

Engendering Algorithmic Oppressions

By Susan V.H. Castro

In 2019, CNN and other news services **reported** that the New York Department of Financial Services was investigating Apple Card for gender bias because its issuer, Goldman Sachs, was giving women much lower credit limits than their male spouses, including Steve Wozniak, even if the couple shared assets and accounts and the women had higher credit scores. Since commercial algorithms are considered proprietary, creditworthiness criteria are opaque to oversight agencies and the public. Was her limit 20 times lower because she makes less money? If their accounts are shared, the income source is irrelevant to repayment ability. The algorithm might be putatively “gender-blind”, but income has long been used as a symbol and surrogate for “who wears the pants”.

In 2020, the world changed. Children were sent home from schools to be parented and educated full-time, primarily by mothers, many single, many who lost their jobs, many with newly online or essential jobs that became more difficult and dangerous for the same or less pay. It’s unclear what algorithms that were trained pre-pandemic will do with this discontinuity, and equally unclear what lending criteria *should* be in this **transitional** period.

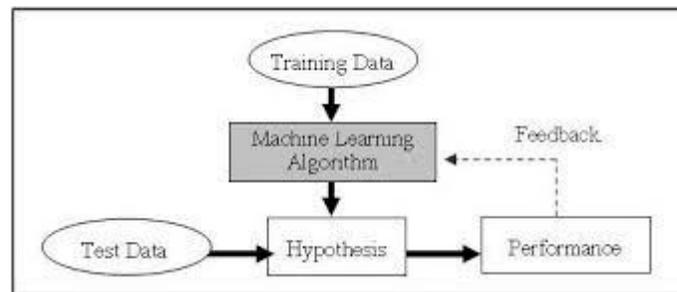
The pandemic hit communities of color harder in the US, killing more Black, Indigenous, and People of Color (BIPOC) than Whites, and widening gaps everywhere. Breathing while Black became a flash point for defunding the police. Though headlines have been largely silent about facial recognition inaccuracy for women (currently the fastest growing prison population), facial recognition software for suspect identification is finally being **taken off the market** because its long-publicized inaccuracy for BIPOC made this use grossly unjust. Facial recognition for biometric security hasn’t lost a step. Women and BIPOC will still be locked out of their own secure spaces more frequently than their peers, thus be subject to greater scrutiny, accidentally-by-design.

The questions these anecdotes raise include how we should conceive the issue, how such problems arise, and what it will take to prevent or correct them. In the following I briefly address these questions, beginning with the idea of **algorithmic oppression**.

Understanding algorithmic oppression

Since the advent of assembly line production, the labor-saving promise of **algorithms**, **i.e. precisely specified instructions for performing a task to be repeated at scale**, has increasingly structured human lives and gradually extended from physical to mental labor. Today, autocomplete finishes our thoughts. Autotag classifies us and autotranslate speaks for us. Algorithms are personalizing our newsfeeds. Apps are awarding financial and social credit based on human factors like **how confidently you type (WeChat)**. Smart devices surveil our desk usage to optimize workflow (**OccupEye**). First round job interviews are conducted and

evaluated by AI (**HireVue**). Algorithms are telling us who's a criminal and who's a terrorist, and how to sentence them (**COMPAS**). They're predicting which (poor) children are at **risk of abuse** (**AFST**) and promising to diagnose disease (**WSI**). These facts are not value-neutral.



Oppression is an anti-ideal describable by three criteria:

1. It involves prolonged or endemic subjection to **unjust** treatment or control;
2. it's a **scale phenomenon** that often operates on classes of people; and
3. it's **inescapable**.

Algorithms by nature tend to meet these criteria. Any precise set of instructions deployed at scale will have a scale effect, sometimes an unjust effect. Consequently, even small flaws in design or implementation can amplify and compound to mass destruction, as **Cathy O'Neill** argued. This is especially likely when they're opaque to oversight, as commercial algorithms are, and they're inescapable, e.g. via contracts of adhesion (e.g. take-it-or-leave-it terms of agreement), or simply via the universal use of an algorithm (e.g. all search engines are based on the same algorithm). As we outsource intellectual labor to algorithms that **homogenize and hegemonize**, the tendency is to punish diversity and compound oppression by math-washing the ills of extant systems and entrenching them under a veil of presumed objectivity, immutability, and quantified correctness. Our current pervasive and expanding use of algorithms thus presents a high risk of oppression. This makes algorithmic oppression a highly apt way to conceive the way in which facts about algorithm design and use are laden with values.

