Outcome Indistinguishability

Cynthia Dwork[*]
Harvard University
dwork@seas.harvard.edu

Michael P. Kim[†]
UC Berkeley
mpkim@berkeley.edu

Omer Reingold[‡]
Stanford Unviersity
reingold@stanford.edu

Guy N. Rothblum[§]
Weizmann Institute of Science
rothblum@alum.mit.edu

Gal Yona[¶]
Weizmann Institute of Science
gal.yona@weizmann.ac.il

Abstract

Prediction algorithms assign numbers to individuals that are popularly understood as individual "probabilities"—what is the probability of 5-year survival after cancer diagnosis?—and which increasingly form the basis for life-altering decisions. Drawing on an understanding of computational indistinguishability developed in complexity theory and cryptography, we introduce *Outcome Indistinguishability*. Predictors that are Outcome Indistinguishable yield a generative model for outcomes that cannot be efficiently refuted on the basis of the real-life observations produced by Nature.

We investigate a hierarchy of Outcome Indistinguishability definitions, whose stringency increases with the degree to which distinguishers may access the predictor in question. Our findings reveal that Outcome Indistinguishability behaves qualitatively differently than previously studied notions of indistinguishability. First, we provide constructions at all levels of the hierarchy. Then, leveraging recently-developed machinery for proving average-case fine-grained hardness, we obtain lower bounds on the complexity of the more stringent forms of Outcome Indistinguishability. This hardness result provides the first scientific grounds for the political argument that, when inspecting algorithmic risk prediction instruments, auditors should be granted oracle access to the algorithm, not simply historical predictions.

Contents

1 Introduction

Prediction algorithms "score" individuals, mapping them to numbers in $[0, 1]$ that are popularly understood as "probabilities" or "likelihoods:" the probability of 5-year survival, the chance that the loan will be repaid on schedule, the likelihood, that the student will graduate within for years, or that it will rain tomorrow. Algorithmic risk predictions increasingly inform consequential decisions, but what can these numbers really mean? Five-year survival, four-year graduation, rain tomorrow, are not repeatable events. The question of "individual probabilities" has been studied for decades across many disciplines without clear resolution.[1]

One interpretation relies on the coarseness of the representation of individuals to the prediction algorithm—the shape of a tumor's boundaries and the age of the patient; the student's grades, test scores, and a few bits about the family situation; the atmospheric pressure, humidity level, and winds—to partition individuals into a small number of "types." This leads to a natural interpretation of the predictions: *amongst the individuals of this type, what fraction exhibit a positive outcome?* In the context of modern data science, however, it is typical to make predictions based on a large number of expressive measurements for each individual—the patient's genome-wide risk factors; the borrower's online transactions and browsing data; the student's social media connections. In this case, when each individual resolves to a unique set of covariates, the frequency-based interpretation fails.

Another view imagines the existence of a party—Nature—who selects, for each individual, a probability distribution over outcomes; then, the realized outcome is determined by a draw from this distribution. Note that Nature may select the outcomes using complete determinism (i.e., probabilities in $\{0, 1\}$). This view of the world gives rise to a statistical model with well-defined individual probabilities, but reasoning about these probabilities from observational data presents challenges. Given only observations of outcomes, we cannot even determine whether Nature assigns integer or non-integer probabilities. Perhaps Nature is deterministic, but we do not have enough information or computing resources to carry out the predictions ourselves. Thus, even if we posit that outcomes are determined by individual probabilities, we cannot hope to recover the exact probabilities governing Nature, so this abstraction does not appear to provide an effective avenue for understanding the meanings of algorithmic risk scores.

