AI Auditing and the Access Question: Exploring Black-box Auditing and its Connection to Hypothesis Testing

Sarah H. Cen* and Rohan Alur

Electrical Engineering and Computer Science, Massachusetts Institute of Technology

1 Introduction

Auditing is the process of evaluating the properties of a system, often to determine whether it satisfies a predetermined set of criteria. With the proliferation of Artificial Intelligence (AI) technologies, auditing will serve as a vital tool for AI oversight and accountability. Without the ability to systematically and consistently test—or *audit*—for compliance, AI regulations are impossible to enforce. Beyond compliance testing, auditing also plays several important roles. Perhaps most fundamentally, it allows for the independent verification of developers' claims that would otherwise go untested. It can also be used to certify whether an AI technology meets industry standards (e.g., privacy standards) that matter to downstream users (e.g., customers) even when they are not legally required. In this way, auditing not only plays an important role in AI accountability, but also takes an important step toward developing trustworthy AI.

Consider, by analogy, the U.S. car industry, in which auditing has three important functions. In the U.S., vehicles must adhere to a variety of federal standards and regulations related to safety and emissions, which are enforced through audits conducted by the National Highway Traffic Safety Administration (NHTSA). Beyond compliance testing, car manufacturers are required to disclose information about their vehicles, such as their fuel economy (i.e., mileage per gallon), which are both internally verified and subject to external audits by the Environmental Protection Agency (EPA). To gain an edge over competitors, car manufacturers also make claims about their vehicles; external and third-party audits validate these claims, legitimizing them and establishing trust between consumers and manufacturers.

There is a growing consensus that the AI industry would benefit from similar auditing mechanisms (Yeung 2018, Bandy 2021, Raji et al. 2022, Raji 2023). For instance, in the European Union (EU), the AI Act mandates a mix of internal and third-party audits in the form of "conformity assessments" that are conducted before the release of an AI system or after a substantial modification (Thelisson and Verma 2024); the General Data Protection Regulation (GDPR) calls for internal audits in the form of impact assessments conducted before data processing (Bieker et al. 2016); and the Digital Services Act (DSA) requires annual internal and external audits of the risks associated with each digital service (Wilman 2022). Moreover, to ensure compliance, regulatory bodies (e.g., under GDPR, data protection authorities in each EU member state) are granted broad authority to conduct investigations that test for compliance. There are even provisions (e.g., in the DSA) that grant researchers special access to data and systems so that they can audit for considerations that the regulatory bodies miss (Wilman 2022).

*Correspondence to Sarah H. Cen at shcen@mit.edu.

Creating a healthy AI auditing ecosystem includes various considerations, such as who conducts the audits, who audits the auditors, what standards auditors test for, whether audits are prospective or retrospective, how often audits are conducted, and more. For many of these considerations, we may be able to look to other industries for inspiration. However, one question is particularly salient:

What is the minimal access needed to effectively and efficiently audit an AI system?

All auditing procedures require some form of access to the AI system, but unpacking what is minimally required matters for two reasons: (1) intellectual property protections and (2) resource constraints. First, the protection of proprietary technologies and data (in particular, of trade secrets) is a key concern for companies. As a result, the amount of access granted to auditors is limited and carefully controlled. For example, even regulatory bodies that audit for GDPR compliance must adhere to strict principles of "necessity" and "proportionality." Second, auditors operate with limited resources (e.g., in labor, budget, and technical expertise) and are therefore interested in the amount of access that allows them to most effectively and efficiently conduct audits. Answering this question is therefore relevant to both AI developers and auditors.

Driven by this question, we seek to understand what can and cannot be achieved using **black-box access** to an AI algorithm f, which is defined as being able to query f on the auditor's choice of inputs $\{x_i : i = 1, \ldots, N\}$ and observe the outputs $\{f(x_i) : i = 1, \ldots, N\}$ but nothing further. In other words, black-box access allows an auditor to observe an algorithm's behavior under any condition (or input) of the auditor's choice, without knowledge of how that behavior is generated.

Compared to human processes, AI lends itself well to black-box auditing. Consider, for example, auditing a firm's (non-AI) hiring practices for discrimination. Here, a black-box audit would require gathering everyone who plays a role in hiring, handing them a stack of applications, asking them to evaluate each according to normal procedures, and observing their decisions. This process is not scalable, as it would require auditors and the firm to invest significant time and resources. Perhaps more worryingly, people can easily manipulate the outcome of the audit by behaving differently when audited. On the other hand, it is efficient to repeatedly query AI in a black-box way, and the results of black-box audits are guaranteed to be faithful to how the AI would behave in practice.

Benefits and limitations of black-box auditing for AI. In this work, we explore the capabilities and limitations of black-box auditing. We begin in Section 2 with a review of auditing in both non-AI and AI contexts, then discuss related work on AI auditing techniques. In Section 3, we unpack the merits and drawbacks of black-box auditing. In particular, black-box audits offer several key benefits: (1) they do not require direct access to proprietary algorithms or data, i.e., do not "white box" the AI system; (2) they are agnostic to the underlying AI mechanisms, meaning that the audit does not need to be updated even if the underlying algorithm, training data, or training pipeline changes; (3) they are less resource-intensive than alternate auditing options, allowing continual, comprehensive, and scalable auditing; and (4) they can be run prospectively[1] and reflect exactly how the algorithm would behave in practice.

At the same time, black-box audits have their blind spots: they cannot speak to the intentions of AI developers, determine whether the developers adopt best practices, or provide insights that may be necessary for accountability mechanisms (e.g., legal recourse). To address these blind spots, it is often necessary to complement black-box access with additional information from other sources (e.g., access to related open-source models) as well as white- or gray-box access (e.g., API access). Full access to the entire learning pipeline is often unnecessary, as auditors cannot develop scalable auditing procedures that are individualized to every training pipeline.

[1] before an AI system is deployed

Hypothesis testing as a framework for black-box auditing. In this work, we discuss how black-box auditing can be operationalized using hypothesis testing. That is, we propose that hypothesis testing can be used to "translate" audit requirements (e.g., AI regulations) into black-box tests. We introduce the well-studied problem of hypothesis testing in Section 4, drawing connections between hypothesis testing and legal principles. In particular, we discuss how the null hypothesis can be viewed as legal presumption, placing the burden of proof on the party that wishes to prove that the alternate hypothesis is true. We examine the four main components of hypothesis testing, highlighting parallels between the design criteria of hypothesis testing and of auditors. We conclude in Section 5 with limitations, challenges, and future work.

Remark. Both authors are, by training, computer scientists. Our hope is to provide a careful analysis of how auditing could be implemented in practice given the rapid pace of AI development, with a particular focus on the "access" question. In an effort to provide an interdisciplinary discussion on AI auditing, this piece is an unusual blend of styles. Our primary goal is to connect tools familiar to technologists with concepts that resonate with regulators; namely, hypothesis testing, evidence gathering, and legal presumption. This piece is therefore intended for two audiences: (i) policymakers interested in the implementation of AI audits, and (ii) computer scientists interested in developing audits.

2 Background

In this section, we briefly review auditing and its relation to AI. Of particular note are Table 1, which summarizes recent AI auditing efforts, and Section 2.3, which reviews related work.

2.1 Auditing: A brief summary

An audit is an assessment of an organization, entity, or process. Audits are conducted for many reasons, including (i) testing for *compliance with regulations*, (ii) determining whether a technology meets *certification standards*, (iii) validating *claims made by system designers*, and (iv) monitoring an organization's *internal practices*. As examples, the 2002 Sarbanes-Oxley Act made financial auditing commonplace as a tool to detect fraud and confirm the accuracy and completeness of financial reports; to earn a fair trade certification, vendors regularly undergo audits to ensure they uphold fair trade practices; and any organizations audit themselves to detect, e.g., financial waste.

Generally, there are three types of auditors: internal, external, and third-party auditors. An **internal** auditor is selected from within an organization that seeks to audit itself. An **external** auditor is an independent party that is hired by the organization (e.g., its board of directors) to perform an audit. External auditors typically enter into a contractual agreement with the organization that outlines, for instance, the scope of the audit and level of engagement. Like an external auditor, a **third-party** auditor is not a part of the organization being audited, but they are additionally not hired by the organization that they are auditing. A third-party auditor may be employed by another entity, such as a regulatory agency. In order to ensure that auditors provide objective and high quality reports, *auditors are themselves subject to audits*. It is customary for auditors to audit one another and, in certain industries, auditors are overseen by government agencies, such as the Public Company Accounting Oversight Board (PCAOB) in the US and the Financial Reporting Council (FRC) in the UK.

Furthermore, audits can also be run retrospectively or prospectively. **Retrospective audits** evaluate a system's past behavior. For example, audits that examine financial records or system performance are retrospective. **Prospective audits** characterize audits that are either (i) performed

before deployment or (ii) run on settings that have not yet been encountered. As an example of (i), prospective audits can be run prior to an AI system's release or after major modifications. As an example of (ii), prospective audits can also be performed in an *ongoing* manner and evaluating how the system *would* behave under conditions that have not yet occurred by probing the system.