Given that most algorithm use is expressly promoted as labor-saving, we should **question whose labor is saved and whether it's worth saving**. The labor invested in developing and maintaining software systems is considerable. The cost to clients is substantive. For this kind of economic defense to succeed, algorithm use must shift us from undesirable work to desirable work (fun programming problems), reduce the total labor to maintain our well-being (increasing leisure time), and distribute benefits fairly. Thus far, satisfying work, material wealth, and leisure time have accrued to very few. Even so, **if algorithms can correct human biases**, they could yield a lighter workload for humans and far better decisions about matters that are crucial to human well-being. In other words, algorithms might offer relief from various forms of oppression. Considering how difficult the alternative has proven – making *people* better – the choice of which tech to abandon, which to pursue, and how to pursue it is a central challenge of our era.

Engendering issues

Most of the work on algorithmic oppression has focused on understanding how it's engendered and identifying alternatives. Some of the root causes of algorithmic oppression involve the flawed ideology that underlies standard tech practice.

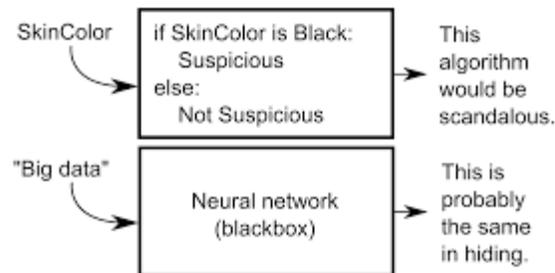
For example, the core tenet of **technochauvinist ideology** is that tech is always the answer. Technochauvinists would argue that facial recognition algorithms should be repaired rather than abandoned. Algorithms do *exactly* what they're designed to do, so if you want algorithms to do better, specify precisely how. **Kearns and Roth**, for example, argue that the general solution is to **program ethical values** like fairness into algorithm parameters. According to their statistical argument, the most accurate model will be driven by the majority rather than the margin. Consequently the most accurate model will be the least accurate, thus least fair, for marginalized people, and this statistical tradeoff between accuracy and fairness is ineliminable. Their position is technochauvinist in that they assume that the best **precise specification of fairness** they can identify is correct, or at least ethically adequate, even if it effectively results in a moral blind alley. Giving up on an algorithm because ethics can't be programmed is not a live option for them, but it should be for us. We should question whether a core ethical value like fairness can be reduced to statistics, or precisely defined at all. What does ethical agency actually require?

Edge case reasoning is also a significant contributor to algorithmic oppression. Since women are half the human species and half of BIPOC are male, the putative margin is often actually the majority – though not the center. As **Wachter-Boettcher** argues, an edge case is effectively someone who's not important enough to care about. When we write off a person or class as an edge case, we limit our customer base and fail to maximize sales – a capital crime – but we also commit a moral failure of respect by denying they are worthy of due care. In contrast, a **stress case** is one that *matters*, one that shows designers where their work breaks down. The moral and professional standard should be to create work that doesn't break down for anyone, Wachter-Boettcher argues, especially for those who need it most or who would suffer most. One might even argue that that algorithm design entails a **fiduciary duty** – a duty of trust – to all stakeholders.

Taking the stress case alternative one step further, **universal design ideology** shifts the conceptual landscape by eliminating background assumptions that generate disabling design. Consider for example the ubiquitous ableist assumption that designing for more 'kinds' of people is more complicated, difficult, or expensive. What universal design demands is rather that we overcome functional fixedness to enable discovery of simpler, cheaper, more robust design. As work concerning **algorithmic ableism** progresses, we should expect benefits for everyone. At the *very* least, being labeled transgender or disabled or Latinx by an algorithm shouldn't cost you custody or health coverage or travel rights.