1.1 Predictions that Withstand Observational Falsifiability

Given the philosophical uncertainty regarding the very existence of randomness, we explore the criteria for an ideal predictor. We can view the outputs of a prediction algorithm as defining a generative model for observational outcomes. Ideally, the the outcomes from this generative model should "look like" the outcomes produced by Nature. To this end, we introduce and study a strong notion of faithfulness—*Outcome Indistinguishability (OI)*. A predictor satisfying outcome indistinguishability provides a generative model that cannot be efficiently refuted on the basis of the real-life observations produced by Nature. In this sense, the probabilities defined by any OI

---

[1]See the inspiring paper, and references therein, of Philip Dawid [Daw15], discussing several notions of *individual risk* based on different philosophical understandings of probability "including Classical, Enumerative, Frequency, Formal, Metaphysical, Personal, Propensity, Chance and Logical conceptions of Probability" and proposing a new approach to characterizing individual risk which, the author concludes, remains elusive.

predictor provide a meaningful model of the "probabilities" assigned by Nature: even granted full access to the predictive model and historical outcomes from Nature, no analyst can invalidate the model's predictions. Our study contributes a computational-theoretic perspective on the deeper discussion of what we should demand of prediction algorithms–a subject of intense study in the statistics community for over 30 years (see, *e.g.*, the forecasting work in [Daw82,FV98,FL99,SSV03, San03])—and how they should be used. For example, one of our results provides scientific teeth to the political argument that, if risk prediction instruments are to be used by the courts (as they often are in the United States), then at the very least oracle access to the algorithms should be granted for auditing purposes.

Outcome Indistinguishability presents a broad framework evaluating algorithmic risk predictions. This paper focuses on the fundamental setting of predicting a binary outcome, given an individual's covariates, a simple prediction setup that already highlights many of the challenges and subtleties that arise while defining and reasoning about OI. Nothing precludes extending OI to reason about algorithms that make predictions about more general (*e.g.*, continuous) outcomes.

Basic notation. We assume that individuals are selected from some discrete domain $\mathcal{X}$, for example, the set of d-bit strings[2]. We model Nature as a joint distribution, denoted $\mathcal{D}^*$, over individuals and outcomes, where $o_i^* \in \{0, 1\}$ represents Nature's choice of outcome for individual $i \in X$. We use $i \sim \mathcal{D}_\mathcal{X}$ to denote a sample from Nature's marginal distribution over individuals and denote by $p_i^* \in [0, 1]$ the conditional probability that Nature assigns to the outcome o_i^*, conditioned on i. We emphasize, however, that Nature may choose $p_i^* \in \{0, 1\}$ to be deterministic; our definitions and constructions are agnostic as to this point.

A *predictor* is a function $p : \mathcal{X} \to [0, 1]$ that maps an individual $i \in \mathcal{X}$ to an estimate $\tilde{p}_i$ of the conditional probability of $o_i^* = 1$. For a predictor $\tilde{p} : \mathcal{X} \to [0, 1]$, we denote by $(i, \tilde{o}_i) \sim \mathcal{D}(\tilde{p})$ the individual-outcome pair, where $i \sim \mathcal{D}_\mathcal{X}$ is sampled from Nature's distribution over individuals, and then the outcome $\tilde{o}_i \sim \text{Ber}(\tilde{p}_i)$ is sampled from the Bernoulli distribution with parameter $\tilde{p}_i$.

Outcome Indistinguishability. Imagine that Nature selects $p_i^* = 1$ for half of the mass of $i \sim \mathcal{D}_\mathcal{X}$ and $p_i^* = 0$ for the remainder. If the two sets of individuals are easy to identify then we can potentially recover a close approximation to p^*. Suppose, however, that the sets are computationally indistinguishable, in the sense that given $i \sim \mathcal{D}_\mathcal{X}$, no efficient observer can guess if $p_i^* = 1$ or $p_i^* = 0$ with probability significantly better than $1/2$. In this case, producing the estimates $\tilde{p}_i = 1/2$ *for every individual* $i \in \mathcal{X}$ captures the best computationally feasible understanding of Nature: given limited computational power, the outcomes produced by Nature may faithfully be modeled as a random. In particular, if Nature were to change the outcome generation probabilities from p^* to $\tilde{p}$ we, as computationally bounded observers, will not notice. In other words, predictors satisfying OI give rise to models of Nature that cannot be falsified based only on observational data.

