1992). What Makes Wrongful Discrimination Wrong? Biases, Preferences, Stereotypes, and Proxies. *University of Pennsylvania Law Review*, 141(1):149.

Allport, G. (1979). *The nature of prejudice.* Basic Book.

Angwin, J., Larson, J., Mattu, S., and Kirchner, L. (2016). Machine bias: There's software used across the country to predict future criminals. And it's biased against blacks. *ProPublica.*

Banaji, M. R. and Greenwald, A. G. (1994). Implicit stereotyping and prejudice. *The psychology of prejudice: The Ontario symposium*, 7:55–76.

Banaji, M. R. and Hardin, C. D. (1996). Automatic Stereotyping. *Psychological Science*, 7(3):136–141.

Barnes, E. (2016). *The Minority Body.* Oxford University Press.

Beeghly, E. (2015). What is a Stereotype? What is Stereotyping? *Hypatia*, 30(4):675–691.

Caliskan, A., Bryson, J. J., and Narayanan, A. (2017). Semantics derived automatically from language corpora contain human-like biases. *Science*, 356(6334):183–186.

Carroll, L. (1895). What the tortoise said to Achilles. *Mind*, 4(14):278–280.

Chomsky, N. (1965). *Aspects of the Theory of Syntax*. MIT Press.

Chouldechova, A. (2016). Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *arXiv preprint arXiv:1610.07524*.

Datta, A., Tschantz, M. C., and Datta, A. (2015). Automated Experiments on Ad Privacy Settings. *Proceedings on Privacy Enhancing Technologies*, 2015(1).

Daum III, H. (2015). *A Course in Machine Learning*. TODO.

De Houwer, J. (2014). A Propositional Model of Implicit Evaluation: Implicit evaluation. *Social and Personality Psychology Compass*, 8(7):342–353.

Dennett, D. C. (1981). A Cure for the Common Code. In *Brainstorms: philosophical essays on mind and psychology*, pages 90–108. MIT Press, Cambridge, Mass.

Devitt, M. (2006). *Ignorance of Language*. Oxford University Press.

Eberhardt, J. L., Goff, P. A., Purdie, V. J., and Davies, P. G. (2004). Seeing Black: Race, Crime, and Visual Processing. *Journal of Personality and Social Psychology*, 87(6):876–893.

Eubanks, V. (2018). *Automating Inequality: How High-Tech Tools Profile, Police and Punish the Poor*. St. Martin's Press.

Fazio, R. H. (1990). Multiple Processes by which Attitudes Guide Behavior: The Mode Model as an Integrative Framework. In *Advances in Experimental Social Psychology*, volume 23, pages 75–109. Elsevier.

Gawronski, B. and Bodenhausen, G. V. (2014). Implicit and Explicit Evaluation: A Brief Review of the Associative-Propositional Evaluation Model: APE Model. *Social and Personality Psychology Compass*, 8(8):448–462.

Gawronski, B., Hofmann, W., and Wilbur, C. J. (2006). Are implicit attitudes unconscious? *Consciousness and Cognition*, 15(3):485–499.

Gendler, T. S. (2008). Alief and belief. *The Journal of Philosophy*, 105(10):634–663.

Greenwald, A. G. and Banaji, M. R. (1995). Implicit social cognition: attitudes, self-esteem, and stereotypes. *Psychological Review*, 102(1):4–27.

Greenwald, A. G. and Nosek, B. A. (2008). Attitudinal dissociation: What does it mean. *Attitudes: Insights from the new implicit measures*, pages 65–82.

Hahn, A., Judd, C. M., Hirsh, H. K., and Blair, I. V. (2014). Awareness of implicit attitudes. *Journal of Experimental Psychology: General*, 143(3):1369–1392.

Holroyd, J. (2016). VIIIWhat Do We Want from a Model of Implicit Cognition? *Proceedings of the Aristotelian Society*, 116(2):153–179.

Holroyd, J. and Sweetman, J. (2016). The Heterogeneity of Implicit Bias. In Brownstein, M. and Saul, J., editors, *Implicit Bias and Philosophy Volume 1: Metaphysics and Epistemology*, pages 80–103. Oxford University Press.