Although out of the scope of this piece, one final consideration for auditing is set of the **properties** or **standards** one is auditing for, which has become a central focus of the U.S. National Institute for Standards and Technology (NIST) and European Commission.

2.2 AI auditing

The auditing of artificial intelligence (AI) systems is fairly nascent. In this section, we discuss legislation that explicitly requires AI audits, legislation that indirectly mandates AI audits, and other contexts in which AI audits arise.

Legally required AI audits. While some organizations audit themselves and some organizations are audited by researchers, there have historically been few regulations that legally require auditing. That is starting to change, and Table 1 in the Appendix provides a (non-exhaustive) list of auditing requirements for the European Union (EU) Artificial Intelligence (AI) Act, the EU General Data Protection Regulation (GDPR), the California Consumer Privacy Act (CCPA), the US Algorithmic Accountability Act (AAA), the New York City (NYC) Local Law 144, Canada's Directive on Automated Decision-Making (CDADM), and the EU Digital Services Act (DSA).

Laws that indirectly result in AI audits. In addition to the laws detailed in Table 1, which explicitly require AI audits, there are several domains in which AI audits are indirectly required. For example, the Dodd-Frank Wall Street Reform and Consumer Protection Act and Sarbanes-Oxley Act (SOX) both require audits as a part of the compliance process. Although neither explicitly mention AI or algorithms, both AI and algorithms have become common tools in the financial services sector, meaning that both acts indirectly mandate audits of AI systems.

Third-party compliance audits. Lastly, audits are a common tool for ensuring compliance with the law, even when the audits themselves they are not explicitly mentioned. Regulatory bodies and third-parties will often audit in order to hold organizations legally or publicly accountable. For example, the Children's Online Privacy Protection Act (COPPA) does not explicitly mention audits, but the Federal Trade Commission (FTC) conducts investigations to check for compliance with COPPA's mandates. As a result, many companies also perform internal audits and regularly monitor their systems to ensure ongoing compliance. Informally, researchers and journalists often perform audits to hold AI developers and operations publicly accountable by, e.g., uncovering flaws in security, reporting unexpected behavior in large language models (LLMs) (Nasr et al. 2023), and surfacing bias or discrimination (Chouldechova 2016).

Remark. As a final note, auditors are typically granted limited access to the systems that they are auditing. In some cases, this limited access is written in the law, e.g., the CCPA states that "nothing in this section [on risk assessments] shall require a business to divulge trade secrets" and GDPR similarly states that "each measure should be appropriate, necessary and proportionate in view of ensuring compliance with this Regulation." In other cases, such as third-party audits, auditors complete their assessments without cooperation from the organization being audited.

Table 1: Examples of legislation that require audits of data-driven or AI algorithms

Law	Enforced by	Performed by	Audit frequency	Audit requirements	Penalty
EU GDPR (2016)	Data Protection Authorities in each EU member state	Data controllers (typically internal)	Before any high-risk data processing	Data Protection Impact Assessments (DPIAs): Descriptions of envisioned data processing operations, purposes, risks to rights & freedoms of data subjects, measures to address risks	Up to €20M or 4% of annual worldwide turnover, whichever is higher
EU AI Act (2023)	National authorities in EU member states	AI system providers (internal); must give national competent authorities & notified bodies access (third-party)	Before high-risk system on market, ongoing post-market monitoring, and whenever system is substantially modified	High-risk AI systems must undergo conformity assessments to ensure they meet requirements for safety, transparency, human oversight, data, and more (as laid ou in Title III, Chapter 2)	Determined by member states; Some infringements up to €30M or 6% of annual worldwide turnover, whichever is higher
CCPA (2018)	California Attorney General	Businesses whose data processing presents significant risks to consumer privacy or security	Cybersecurity Audit must be performed on annual basis; Risk Assessment performed on regular basis (unspecified)	Cybersecurity Audit must assess effectiveness of business' cybersecurity measures in protecting consumer personal information. Risk Assessment should evaluate the processing of personal information and weigh the benefits of processing against potential risks to consumer rights	Up to $7.5K per intentional violation; additional penalties given by California Privacy Protection Agency
US AAA (2023†)	Federal Trade Commission (FTC)	Covered entities (businesses using AI systems)	Ongoing testing and evaluation with annual reports	Evaluation of automated decision system's or augmented critical decision process' potential impacts on consumers, considering privacy, bias, fairness, transparency, and more	Determined by the FTC
NYC 144 (2021)	NYC Dept. of Consumer & Worker Protection	Employer/agency using Automated Employment Decision Tool (internal); can use independent auditor (external)	Prior to first use and annually	Checks whether automated employment decision tools have disparate impact on persons of any "component 1 category"; summary must be made publicly available	Up to $1.5K per instance; others determined by enforcement body
CDADM (2019)	Treasury Board of Canada Secretariat	Federal institutions using automated decision systems	Early in development, before production, and after major changes to system	Requires assessments and reports on the use of automated decision-making systemsand ther effect on individual or community rights, economic interests, sustaintability, and more.	Unspecified, as determined by the Treasury Board
EU DSA (2022)	Digital Service Coordinators in each EU member state and the EC	Independent organizations with restrictions (e.g., cannot audit > 10 consecutive years or provide non-audit services 1 year before/after audit)	Annually	Tests compliance with the obligations set out in Chapter III of the DSA and voluntary commitments (e.g., in code of conduct or crisis protocol)	Up to 6% of annual worldwide turnover; ongoing penalties until cease infringement of up to 5% average daily turnover

† Proposed but not passed

2.3 AI auditing techniques

Background: Audit studies. Audit studies have a long history in the social sciences. Bertrand and Mullainathan (2004) find in a study of labor market discrimination that resumes with White-sounding names receive 50 percent more callbacks, on average, than identical resumes with African-American-sounding names. This evidence was gathered by submitting a set of fictitious resumes in response to real help-wanted ads, allowing researchers to experimentally manipulate the perceived race of job applicants in a way akin to the black-box audits described in this work.

Taking inspiration from this tradition, there is now a growing body of literature that audits algorithmic systems for evidence of consumer harm. Investigators have employed audit studies to examine self-preferencing in search results (Luca et al. 2016, Edelman and Lai 2016, Jeffries and Yin 2020, Gleason et al. 2023), discrimination in online platforms (Sweeney 2013, Sandvig et al. 2014, Ayres et al. 2015, Datta et al. 2015, Edelman et al. 2015, Kricheli-Katz and Regev 2016, Wagner et al. 2015, Hannák et al. 2017b, Metaxa et al. 2021a), and the effects of algorithmic personalization (particularly on political polarization) (Kliman-Silver et al. 2015, Hannák et al. 2017a, Metaxa et al. 2019, Huszár et al. 2022, Nyhan et al. 2023, Hosseinmardi et al. 2023). A key challenge is that the inputs to these systems are often highly complex, and may not be directly manipulable by researchers. This motivates other causal identification strategies, e.g., by identifying natural experiments in observational data (see Yao et al. (2020) for a recent survey or Angrist and Pischke (2008), Pearl (2009), Imbens and Rubin (2015) for textbook treatments). For additional background on audit studies, including the legal and ethical questions that arise, as well as recommendations for best practices, we refer to Metaxa et al. (2021b). For systematic reviews of the algorithm auditing literature, we refer to Bandy (2021) and Urman et al. (2024).

Frameworks for algorithmic auditing. Perhaps most closely related to this work are general frameworks for ensuring that algorithms satisfy normative and regulatory constraints. Raji et al. (2020) propose a framework which guides the development life cycle of an algorithmic decision pipeline, and Mitchell et al. (2019) propose standardized documentation and benchmarks to improve model transparency. This work is complementary to ours, as the guidelines are targeted at developers who can access the inner workings of the algorithm. In contrast, we provide an in-depth discussion of black-box audits, propose a way to translate between the law and audit procedure, and describe an open problem related to query complexity. Lam et al. (2023) take a different perspective, and instead propose the notion of a *socio-technical* audit to directly study the interplay between algorithms and their users. In particular, a socio-technical audit involves experimentally manipulating the *outputs* of an algorithm—for example, via a browser extension which manipulates search results or social media feeds—to study human components of a system (e.g., how user react or modify their behavior) in addition to algorithmic components.

Finally, (Blattner et al. 2021) propose a framework for regulating algorithms based on model explanations that are tailored to capture specific model characteristics—for example, racial disparities in the model's predictions—rather than to best explain the model's average performance. We further discuss the relationship of auditing and these interpretability techniques in Appendix A.

Black-box auditing. Our work focuses on black-box auditing, where the auditor may only *query* the model, rather than e.g., inspecting source code, model architecture or training procedures. This approach is intended to enable third party oversight of algorithms (Raji et al. 2022), even in the face of limited cooperation by algorithm providers (Costanza-Chock et al. 2023). This aligns with the perspective taken in Cen and Shah (2020), Cen et al. (2023b), which propose auditing procedures for algorithms which curate content on social media platforms. It is also the approach

taken in Rastegarpanah et al. (2021), who develop algorithms for testing compliance with the GDPR's data minimization principle (that an algorithm uses only "the minimal information that is necessary for performing the task at hand" (Rastegarpanah et al. 2021)). This is also similar to the perspective taken in Lee (2022), Akpinar et al. (2022), which propose the use of black-box audits to assess *counterfactuals*. For example, such an audit might ask whether, for a given individual, the algorithmic recommendation changes if the individual's race were different. As we discuss in Section 5, these localized audits can be useful for individuals seeking recourse for algorithmic harms.