The **race to release** is another feature of tech practice that is highly conducive to oppressive outcomes. For example, it's standard practice to alpha test on a few early adopters, then beta test

in the field to work the rest of the bugs out. When the product is a video game and the worst-case scenario is a power cycle, beta testing in the field makes sense. You get a faster rollout and more thorough testing than you can easily do in house. Even better, **debugging labor is outsourced to users**, whose complaints can be patched with updates. But we can't generalize from sensible practice for video games to child abuse prediction or medical diagnosis. The worst-case scenarios are not comparable. Perhaps something akin to our standards for **experimenting on human subjects** should apply.



Moreover, vanishingly few users have the data access or the technical skill to assess algorithm performance and even when bugs are evident to users, mere updates make no real progress against oppression. An update might remove the porn from [top Google search results on “black girls”](#), but not from Google searches for “Chinese girls” or DuckDuckGo searches for “black girls”. Today *she is autotagged a “newsreader”* though *he* an “expert”; *her* expertise is autotranslated into *his*; users report and we patch. Tomorrow the algorithm promotes cis-male superiority in a novel way. If the root cause here is the oppositionality of the analogical inference model for [word2vec](#), updates would be infinite.

Poor standard practice is often **rationalized** by appeal to flawed ideology. The ideology of **data freedom**, for example, mandates that “we” collect all possible data, share it with everyone, use it in every possible way to maximize its value, and monetize that value to support the cycle. Anonymizing is a mere formality because we’re all trustworthy here and privacy is no longer a social norm. Given a right to user data, tech companies are entitled to make data freedom inescapable. This largely covert rationalized ideology has led social media companies, ISPs, and smart device providers to routinely mislead users about what data they collect, who has access, and how it’s used. Data freedom ideology is also hypocritical: Company data and algorithm design are proprietary. Perhaps it’s time to rethink property rights in data.

To give one last example of how the flawed ideology that underlies standard practice engenders algorithmic oppression, the general shift from user-funded subscription to **ad-supported funding** has some moral merit. One way that email has become more widely available is by funding it through advertising. Unfortunately, the implementation of ad-supported technology has proven problematic. First, ad-support is driven by user engagement and the most effective way to maximize user engagement, thus profits, is to design addictive algorithms. [Addiction to your FitBit](#) might not be so terrible, but the same methods of dependence creation can automate debt slavery under the guise of financial empowerment. Exploitation is not an acceptable outcome. Second, ad-support algorithms rely on profiling and clustering methods

to **recommend** and personalize, which can have unintended consequence like siloing and **radicalization**. Perhaps it's not oppressive to silo entertainment recommendations based on usage rather than actual user preferences, but career siloing by HR algorithm design would be. Funding can engender algorithmic oppression, too.

Whose job is it?

As tech historians like **Abbate**, **Ensmenger**, and **Hicks** document, the systematic exclusion of women from the programming industry these women founded was no accident.

Those **exclusionary practices have been consequential**. Algorithms are inequitably shaping our identities and determining our access to resources like health care and housing. Reclaiming the tech industry as a career domain for people of all demographics is a necessary part of the solution, but we specifically need people who can do **quality control for social justice**. Programmers and computer architects must understand data structures, certainly, but tech developers and businesspeople at all levels of industry also need to have the spectrum of experience and evaluative reasoning skills required to *foresee* adverse effects on vulnerable populations as well as the motivation and authority to *prevent* and *correct* them, even when it impacts the bottom line. This requires a root shift in tech education and business culture.

User development is also required. After all, the practice of sexting is a root cause of those unfortunate autocorrects that might get you fired. (Autocorrect doesn't codeswitch well between context-sensitive norms for private, professional, and civil conduct.) Algorithms that mine our data and learn from our *actual* behavior will by default replicate and compound our biases and flaws rather than exemplify our unrealized ideals. We can't debias algorithms without debiasing ourselves. Movements like #MeToo and Black Lives Matter *do* change social norms and spur personal development. It remains to be seen what their reach will be.