Horgan, T. and Timmons, M. (2007). Morphological Rationalism and the Psychology of Moral Judgement. *Ethical Theory and Moral Practice*, 10(3):279–295.

Klare, B. F., Burge, M. J., Klontz, J. C., Vorder Bruegge, R. W., and Jain, A. K. (2012). Face Recognition Performance: Role of Demographic Information. *IEEE Transactions on Information Forensics and Security*, 7(6):1789–1801.

Kleinberg, J., Mullainathan, S., and Raghavan, M. (2016). Inherent trade-offs in the fair determination of risk scores. *arXiv preprint arXiv:1609.05807*.

Leslie, S.-J. (2017). The original sin of cognition: Fear, prejudice, and generalization. *The Journal of Philosophy*, 114(8):393–421.

Leslie, Sarah-Jane (2015). Generics Oversimplified: Generics Oversimplified. *Nous*, 49(1):28–54.

Levy, N. (2015). Neither Fish nor Fowl: Implicit Attitudes as Patchy Endorsements: Neither Fish nor Fowl: Implicit Attitudes as Patchy Endorsements. *Nous*, 49(4):800–823.

Lippert-Rasmussen, K. (2014). *Born free and equal? a philosophical inquiry into the nature of discrimination*. Oxford University Press, Oxford, New York.

Lowry, S. and Macpherson, G. (1988). A blot on the profession. *British medical journal (Clinical research ed.)*, 296(6623):657.

Machery, E. (2016). De-Freuding implicit attitudes. In *Implicit Bias & Philosophy: Metaphysics and Epistemology*, volume 1, pages 104–129. Oxford University Press.

Machery, E. (2017). Do Indirect Measures of Biases Measure Traits or Situations? *Psychological Inquiry*, 28(4):288–291.

Madva, A. and Brownstein, M. (2016). Stereotypes, Prejudice, and the Taxonomy of the Implicit Social Mind. *Nous*.

Mandelbaum, E. (2015). Attitude, Inference, Association: On the Propositional Structure of Implicit Bias. *Nous*, DOI:10.1111/nous.12089, p. 1–30.

Massey, D. S. and Denton, N. A. (1993). *American apartheid: segregation and the making of the underclass*. Harvard University Press, Cambridge, Mass.

Miconi, T. (2017). The impossibility of “fairness”: a generalized impossibility result for decisions. *arXiv:1707.01195 [cs, stat]*. arXiv: 1707.01195.

Narayanan, A. (2016). Language necessarily contains human biases, and so will machines trained on language corpora.

Nickel, B. (2016). *Between Logic and the World*. Oxford University Press.

O’Neil, C. (2016). *Weapons of Math Destruction: How big data increases inequality and threatens democracy*. Crown Publishing Group.

Price, R. (2016). Microsoft is deleting its AI chatbot’s incredibly racist tweets. *Business Insider*.

Putnam, H. (1975). The Meaning of ‘Meaning’. In *Mind, Language and Reality*, pages 215–271. Cambridge University Press.

Quilty-Dunn, J. and Mandelbaum, E. (2017). Inferential Transitions. *Australasian Journal of Philosophy*, pages 1–16.

Rosch, E. (1978). Principles of categorization. In Rosch, E. and Lloyd, B. L., editors, *Cognition and Categorization*. Lawrence Erlbaum Associates, Hillsdale, NJ.

Ryle, G. (1949). *The Concept of Mind*. Barnes and Noble.

Schwitzgebel, E. (2002). A phenomenal, dispositional account of belief. *Nous*, 36(2):249–275.

Stabler, E. (1983). How are grammars represented. *Behavioral and Brain Sciences*, 6(3):391–421.

Stephens-Davidowitz, S. (2014). Opinion | Google, Tell Me. Is My Son a Genius? *The New York Times*.

VandenBos, G. R., editor (2015). *APA dictionary of psychology (2nd ed.)*. American Psychological Association, Washington.

Wu, X. and Zhang, Z. (2016). Automated Inference on Criminality using Face Images. *arXiv preprint arXiv:1611.04135*.

Yumusak, E. (2017). “Implicit Bias and the Unconscious”. In *The 2017 Minds Online Conference*. The Brains Blog.