In the most basic form of the definition, a predictor $\tilde{p} : \mathcal{X} \to [0, 1]$ is Outcome Indistinguishable with respect to a family of distinguishers $\mathcal{A}$ if samples from Nature's distribution $(i, o_i^*) \sim \mathcal{D}^*$ cannot be distinguished by $\mathcal{A}$ from samples from the predictor's distribution $(i, \tilde{o}_i) \sim \mathcal{D}(\tilde{p})$, meaning that for

---

[2]Individuals can be arbitrarily complex; they are represented to the algorithm as elements of $\mathcal{X}$. Strictly speaking, distributions over $\mathcal{X}$ are induced distributions over the representations, and our results apply whether or not there are collisions. We do not assume that Nature's view is restricted to the representation.

each algorithm $A \in \mathcal{A}$, the probability that A outputs 1 is (nearly) the same on samples from each of the two distributions, $\mathcal{D}^*$ and $\mathcal{D}(\tilde{p})$.

Definition (Outcome Indistinguishability). *Fix Nature's distribution $\mathcal{D}^*$. For a class of distinguishers[3] $\mathcal{A}$ and $\varepsilon > 0$, a predictor $\tilde{p} : \mathcal{X} \to [0, 1]$ satisfies $(\mathcal{A}, \varepsilon)$-outcome indistinguishability (OI) if for every $A \in \mathcal{A}$,*

$$\left| \Pr_{(i, o_i^*) \sim \mathcal{D}^*} [\, A(i, o_i^*; \tilde{p}) = 1 \,] - \Pr_{(i, \tilde{o}_i) \sim \mathcal{D}(\tilde{p})} [\, A(i, \tilde{o}_i; \tilde{p}) = 1 \,] \right| \leq \varepsilon.$$

The definition of Outcome Indistinguishability can be extended in many ways, for example to settings where distinguishers receive multiple samples from each distribution, or when they have access to the program implementing $\tilde{p}$, and to the case of non-Boolean outcomes.

In the extreme, when we think of $\mathcal{A}$ as the set of all efficient computations, outcome indistinguishability sets a demanding standard for predictors that model Nature. Given an OI predictor $\tilde{p}$, even the most skeptical scientist—who, for example, does not believe that Nature can be captured by a simple computational model—cannot refute the model's predictions through observation alone. This framing seems to give an elegant computational perspective on the scientific method, when consider $\tilde{p}$ as expressing a hypothesis that cannot be falsified through observational investigation.

1.2 Our Contributions

The most significant contributions of this work can be summarized as follows:

(1) We define a practically-motivated four-level hierarchy of increasingly demanding notions of *Outcome Indistinguishability.* The levels of the hierarchy arise by varying the degree to which the distinguishers may access the predictive model in question.

(2) We provide tight connections between the two lower levels of the hierarchy to *multi-accuracy* and *multi-calibration*, two notions defined and studied in [HKRR18]. Establishing this connection immediately gives algorithmic constructions for these two levels.

(3) We describe a novel algorithm that constructs OI predictors directly. This construction establishes an upper bound on the complexity of OI for the upper levels of the hierarchy (and, consequently, also allows us to recover the results of [HKRR18] through the OI framework).

(4) We show a *lower bound* for the upper levels of the hierarchy, demonstrating the tightness of our constructions. We prove that, under plausible complexity-theoretic assumptions, at the top two levels of the hierarchy the complexity of the predictors cannot be polynomial in the complexity of the distinguishers in $\mathcal{A}$ and in the distinguishing advantage $1/\varepsilon$.

Additionally, we revisit the apparent interchangeability of the terms "test" and "distinguisher" in the literature on pseudorandomness, drawing a distinction that is relevant to the forecasting problem. Our reults clarify the mathematical relationship between notions in the two literatures.