Finally, contemporaneous work by Casper et al. (2024) argues that a black-box approach is insufficient for rigorous auditing, and highlight the limitations of black-box queries. These include (1) the difficulty of developing a global understanding of how a system behaves, (2) the inability to study system components separately, (3) the possibility that overly simplistic black-box audits can produce misleading results, (4) the limitations of black-box interpretability methods and (5) the inability to suggest *remedies* when models are noncompliant. We share the view that broader access (e.g., to model weights, gradients or source code) can enable more in depth auditing of algorithmic systems, and we discuss the benefits and limitations of black-box auditing at length in Section 3. Given the strictly controlled access provided to auditors (see our remark above) and concerns such as privacy, our discussion of black-box audits is driven by a desire to explore what can be achieved with black-box access, which can be supplemented with further access to cover its blind spots.

Nonetheless, we present constructive results indicating that black-box access is sufficient to audit for many properties of interest. We discuss our focus on black-box auditing at length in Section 3. We also discuss additional related work in Appendix A.

3 Types of auditing access

Auditing AI systems necessarily requires granting the auditor some form of *access* to the underlying model(s). As discussed in the previous section, the appropriate level of access is context-specific and must balance competing priorities, such as data privacy, intellectual property protections, resource constraints, necessity, proportionality, and more. In this section we discuss the relative merits of four kinds of access: access to the training data, the training procedure, the model architecture, and white- and black-box access to the trained model. Our discussion below hews closely to that of Cen et al. (2023a).

3.1 Option 1: Access to training data

AI models generate outputs by learning patterns and relationships that are exhibited in their *training data*. Because this dataset is so foundational to an AI system, an auditor may wish to audit the training data. Indeed, the data on which a model is trained can be suggestive of potential harms and failure modes. For example, over- or under-representation of a population in the training data can lead to bias (Chouldechova and Roth 2018). Similarly, differences between the test and training data distributions can lead to generalization failures (Zhou et al. 2021).

Nonetheless, access to the training data alone is typically insufficient for a rigorous audit. The primary reason is that the same training data can induce many different downstream models, whose behavior ultimately depends on the entire training pipeline (e.g., the choice of hyperparameters, model architecture, and learning algorithm). Although they arise from the same training data, these models may differ substantially along nearly any dimension of interest, including accuracy, fairness, and robustness. This "model multiplicity" (Black et al. 2022) or "underspecification" (D'Amour et al. 2022) is an unavoidable feature of most modern machine learning pipelines. As

such, it is not only difficult, but often impossible for an auditor to conclusively characterize an AI system's behavior from its training data alone.

There are, still, many reasons why an auditor may wish to examine an AI system's training data. For instance, auditing a company's data acquisition, cleaning, balancing, privacy, and provenance practices can encourage good data hygiene. Although requiring that the training data meet a strict set of property requirements does not generally have the intended effect (most bright-line rules are easily circumvented due to the underspecification phenomenon identified above), *data disclosure audits* can encourage good company practices and prevent downstream harms. That is, auditors could assess the comprehensiveness and accuracy of information that companies disclose about their training data and practices. Consider, for example, a company wishing to use GPT. It is useful for this company to know the time frame across which GPT's training data is collected; data disclosure audits would check that critical information or appropriate warnings are given by AI providers.

3.2 Option 2: Access to training procedures

One can alternatively request access to an AI system's training procedure. By "training procedure," we mean the high-level steps that the AI developer took in order to produce the final, trained model. As simple examples, auditors could require that AI developers describe the broad class of models that they chose (e.g., transformers or decision trees), the objective functions that they optimize (e.g., the factors that a social media algorithm optimizes in its pipeline), and the algorithm that they applied on the chosen model in order to achieve the desired objective (e.g., stochastic gradient descent or CART), and other training information (e.g., the amount of training resources). One can think of the training procedure as a roadmap for how the AI system is trained and produced.

For example, it was found that Facebook's objective function weighted "reacts" five times higher than "likes" when inferring user preferences Lonas (2021), which resulted in the unintentional amplification of emotional content. An audit of Facebook's training procedure, including their objective functions, may have encouraged the company to scrutinize and justify (or, if appropriate, abandon) such design choices. In this way, auditing training procedures can serve as simple sanity checks that alert model developers to potential non-compliance and even proactively identify avenues for model improvement.

However, as we saw with training data access, the same training procedure can yield many downstream models, and there is no guarantee that they behave similarly. The resulting model depends on various other factors, including the training data, model weights at initialization, and more. Therefore, while an auditor who is given access to a system's training procedure can perform sanity checks, they cannot infer much more about the system's behavior. Furthermore, since the training procedure lays out the steps taken to produce the AI system, access to training procedures should be carefully controlled. Of the forms of access discussed in this section, the training procedure is arguably the most valuable information associated with a commercial product.

3.3 Option 3: Access to the model skeleton

The next form of access we consider is access to the model "skeleton" (or untrained model). By "skeleton," we mean the specific model (e.g., neural network architecture) that is used, without the parameters, training data, or training procedure.

The defining feature of this form of access is that it reveals the key interfaces within the machine learning pipeline. From the model skeleton, an auditor can determine the expected inputs (e.g., types of features) and outputs (e.g., a number between 0 and 1) of the model. The auditor can also ascertain how many components make up the AI system and the relationship between different

components of the AI system. For example, suppose that a job applicant's information and resume are first sorted into one of several job categories, processed by appropriate algorithms, before being assigned a score between 0 and 1, which is finally thresholded to produce a hiring recommendation. Then, this entire "flow" would be captured by the model skeleton.

Access to the model skeleton provides perhaps the most interpretable view of an AI system. Indeed, when the social media platform X (formerly known as Twitter) voluntarily released their recommendation model skeleton, (twi), it revealed qualitative insights into X's content curation algorithm. For example, the public could glean that X "sources half of a user's content from in-network Tweets (i.e., from accounts that the user follows) and the other half from out-of-network Tweets" (Cen et al. 2023a). As with the training procedure, a model skeleton allows the auditor to perform sanity checks to ensure that an AI system does not have obvious flaws. It can even be used to identify discrepancies between an AI company's claims and the deployed system.

The model skeleton alone, however, is not enough to characterize the precise behavior of an AI system. Indeed, as mentioned in Section 3.1, even models with the exact same skeleton can behave very differently from one another. In this way, it is difficult to verify whether an AI system complies with a specific rule or meets a given standard from access to the model skeleton alone. Moreover, depending on the technical fluency of an auditor, the skeleton may be too opaque to determine the implications of choosing one model skeleton over another.

3.4 Option 4: White-box and black-box access to the trained model

The final option is to provide access to the final, trained model. This option is particularly appealing because an auditor can directly test and probe the end product. That is, they can interface with the same system that is ultimately deployed. Unlike the previous options, access to a trained model allows an auditor to unambiguously determine how the model would behave.

There are two versions of providing access to the final model: white-box access (access to the entire trained model including the weights) and black-box access (the ability to probe the model on inputs and observations of the outputs, as defined in Section 1).[2]

An auditor can, from black-box access alone, test whether an AI system satisfies certain criteria of interest, from how well it performs on an outlier population to whether it has disparate impact on different races. White-box access to the trained model, which strictly subsumes black-box access, provides the auditor with significantly more information. An auditor could, for example, take gradients with respect to various inputs of interest (a technique that has been used to gain insight into the "logic" behind an AI model). There is precedent for white-box access in other domains, such as the automotive industry, where an inspector can examine the vehicle in its entirety.

Although white-box access is appealing, black-box access is often sufficient. By analogy, an auditor with black-box access can crash test a car whereas an auditor with white-box access would also be able to inspect every component of the car. Black-box access would allow an auditor to test how the model behaves end-to-end without necessarily requiring that the auditor be technically proficient (which is often needed if an auditor wishes to leverage the white-box access option).[3] Because black-box access can reduce the risk of leaking proprietary information, it might all that is "necessary" or "proportionate" depending on the context of interest.

[2]Note that white-box access to the training pipeline and white-box access to the trained model are distinct. "White-box" generally refers to unhindered access, which can be applied to different parts of the AI pipeline. By contrast, "black-box access" is distinct and generally only corresponds to the definition given in Section 1.

[3]Note, however, that if auditors opt for black-box access, it might be appropriate to cap the number of queries since any trained model can, in theory, be fully reconstructed from infinite black-box queries.

Compared to the other three options discussed in this post, both white- and black-box access to the trained model are outcome-focused. They do not heed an AI developer's intention, philosophy, or technique. They ignore the means used to obtain an AI system, concentrating solely on the ends. They assess an AI model based on its end-to-end behavior. In a way, this approach is desirable. Indeed, even if AI developers curate a pristine training dataset, their AI model can still produce poor results, which only becomes apparent with access to the trained model. On the flip side, access to the trained model but nothing else is not always satisfactory from a broader accountability perspective. An individual contesting an AI-driven decision may, for instance, wish to understand how that decision came about—to trace the choices an AI developer made that led to the decision.

3.5 Considering all the options

Of the options described above, the fourth—access to the trained model—provides an auditor with the greatest flexibility with the least amount of ambiguity. That is, an auditor can test a model based on various criteria (e.g., to determine whether it satisfies a property known as calibration) and remain confident that their findings pertain to the specific AI system of interest. In contrast, an auditor cannot definitively say whether an AI system satisfies, for example, calibration from the training data, training procedure, or untrained model alone.

Still, none of the four options stands above the rest in every way. The first three options serve as useful sanity checks, speak to the intentions of an AI developer, and encourage AI developers to adopt good practices. Moreover, they can ensure that other accountability mechanisms are achievable—for example, an individual contesting an AI-driven decision can cite poorly cleaned data (as provided by Option 1) or an overly simplistic objective function (as provided by Option 2) to argue that the AI-driven decision is inappropriate for them.

Auditors can therefore complement access to trained models with limited and carefully chosen access to the training data, training procedure, and untrained model. Auditors can even adopt a tiered system, where companies of different magnitudes face different access requirements. There are, additionally, other forms of access that we omit in our discussion, such as API access and access to a model's training checkpoints.

4 Black-box auditing as hypothesis testing

Thus far, we have discussed the benefits and challenges facing AI auditing, focusing in particular on what information an auditor can glean from different forms of access to an AI system. Our discussion led us to black-box auditing, which is compelling in its ability to directly probe an AI system. In this section, we discuss how black-box auditing can be formalized as hypothesis testing.

Although hypothesis testing as a tool for auditing is not new, our main contribution is to identify precisely how hypothesis testing using black-box access parallels legal procedure. In other words, *we propose that hypothesis testing can serve as a "translation" between the law and the implementation of audits*. Given this translation, policymakers would not need to design audits from scratch each time, nor would they need to fully understand every AI system that is being audited. Rather, policymakers would simply need to specify the parameters of the hypothesis test.

In particular, we discuss how the choice of null hypothesis in a hypothesis test can be viewed as a legal presumption, placing the burden of proof on the party that wishes to either certify compliance or demonstrate the noncompliance of a particular algorithm. We further touch on how hypothesis testing maps to other components of the auditing process, including the balance between

gathering sufficient evidence and protecting trade secrets, and the way in which the error tolerance of a hypothesis test can model auditor leniency.

4.1 Setup

Consider a model developer or operator, who we refer to as the *AI provider* for the remainder of this work. The provider employs an algorithm $f \in \mathcal{F}$, where $\mathcal{F}$ is a class of mappings from values in $\mathcal{X}$ to distributions over $\mathcal{Y}$ as denoted by $\Delta(\mathcal{Y})$ (we assume throughout that $\mathcal{Y}$ is countable). For example, in the context of lending decisions, f could map an applicant's characteristics $x \in \mathcal{X}$ to a measure $f(x) \in \mathcal{Y} \subset [0, 1]$ of the applicant's creditworthiness. Let p_x denote the true (possibly unknown) marginal distribution of x.

The auditor is interested in determining whether the provider's algorithm f complies with a requirement of interest. We denote this requirement by $g : \mathcal{F} \to \mathbb{R}$.

Definition 1. *We say that an algorithm $f \in \mathcal{F}$ is* g-compliant *if and only if $g(f) \leq 0$.*

When the property g is clear from context, we simply say an algorithm is *compliant*. In a *black-box audit*, the auditor has access to N input-output pairs $(x_i, f(x_i))$. We refer to the pairs as the auditor's *evidence* and denote it by $\mathcal{E} = \{(x_i, f(x_i)) : i = 1, \ldots, N\}$.

In practice, the auditor has a limited amount of evidence due to practical considerations (gathering evidence $\mathcal{E}$ can be costly) as well as concerns that if m is too large relative to the complexity of $\mathcal{F}$ or $\mathcal{X}$, the auditor can reverse engineer the algorithm f, which may be an intellectual property concern.[4] The auditor's task is then as follows:

Determine whether f is g-compliant given a limited set of evidence $\mathcal{E}$.

We will make the notion of "limited" precise in the following sections. First, we provide several examples illustrating the scope of Definition 1.

Example 1 (Maximum loss)**.** *Requiring that f's maximum loss ℓ over some $S \subseteq \mathcal{X}$ is at most η is equivalent to requiring that $g(f) \leq 0$, where $g(f) = \max_{x \in S} \ell(f(x), x) - \eta$. Depending on the definition of loss, one can audit for minimax fairness (by defining loss as negative performance), worst-case harm (by defining loss as the output's harm, e.g., toxicity level), and even copyright infringement (by defining loss as the dissimilarity between x and the copyrighted work).*

Example 2 (Group fairness)**.** *In the area of algorithmic fairness, group fairness generally reflects a notion of parity across groups. For example, one notion of group fairness known as "statistical parity" requires that the rate at which a binary classifier $f : \mathcal{X} \to \{0, 1\}$ selects members of group G_1 is at most $\eta > 0$ far from the rate at which f selects members of group G_2 under some distribution p_x over $\mathcal{X}$. This is equivalent to requiring that $g(f) \leq 0$, where*

$$g(f) = |\mathbb{E}_{p_x}[f(x) \,|\, x_G = G_1] - \mathbb{E}_{p_x}[f(x) \,|\, x_G = G_2]| - \eta.$$

The expectation above is taken over $x \sim p_x$, and $x_G \in \{G_1, G_2\}$ is the feature in x denoting group membership.

Example 3 (Individual fairness)**.** *Another notion of algorithmic fairness requires that "similar individuals be treated similarly," as captured by the criterion: $D(f(x), f(x')) \leq Ld(x, x')$ for all $x, x' \in \mathcal{X}$; distance metrics D and d on $\mathcal{Y}$ and $\mathcal{X}$, respectively; and Lipschitz constant $L > 0$ (Dwork et al. 2011). This is equivalent to requiring that $g(f) \leq 0$, where $g(f) = \max_{x,x' \in \mathcal{X}} \frac{D(f(x), f(x'))}{d(x,x')} - L$.*

Other properties, including calibration and differential privacy, can also be cast under Definition 1.

[4] For example, if f is deterministic and $N \geq |\mathcal{X}|$, then the auditor can recover f's exact behavior.

4.2 Black-box auditing as hypothesis testing

Given an algorithm f and auditing criterion g, the auditor's goal is to determine whether f is g-compliant using $\mathcal{E}$. Below, we discuss two possible hypothesis tests and their implications before describing the general hypothesis testing procedure in Section 4.3.

Presumption of compliance. Consider an auditor who seeks to discern which of the following hypotheses holds:

$$H_0 : g(f) \leq 0, \qquad H_1 : g(f) > 0. \tag{1}$$

Let $H \in \{H_0, H_1\}$ denote the ground-truth state. For example, if f is compliant, then $H = H_0$; if f is not, then $H = H_1$. The auditor does not know H a priori (otherwise, the auditor would know whether f is or is not compliant before the audit). Therefore, the auditor's goal is to develop a decision test or *rule* $\hat{H}$ such that $\hat{H}(\mathcal{E}) = H_0$ if the auditor believes f is compliant and $\hat{H}(\mathcal{E}) = H_1$, otherwise. The auditor would like $\hat{H}$ to match H for all $f \in \mathcal{F}$, as we formalize in Section 4.3.

In statistical inference, H_0 is known as the *null hypothesis*. In practice, the implication is that the auditor's *presumption* under (1) is that f is compliant. The auditor therefore assumes (and reports) that f is compliant unless the evidence allows them to confidently reject this presumption.

The following question arises: Should the auditor always presume compliance? As shown next, the hypothesis test can be reversed.

Presumption of non-compliance. Consider a different set of hypotheses:

$$J_0 : g(f) > 0, \qquad J_1 : g(f) \leq 0 \tag{2}$$

Relative to (1), the null and alternate hypotheses have been swapped. As before, there is a ground-truth state $J \in \{J_0, J_1\}$, and the auditor's goal is to develop a decision rule $\hat{J}$ such that, given evidence $\mathcal{E}$, the decision $\hat{J}$ approximates J well across all $f \in \mathcal{F}$. In this case, the null hypothesis J_0 (i.e., the legal presumption) is that the algorithm if not compliant.

Burden of proof: Which test should the auditor use? The null hypothesis reflects the auditor's presumption and, accordingly, who bears the burden of proof in the auditing process. Suppose, for example, there is a law requiring that AI providers are non-discriminatory, but the law does not require the AI provider to disclose any information to auditors. Then, under (1), the AI provider is not incentivized to disclose any information (i.e., to contribute any evidence $\mathcal{E}$ to the auditing process): since the auditor can only reject the null hypothesis $H_0 : g(f) \leq 0$ if they have enough evidence to do so, the burden of proof is on the auditor (or corresponding plaintiff).

On the other hand, under (2), the burden of proof is on the AI provider. That is, the AI provider is incentivized to give the auditor enough evidence to convince the auditor to reject the null hypothesis J_0. In this way, the choice of hypothesis test should reflect the desired legal presumption and corresponding placement of burden of proof. This choice may vary across contexts. For example, if the auditor's evidentiary burden is too great under (1), and the law may wish to shift the evidentiary burden by adopting (2).

4.3 Hypothesis testing procedure

Our primary goal is to examine the suitability of hypothesis testing as a framework for black-box auditing. As such, we describe the general procedure and considerations for hypothesis testing

below. We refer readers interested in additional background on hypothesis testing to Casella and Berger (2008) for a textbook treatment.

For the remainder of this work, we adopt a presumption of compliance as given in (1), though our results can be equivalently applied to (2). We discuss four main components of hypothesis testing next: the evidence, decision rule, model, and tolerance.

1. **Evidence.** The auditor has access to evidence $\mathcal{E}$, as defined in Section 4.2. Note that this evidence may be supplemented with other information (e.g., from API access or access to an open-source model related to f). The auditor is generally limited in the amount of evidence they can gather, for example, due to strictly controlled access to algorithm f or its training data or due to limited resources.

2. **Decision rule.** Given evidence $\mathcal{E}$, the auditor's goal is to develop an audit—in other words, a decision rule $\hat{H}$—that maps evidence $\mathcal{E}$ to a decision H_0 or H_1, which correspond to deciding whether to report that f is compliant or non-compliant, respectively. As discussed in Section 4.2, the auditor adopts the default decision H_0 unless the evidence is convincing enough for the auditor to reject H_0, as we review next.

3. **Design criteria & tolerance.** A decision rule $\hat{H}$ is evaluated based on two quantities: the false positive rate (FPR) and true positive rate (TPR):

$$\text{FPR} = \mathbb{P}\left(\hat{H} = H_1 | H = H_0\right),$$
$$\text{TPR} = \mathbb{P}\left(\hat{H} = H_1 | H = H_1\right),$$

   where $\mathbb{P}$ is taken with respect to randomness in the evidence $\mathcal{E}$ and decision rule $\hat{H}$. (Observe that the true negative rate and false negative rate can be computed directly from the FPR and TPR.) The field of hypothesis testing is largely concerned with finding rules that maximize the TPR while minimizing the FPR. Although we do not review them here, one approach is to restrict the maximum allowable FPR (known as the *significance level*) to ζ and find the decision rule that achieves the maximum TPR among all rules that have an FPR no more than ζ and for all possible algorithms in $\mathcal{F}$. This rule is known as the uniformly most powerful (UMP) test and can be viewed as an ideal benchmark (though it does not always exist). The maximum allowable FPR can be viewed as the *tolerance* of an audit.

4. **Model.** The final ingredient of any hypothesis test is the model. To explain the model, consider the following intuition. Given some evidence $\mathcal{E}$, the auditor's job is determine whether $g(f) \leq 0$. In other words, the auditor would like to use $\mathcal{E}$ to infer something about f. In order to do so, the auditor must make some assumption about how f generates $\mathcal{E}$; that is, to map from $\mathcal{E}$ back to f, the auditor needs a model of how f maps to $\mathcal{E}$.

   The model is ultimately what allows the auditor to develop an appropriate decision rule $\hat{H}$ and underlies both the FPR and TPR. Without a model, the auditor lacks any assumptions on which to build a decision rule; moreover, both FPR and TPR are implicitly defined with respect to a model (i.e., data generating process).

   The model maps the possible true states H_0 and H_1 to evidence $\mathcal{E} = \{(x, f(x)) : x \in \bar{\mathcal{X}}\}$. Thus, given a model, the auditor seeks to estimate the true state from evidence. For example, the auditor may assume that x are drawn from some distribution $\mathcal{D}$. The auditor knows that the outputs $f(x)$ are determined by f. By definition, the auditor does not know the very algorithm f that they wish to audit, but the auditor may assume that f belongs to some

model family $\bar{\mathcal{F}} \subset \mathcal{F}$. In this case, the model is determined by $(\mathcal{D}, \bar{\mathcal{F}})$.

To construct a decision rule satisfying the design criteria above, the auditor must necessarily make assumptions that, together, form a *model*. The model should describe the data generating process mapping true states H_0 and H_1 to evidence $\mathcal{E}$. For example, the auditor may assume that the input-output pairs $(x_i, f(x_i))$ are drawn i.i.d. from some known distribution p_x and that the unknown model f lies in some known class $\mathcal{F}_A$ (e.g., the set of all linear classifiers). Given a model mapping the true state H to the evidence $\mathcal{E}$, the auditor seeks to estimate H from $\mathcal{E}$. The choice of model is ultimately up the auditor, who faces a key trade-off: a model should be general enough to capture the true data generating process while also being narrow enough for the auditor to formulate $\hat{H}$.

Summary. Putting these four components together, the auditor takes the following steps: decide on an appropriate model and tolerance, develop a decision rule, gather evidence, and apply the decision rule. As mentioned previously, the auditor faces several constraints, foremost of which is limited evidence $\mathcal{E}$ with which to conduct the audit. The auditor may also have other concerns, such as ensuring that the audit is manipulation-proof (Yan and Zhang 2022).

5 Limitations, challenges and future work

While black-box auditing is a powerful tool for detecting algorithmic harms, we view our approach as merely one component of a holistic approach to ensure responsible oversight of algorithms. Furthermore, while the framework we propose is quite general, it is subject to a number of technical limitations which limit its practical scope. In this section we provide an overview of these limitations and suggest promising directions for future work.

Multiple hypothesis testing. While our work considers testing a single property of a model, auditors are typically interested in auditing for multiple criteria or running repeated audits over time. This can present additional challenges, as (1) the reuse of data across audits will invalidate basic statistical guarantees and (2) even audits run on independent samples will not (on their own) control the family-wise error rate or false discovery rate (Benjamini and Hochberg 1995). These issues are exacerbated when the number of audits is not known ex-ante, and may depend on the results of prior audits. While this is a well-studied problem in the statistics literature (Benjamini and Hochberg 1995, Tian and Ramdas 2019), approaches which are tailored to a specific auditing context may yield additional performance improvements. For example, the auditor may believe that the results of certain audits are independent from one another, or may use the kind of statistical audits we propose here to surface issues which can then be investigated more thoroughly (e.g., by requiring additional disclosure from the AI provider).

Choosing queries S. A key challenge in black-box auditing is choosing the set of inputs $S \subseteq \mathcal{X}$ on which to query f. The simplest methods sample i.i.d. from some population of interest, or otherwise specify S *a priori*. This can be a challenging task, particularly when the reference population is hard to define. Furthermore, it is sometimes more natural to construct queries in an *online* fashion (e.g., Yan and Zhang (2022)), where successive queries are chosen conditional on the output of prior queries. We leave a more detailed exploration of adaptive black-box auditing to future work.

Explanations, recourse and the limits of auditing. While auditing can be a powerful tool for *revealing* unwanted behavior, it does not necessarily indicate what the provider should do to correct or mitigate these issues. Indeed, as argued in Casper et al. (2024) and discussed in Section 3, is it possible that 'white-box' approaches can be more informative in this regard. Furthermore, black-box auditing does not necessarily enable appropriate recourse when an individual is harmed by an algorithm. In particular, the result of a black-box audit do not always reveal whether the model make a mistake or otherwise behaved unreasonably on a *specific instance*. In such cases, a more localized (or 'counterfactual-based') approach to auditing might be appropriate (Cen and Raghavan 2022, Lee 2022, Akpinar et al. 2022).

Acknowledgments

The authors gratefully acknowledge funding from the MIT-IBM project on Causal Representation, the National Science Foundation (NSF) grant CNS-1955997, and the Air Force Research Laboratory (AFOSR) grant FA9550-23-1-0301.

References

Twitter's recommendation algorithm. https://blog.twitter.com/engineering/en_us/topics/open-source/2023/twitter-recommendation-algorithm.

Compas risk scales : Demonstrating accuracy equity and predictive parity performance of the compas risk scales in broward county. 2016. URL https://api.semanticscholar.org/CorpusID:51920414.

Philip Adler, Casey Falk, Sorelle A. Friedler, Gabriel Rybeck, Carlos Scheidegger, Brandon Smith, and Suresh Venkatasubramanian. Auditing black-box models for indirect influence, 2016.

Nil-Jana Akpinar, Liu Leqi, Dylan Hadfield-Menell, and Zachary Lipton. Counterfactual metrics for auditing black-box recommender systems for ethical concerns. In *ICML 2022 Workshop on Responsible Decision Making in Dynamic Environments*, Baltimore, Maryland, USA, 2022.

Rohan Alur, Loren Laine, Darrick K. Li, Manish Raghavan, Devavrat Shah, and Dennis Shung. Auditing for human expertise, 2023.

Rohan Alur, Manish Raghavan, and Devavrat Shah. Distinguishing the indistinguishable: Human expertise in algorithmic prediction, 2024.

Joshua D. Angrist and Jörn-Steffen Pischke. *Mostly Harmless Econometrics: An Empiricist's Companion*. Princeton University Press, 2008.

Ian Ayres, Mahzarin Banaji, and Christine Jolls. Race effects on ebay. *The RAND Journal of Economics*, 46(4):891–917, 2015.

Raghu Bahadur and Leonard Savage. The nonexistence of certain statistical procedures in nonparametric problems, 1956.

Jack Bandy. Problematic machine behavior: A systematic literature review of algorithm audits. *Proc. ACM Hum.-Comput. Interact.*, 5(CSCW1), apr 2021. doi: 10.1145/3449148. URL https://doi.org/10.1145/3449148.

Robert P Bartlett, Adair Morse, Nancy Wallace, and Richard Stanton. Algorithmic accountability: A legal and economic framework, 2019. URL http://faculty.haas.berkeley.edu/morse/research/papers/AlgorithmicAccountability_BartlettMorseStantonWallace.pdf.

Yoav Benjamini and Yosef Hochberg. Controlling the false discovery rate: A practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society. Series B (Methodological)*, 57(1):289–300, 1995.

Richard A. Berk, Hoda Heidari, Shahin Jabbari, Michael Kearns, and Aaron Roth. Fairness in criminal justice risk assessments: The state of the art. *Sociological Methods & Research*, 50:3 – 44, 2017. URL https://api.semanticscholar.org/CorpusID:12924416.

Marianne Bertrand and Sendhil Mullainathan. Are emily and greg more employable than lakisha and jamal? a field experiment on labor market discrimination. *Am. Econ. Rev.*, 94(4):991–1013, August 2004.

Felix Bieker, Michael Friedewald, Marit Hansen, Hannah Obersteller, and Martin Rost. A process for data protection impact assessment under the european general data protection regulation. In *Annual Privacy Forum*, 2016. URL https://api.semanticscholar.org/CorpusID:7904695.

Battista Biggio and Fabio Roli. Wild patterns: Ten years after the rise of adversarial machine learning. 2017. doi: 10.1016/j.patcog.2018.07.023.

Battista Biggio, Igino Corona, Davide Maiorca, Blaine Nelson, Nedim Srndic, Pavel Laskov, Giorgio Giacinto, and Fabio Roli. Evasion attacks against machine learning at test time. 2017. doi: 10.1007/978-3-642-40994-3_25.

Emily Black, Samuel Yeom, and Matt Fredrikson. Fliptest: Fairness testing via optimal transport. 2019. doi: 10.1145/3351095.3372845.

Emily Black, Manish Raghavan, and Solon Barocas. Model multiplicity: Opportunities, concerns, and solutions. In *2022 ACM Conference on Fairness, Accountability, and Transparency (FAccT '22)*, page 23, Seoul, Republic of Korea, June 21-24 2022. ACM. doi: 10.1145/3531146.3533149. URL https://www.cs.cmu.edu/afs/cs.cmu.edu/user/emilybla/www/Model_Multiplicity_3.pdf.

Laura Blattner, Scott Nelson, and Jann Spiess. Unpacking the black box: Regulating algorithmic decisions, 2021.

Wieland Brendel, Jonas Rauber, and Matthias Bethge. Decision-based adversarial attacks: Reliable attacks against black-box machine learning models. *ArXiv*, abs/1712.04248, 2017. URL https://api.semanticscholar.org/CorpusID:2410333.

Nadia Burkart and Marco F. Huber. A survey on the explainability of supervised machine learning. 2020. doi: 10.1613/jair.1.12228.

Toon Calders and Sicco Verwer. Three naive bayes approaches for discrimination-free classification. *Data Mining and Knowledge Discovery*, 21:277–292, 2010. URL https://api.semanticscholar.org/CorpusID:12856537.

Toon Calders, Faisal Kamiran, and Mykola Pechenizkiy. Building classifiers with independency constraints. *2009 IEEE International Conference on Data Mining Workshops*, pages 13–18, 2009. URL https://api.semanticscholar.org/CorpusID:3945595.

George Casella and Roger L. Berger. *Statistical Inference*. Thomson Learning, 2008.

Stephen Casper, Carson Ezell, Charlotte Siegmann, Noam Kolt, Taylor Lynn Curtis, Benjamin Bucknall, Andreas Haupt, Kevin Wei, Jérémy Scheurer, Marius Hobbhahn, Lee Sharkey, Satyapriya Krishna, Marvin Von Hagen, Silas Alberti, Alan Chan, Qinyi Sun, Michael Gerovitch, David Bau, Max Tegmark, David Krueger, and Dylan Hadfield-Menell. Black-box access is insufficient for rigorous ai audits, 2024.

Sarah H. Cen and Manish Raghavan. The right to be an exception to a data-driven rule, 2022.

Sarah H. Cen and Devavrat Shah. Regulating algorithmic filtering on social media. In *Neural Information Processing Systems*, 2020. URL https://api.semanticscholar.org/CorpusID:219721119.

Sarah H. Cen, Cosimo L. Fabrizio, James Siderius, Aleksander Madry, and Martha Minow. Auditing ai: How much access is needed to audit an ai system? https://aipolicy.substack.com/p/ai-accountability-transparency-2, 2023a.

Sarah H. Cen, Aleksander Madry, and Devavrat Shah. A user-driven framework for regulating and auditing social media, 2023b.

John J. Cherian and Emmanuel J. Candès. Statistical inference for fairness auditing, 2023.

Alexandra Chouldechova. Fair prediction with disparate impact: A study of bias in recidivism prediction instruments, 2016.

Alexandra Chouldechova and Aaron Roth. The frontiers of fairness in machine learning, 2018.

Sam Corbett-Davies, Emma Pierson, Avi Feller, Sharad Goel, and Aziz Z Huq. Algorithmic decision making and the cost of fairness. *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2017. URL https://api.semanticscholar.org/CorpusID:3228123.

Sasha Costanza-Chock, Emma Harvey, Inioluwa Deborah Raji, Martha Czernuszenko, and Joy Buolamwini. Who audits the auditors? recommendations from a field scan of the algorithmic auditing ecosystem. 2023. doi: 10.1145/3531146.3533213.

Cynthia S Crowson, Elizabeth J Atkinson, and Terry M Therneau. Assessing calibration of prognostic risk scores. *Stat. Methods Med. Res.*, 25(4):1692–1706, August 2016.

Nilesh Dalvi, Pedro Domingos, Mausam, Sumit Sanghai, and Deepak Verma. Adversarial classification. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '04, page 99–108, New York, NY, USA, 2004. Association for Computing Machinery. ISBN 1581138881. doi: 10.1145/1014052.1014066. URL https://doi.org/10.1145/1014052.1014066.

Alexander D'Amour, Katherine Heller, Dan Moldovan, Ben Adlam, Babak Alipanahi, Alex Beutel, Christina Chen, Jonathan Deaton, Jacob Eisenstein, Matthew D. Hoffman, Farhad Hormozdiari, Neil Houlsby, Shaobo Hou, Ghassen Jerfel, Alan Karthikesalingam, Mario Lucic, Yian Ma, Cory McLean, Diana Mincu, Akinori Mitani, Andrea Montanari, Zachary Nado, Vivek Natarajan, Christopher Nielson, Thomas F. Osborne, Rajiv Raman, Kim Ramasamy, Rory Sayres, Jessica Schrouff, Martin Seneviratne, Shannon Sequeira, Harini Suresh, Victor Veitch, Max Vladymyrov, Xuezhi Wang, Kellie Webster, Steve Yadlowsky, Taedong Yun, Xiaohua Zhai, and D. Sculley. Underspecification presents challenges for credibility in modern machine learning. *J. Mach. Learn. Res.*, 23(1), jan 2022. ISSN 1532-4435.

Amit Datta, Michael Carl Tschantz, and Anupam Datta. Automated experiments on ad privacy settings: A tale of opacity, choice, and discrimination. In *Privacy Enhancing Technologies Symposium*, 2015.

Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Rich Zemel. Fairness through awareness, 2011.

Benjamin Edelman and Zhenyu Lai. Design of search engine services: Channel interdependence in search engine results. *Journal of Marketing Research*, 53, 03 2016. doi: 10.1509/jmr.14.0528.

Benjamin G. Edelman, Michael Luca, and Dan Svirsky. Racial discrimination in the sharing economy: Evidence from a field experiment. http://ssrn.com/abstract=2701902, 2015.

Harrison Edwards and Amos J. Storkey. Censoring representations with an adversary. *CoRR*, abs/1511.05897, 2015. URL https://api.semanticscholar.org/CorpusID:4986726.

Michael Feldman, Sorelle A. Friedler, John Moeller, Carlos Eduardo Scheidegger, and Suresh Venkatasubramanian. Certifying and removing disparate impact. *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2014. URL https://api.semanticscholar.org/CorpusID:2077168.

Abraham D. Flaxman, Adam Tauman Kalai, and H. Brendan McMahan. Online convex optimization in the bandit setting: gradient descent without a gradient, 2004.

Anthony W. Flores, Kristin Bechtel, and Christopher T. Lowenkamp. False positives, false negatives, and false analyses: A rejoinder to "machine bias: There's software used across the country to predict future criminals. and it's biased against blacks". *Federal Probation*, 80:38, 2016. URL https://api.semanticscholar.org/CorpusID:15391140.

Jeffrey Gleason, Desheng Hu, Ronald E Robertson, and Christo Wilson. Google the gatekeeper: How search components affect clicks and attention. *Proceedings of the International AAAI Conference on Web and Social Media*, 17:245–256, 6 2023.

Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *CoRR*, abs/1412.6572, 2014. URL https://api.semanticscholar.org/CorpusID:6706414.

Steve Hanneke et al. Theory of disagreement-based active learning. *Foundations and Trends® in Machine Learning*, 7(2-3):131–309, 2014.

Anikó Hannák, Piotr Sapieżyński, Arash Molavi Khaki, David Lazer, Alan Mislove, and Christo Wilson. Measuring personalization of web search, 2017a.

Anikó Hannák, Claudia Wagner, David Garcia, Alan Mislove, Markus Strohmaier, and Christo Wilson. Bias in online freelance marketplaces: Evidence from taskrabbit and fiverr. In *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing*, pages 1914–1933, 2 2017b.

Moritz Hardt, Eric Price, and Nathan Srebro. Equality of opportunity in supervised learning, 2016.

Elad Hazan. Introduction to online convex optimization, 2019.

Homa Hosseinmardi, Amir Ghasemian, Miguel Rivera-Lanas, Manoel Horta Ribeiro, Robert West, and Duncan J. Watts. Causally estimating the effect of youtube's recommender system using counterfactual bots, 2023.

Ferenc Huszár, Sofia Ira Ktena, Conor O'Brien, Luca Belli, Andrew Schlaikjer, and Moritz Hardt. Algorithmic amplification of politics on twitter. *Proc. Natl. Acad. Sci. U. S. A.*, 119(1):e2025334119, January 2022.

Andrew Ilyas, Logan Engstrom, Anish Athalye, and Jessy Lin. Black-box adversarial attacks with limited queries and information. In *International Conference on Machine Learning*, 2018. URL https://api.semanticscholar.org/CorpusID:5046541.

Guido W. Imbens and Donald B. Rubin. *Causal Inference for Statistics, Social, and Biomedical Sciences: An Introduction*. Cambridge University Press, 2015.

Adrianne Jeffries and Leon Yin. Google's top search result? surprise! it's google. https://themarkup.org/google-thegiant/2020/07/28/google-search-results-prioritize-googleproducts-over-competitors, 2020. Accessed: 2023-04-18.

James E. Johndrow and Kristian Lum. An algorithm for removing sensitive information: Application to race-independent recidivism prediction. *The Annals of Applied Statistics*, 2017. URL https://api.semanticscholar.org/CorpusID:51782788.

Faisal Kamiran and Toon Calders. Classifying without discriminating. *2009 2nd International Conference on Computer, Control and Communication*, pages 1–6, 2009. URL https://api.semanticscholar.org/CorpusID:1102398.

Jon Kleinberg, Sendhil Mullainathan, and Manish Raghavan. Inherent trade-offs in the fair determination of risk scores, 2016.

C. Kliman-Silver, A. Hannak, D. Lazer, C. Wilson, and A. Mislove. Location, location, location: The impact of geolocation on web search personalization. In *Proceedings of the Internet Measurement Conference*, 2015.

Tamar Kricheli-Katz and Tali Regev. How many cents on the dollar? women and men in product markets. *Science Advances*, 2(2), 2016.

Alexey Kurakin, Ian J. Goodfellow, and Samy Bengio. Adversarial machine learning at scale. *ArXiv*, abs/1611.01236, 2016. URL https://api.semanticscholar.org/CorpusID:9059612.

Michelle S. Lam, Ayush Pandit, Colin H. Kalicki, Rachit Gupta, Poonam Sahoo, and Danaë Metaxa. Sociotechnical audits: Broadening the algorithm auditing lens to investigate targeted advertising. *Proc. ACM Hum.-Comput. Interact.*, 7(CSCW2), oct 2023. doi: 10.1145/3610209. URL https://doi.org/10.1145/3610209.

Seung C. Lee. A black box approach to auditing algorithms. *Issues In Information Systems*, 2022.

Zachary C. Lipton. The mythos of model interpretability, 2016.

Lexi Lonas. Facebook formula gave anger five times weight of likes, documents show — thehill.com. https://thehill.com/policy/technology/578548-facebook-formula-gave-anger-five-times-weight-of-likes-documents-show/, Oct 2021. [Accessed 26-03-2024].

Daniel Lowd and Christopher Meek. Adversarial learning. In *Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining*, KDD '05, page 641–647, New York, NY, USA, 2005a. Association for Computing Machinery. ISBN 159593135X. doi: 10.1145/1081870.1081950. URL https://doi.org/10.1145/1081870.1081950.

Daniel Lowd and Christopher Meek. Good word attacks on statistical spam filters. In *International Conference on Email and Anti-Spam*, 2005b. URL https://api.semanticscholar.org/CorpusID:1933015.

Michael Luca, Tim Wu, Sebastian Couvidat, Daniel Frank, and William Seltzer. Does google content degrade google search? experimental evidence. 2016. URL https://api.semanticscholar.org/CorpusID:53570166.

Scott Lundberg and Su-In Lee. A unified approach to interpreting model predictions, 2017.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks, 2017.

Subha Maity, Songkai Xue, Mikhail Yurochkin, and Yuekai Sun. Statistical inference for individual fairness, 2021.

Natalie Maus, Patrick Chao, Eric Wong, and Jacob Gardner. Black box adversarial prompting for foundation models, 2023.

Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. A survey on bias and fairness in machine learning, 2019.

Danaë Metaxa, Michelle A. Gan, and James A. Landay. An image of society: Gender and racial representation and impact in image search results for occupations. 2021a. URL https://api.semanticscholar.org/CorpusID:233322946.

Danaë Metaxa, Joon Sung Park, Ronald E Robertson, Karrie Karahalios, Christo Wilson, Jeff Hancock, and Christian Sandvig. Auditing algorithms: Understanding algorithmic systems from the outside in. *Found. Trends® Hum.–Comput. Interact.*, 14(4):272–344, 2021b.

Danaë Metaxa, Joon Park, James Landay, and Jeff Hancock. Search media and elections: A longitudinal investigation of political search results. *Proceedings of the ACM on Human-Computer Interaction*, 3: 1–17, 11 2019. doi: 10.1145/3359231.

Margaret Mitchell, Simone Wu, Andrew Zaldivar, Parker Barnes, Lucy Vasserman, Ben Hutchinson, Elena Spitzer, Inioluwa Deborah Raji, and Timnit Gebru. Model cards for model reporting. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 220–229, 1 2019.

Christoph Molnar. *Interpretable Machine Learning*. 2022. URL https://christophm.github.io/interpretable-ml-book/.

Milad Nasr, Nicholas Carlini, Jonathan Hayase, Matthew Jagielski, A Feder Cooper, Daphne Ippolito, Christopher A Choquette-Choo, Eric Wallace, Florian Tramèr, and Katherine Lee. Scalable extraction of training data from (production) language models. *arXiv preprint arXiv:2311.17035*, 2023.

Anh Nguyen, Jason Yosinski, and Jeff Clune. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images, 2014.

Brendan Nyhan, Jaime Settle, Emily Thorson, Magdalena Wojcieszak, Pablo Barberá, Annie Y Chen, Hunt Allcott, Taylor Brown, Adriana Crespo-Tenorio, Drew Dimmery, Deen Freelon, Matthew Gentzkow, Sandra González-Bailón, Andrew M Guess, Edward Kennedy, Young Mie Kim, David Lazer, Neil Malhotra, Devra Moehler, Jennifer Pan, Daniel Robert Thomas, Rebekah Tromble, Carlos Velasco Rivera, Arjun Wilkins, Beixian Xiong, Chad Kiewiet de Jonge, Annie Franco, Winter Mason, Natalie Jomini Stroud, and Joshua A Tucker. Like-minded sources on facebook are prevalent but not polarizing. *Nature*, 620(7972):137–144, August 2023.

Judea Pearl. *Causality: Models, Reasoning, and Inference*. Cambridge University Press, 2009.

Fábio Perez and Ian Ribeiro. Ignore previous prompt: Attack techniques for language models, 2022.

Geoff Pleiss, Manish Raghavan, Felix Wu, Jon M. Kleinberg, and Kilian Q. Weinberger. On fairness and calibration. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5680–5689, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/b8b9c74ac526fffbeb2d39ab038d1cd7-Abstract.html.

Inioluwa Deborah Raji. The Anatomy of AI Audits: Form, Process, and Consequences. In *The Oxford Handbook of AI Governance*. Oxford University Press, 2023. ISBN 9780197579329. doi: 10.1093/oxfordhb/9780197579329.013.28. URL https://doi.org/10.1093/oxfordhb/9780197579329.013.28.

Inioluwa Deborah Raji, Andrew Smart, Rebecca N. White, Margaret Mitchell, Timnit Gebru, Ben Hutchinson, Jamila Smith-Loud, Daniel Theron, and Parker Barnes. Closing the ai accountability gap: defining an end-to-end framework for internal algorithmic auditing. *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, 2020. URL https://api.semanticscholar.org/CorpusID:209862020.

Inioluwa Deborah Raji, Peggy Xu, Colleen Honigsberg, and Daniel E. Ho. Outsider oversight: Designing a third party audit ecosystem for ai governance, 2022.

Ashesh Rambachan, Jon M. Kleinberg, Sendhil Mullainathan, and Jens Ludwig. An economic approach to regulating algorithms. *NBER Working Paper Series*, 2020. URL https://api.semanticscholar.org/CorpusID:214775707.

Bashir Rastegarpanah, Krishna P. Gummadi, and Mark Crovella. Auditing black-box prediction models for data minimization compliance. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2021*, 2021.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should i trust you?": Explaining the predictions of any classifier, 2016.

Joseph P. Romano. On non-parametric testing, the uniform behaviour of the t-test, and related problems, 2004. URL https://www.jstor.org/stable/4616851.

Christian Sandvig, Kevin Hamilton, Karrie Karahalios, and Cédric Langbort. Auditing algorithms : Research methods for detecting discrimination on internet platforms. 2014. URL https://api.semanticscholar.org/CorpusID:15686114.

Latanya Sweeney. Discrimination in online ad delivery. http://ssrn.com/abstract=2208240, 2013.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks, 2013.

Eva Thelisson and Himanshu Verma. Conformity assessment under the EU AI act general approach. *AI Ethics*, January 2024.

Jinjin Tian and Aaditya Ramdas. Online control of the familywise error rate, 2019.

Florian Tramèr, Vaggelis Atlidakis, Roxana Geambasu, Daniel Hsu, Jean-Pierre Hubaux, Mathias Humbert, Ari Juels, and Huang Lin. Fairtest: Discovering unwarranted associations in data-driven applications, 2015.

Aleksandra Urman, Mykola Makhortykh, and Aniko Hannak. Mapping the field of algorithm auditing: A systematic literature review identifying research trends, linguistic and geographical disparities, 2024.

Claudia Wagner, David Garcia, Mohsen Jadidi, and Markus Strohmaier. It's a man's wikipedia? assessing gender inequality in an online encyclopedia. In *International AAAI Conference on Web and Social Media*, 2015.

Folkert Wilman. The digital services act (dsa) - an overview. 12 2022. URL https://ssrn.com/abstract=4304586.

Blake E. Woodworth, Suriya Gunasekar, Mesrob I. Ohannessian, and Nathan Srebro. Learning non-discriminatory predictors. *ArXiv*, abs/1702.06081, 2017. URL https://api.semanticscholar.org/CorpusID:2047106.

Lei Xu, Yangyi Chen, Ganqu Cui, Hongcheng Gao, and Zhiyuan Liu. Exploring the universal vulnerability of prompt-based learning paradigm, 2022.

Songkai Xue, Mikhail Yurochkin, and Yuekai Sun. Auditing ML models for individual bias and unfairness. In Silvia Chiappa and Roberto Calandra, editors, *The 23rd International Conference on Artificial Intelligence and Statistics, AISTATS 2020, 26-28 August 2020, Online [Palermo, Sicily, Italy]*, volume 108 of *Proceedings of Machine Learning Research*, pages 4552–4562. PMLR, 2020. URL http://proceedings.mlr.press/v108/xue20a.html.

Tom Yan and Chicheng Zhang. Active fairness auditing. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvári, Gang Niu, and Sivan Sabato, editors, *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pages 24929–24962. PMLR, 2022. URL https://proceedings.mlr.press/v162/yan22c.html.

Liuyi Yao, Zhixuan Chu, Sheng Li, Yaliang Li, Jing Gao, and Aidong Zhang. A survey on causal inference, 2020.

Karen Yeung. A study of the implications of advanced digital technologies (including ai systems) for the concept of responsibility within a human rights framework. *Social Science Research Network*, 2018. URL https://api.semanticscholar.org/CorpusID:158736157.

Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez-Rodriguez, and Krishna P. Gummadi. Fairness beyond disparate treatment & disparate impact: Learning classification without disparate mistreatment. *Proceedings of the 26th International Conference on World Wide Web*, 2016. URL https://api.semanticscholar.org/CorpusID:1911971.

Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks, 2013.

Jingwei Zhang, Tongliang Liu, and Dacheng Tao. An optimal transport view on generalization, 2018.

Kaiyang Zhou, Ziwei Liu, Yu Qiao, Tao Xiang, and Chen Change Loy. Domain generalization: A survey. 2021. doi: 10.1109/TPAMI.2022.3195549.

Appendix

A Additional related work

Algorithmic fairness. Our work is inspired by a large literature on algorithmic fairness. This methodological work is itself inspired by well-publicized instances of real-world algorithmic discrimination (e.g, Chouldechova (2016)). Of particular relevance to our work are the many definitions of fairness which have been proposed, including the notion of individual fairness (Dwork et al. 2011), equalized odds (Hardt et al. 2016), statistical parity or disparate impact (Calders et al. 2009, Kamiran and Calders 2009, Calders and Verwer 2010, Feldman et al. 2014, Edwards and Storkey 2015, Zafar et al. 2016, Johndrow and Lum 2017, Woodworth et al. 2017) and calibration (Crowson et al. 2016, com 2016), (Flores et al. 2016, Pleiss et al. 2017, Berk et al. 2017). Choosing a particular fairness measure is highly nontrivial, as imposing fairness constraints generally comes at some cost to model accuracy ((Corbett-Davies et al. 2017)). Furthermore, many seemingly natural definitions of fairness turn out to be incompatible with each other ((Kleinberg et al. 2016, Pleiss et al. 2017)). This motivates alternative approaches to fairness which do not directly alter model training procedures (Rambachan et al. 2020).

Our work is most closely related to a smaller but growing literature which develops tests for specific kinds of algorithmic harms or failures. For example, Black et al. (2019), Yan and Zhang (2022), Cherian and Candès (2023) develop tests for disparities in performance on important (and perhaps legally protected) subgroups, Xue et al. (2020) and Maity et al. (2021) propose algorithms to detect violations of individual fairness, Tramèr et al. (2015) and Adler et al. (2016) develop methods to understand how protected attributes influence model behavior (including indirectly). Alur et al. (2023, 2024) propose tests to detect whether algorithms fail to incorporate contextual information which may be available to a human decision maker, and Bartlett et al. (2019) propose a framework for detecting 'input' or proxy discrimination. For additional background we refer to Chouldechova and Roth (2018) and Mehrabi et al. (2019) for surveys of the literature.

Explainable machine learning. Our work is also related to a large and growing literature on *explainable* (or interpretable) machine learning. Although we cannot provide a complete overview here, notable works include *LIME* (Ribeiro et al. 2016), a technique for providing explanations for individual model predictions via black-box access, and *SHAP* (Lundberg and Lee 2017), a technique for attributing individual model predictions to specific inputs ('features'). Zeiler and Fergus (2013) propose a method for visualizing intermediate layers of a convolutional neural network. These works are broadly motivated by a desire to understand *why* and *how* machine learning models (particularly nonlinear models) make predictions. For additional background, including the challenges of defining model interpretability, we refer to Lipton (2016). For a survey and book-length treatment of specific techniques for model interpretability, we refer to Burkart and Huber (2020), Molnar (2022), respectively.

Adversarial attacks. Finally, our work on black-box auditing is complementary to a rich literature on adversarial machine learning, which seeks to discover (or mitigate against) adversarial inputs—often small perturbations of non-adversarial inputs—which 'fool' an algorithm into producing incorrect or incoherent outputs. Indeed, the robustness of algorithmic predictors to adversarial attacks is itself a natural property of interest for both internal and external auditors. Furthermore, the task of *generating* adversarial inputs using a sequence of black-box queries is very similar to

the problem of auditing for extreme values, and both are naturally addressed via the machinery of online convex optimization.

Work on adversarial attacks against machine learning models dates to early email spam filters (Dalvi et al. 2004, Lowd and Meek 2005a,b). Much of the more recent literature on the vulnerability of deep neural networks to adversarial attacks can be traced to Szegedy et al. (2013), who document the sensitivity of neural networks to imperceptible perturbations of their inputs. Notable work on adversarial attacks of deep neural networks includes Nguyen et al. (2014), Goodfellow et al. (2014), Kurakin et al. (2016), Biggio et al. (2017), Brendel et al. (2017), Ilyas et al. (2018). To address these vulnerabilities, Madry et al. (2017) propose an approach for training adversarially robust neural networks. More recently, Perez and Ribeiro (2022), Xu et al. (2022), Maus et al. (2023) propose techniques for generating adversarial *prompts* for modern foundation models. For additional background on adversarial machine learning, we refer to Biggio and Roli (2017).