---

[3]Like the predictors, distinguishers have access only to the finite *representations* of individuals as elements of $\mathcal{X}$.

Next, we present a colloquial illustration of the different notions of the hierarchy. While very natural, the notions within the hierarchy have not been fully considered in the literature on either forecasting or pseudorandomness.

The Outcome Indistinguishability Hierarchy. Imagine a medical board that wishes to audit the output of a program $\tilde{p}$ used to estimate the chances of five-year survival of patients under a given course of treatment. We can view the medical board as a distinguisher $A \in \mathcal{A}$. To perform the audit, the board receives historical files of patients and their five-year predicted (*i.e.*, drawn from $\mathcal{D}(\tilde{p})$) or actual (drawn from $\mathcal{D}^*$) outcomes. The requirement is that these two cases be indistinguishable to the board.

(1) To start, the board is only given samples, and must distinguish Nature's samples $(i, o_i^*) \sim \mathcal{D}^*$ from those sampled according to the predicted distribution $(i, \tilde{o}_i) \sim \mathcal{D}(\tilde{p})$. The board gets no direct access to predictions $\tilde{p}_i$ of the program; we call this variant *no-access-OI*.

(2) Naturally, the board may ask to see the predictions $\tilde{p}_i$ for each sampled individual. In this extension—*sample-access-OI*— the board must distinguish samples of the form $(i, o_i^*, \tilde{p}_i)$ and $(i, \tilde{o}_i, \tilde{p}_i)$, again for $(i, o_i^*) \sim \mathcal{D}^*$ and $(i, \tilde{o}_i) \sim \mathcal{D}(\tilde{p})$.

(3) *Oracle-access-OI* allows the board to make queries to the program $\tilde{p}$ on arbitrary individuals, perhaps to examine how the algorithm behaves on related (but unsampled) patients.

(4) Finally, in *code-access-OI*, the board is allowed to examine not only the predictions from $\tilde{p}$ but also the actual code, *i.e.*, the full implementation details of the program computing $\tilde{p}$.

On a different axis, we also consider multiple-sample variants of OI and show how these relate to the single-sample variants described above. Multiple-sample OI is closer to the problem of online *forecasting* (*e.g.*, daily weather forecasting); we explore connections between this variant of OI and the forecasting literature in Section 1.4.

The Lower Levels of the OI Hierarchy. We begin by examining the relationship between the different levels of the hierarchy. We show that no-access-OI and sample-access-OI are closely related to the notions of multi-accuracy and multi-calibration [HKRR18], respectively, studied in the algorithmic fairness literature. Very loosely, for a collection $\mathcal{C}$ of subpopulations of individuals, $(\mathcal{C}, \alpha)$-multi-calibration asks that a predictor $\tilde{p}$ be calibrated (up to α error[4]) not just overall, but also when we restrict our attention to subpopulations $S \subseteq \mathcal{X}$ for every set $S \in \mathcal{C}$. Here, calibration over S means that if we restrict our attention to individuals $i \in S$ for which $\tilde{p}_i = v$, then the fraction individuals with positive outcomes (i.e., $i \in S$ such that $o_i^* = 1$) is roughly v. We prove that sample-access-OI with respect to a set of distinguishers $\mathcal{A}$ is "equivalent" to $\mathcal{C}$-multi-calibration in the sense that each notion can enforce the other, for closely related classes $\mathcal{C}$ and $\mathcal{A}$.

Theorem 1 (Informal)**.** *For any class of distinguishers $\mathcal{A}$ and $\varepsilon > 0$, there exists a (closely related) collection of subpopulations $\mathcal{C}_\mathcal{A}$ and $\alpha_\varepsilon > 0$, such that $(\mathcal{C}_\mathcal{A}, \alpha_\varepsilon)$-multi-calibration implies*

---

[4]Defining approximate calibration is subtle; see a discussion in Section 4 with the formal definition of approximate multi-calibration.

$(\mathcal{A}, \varepsilon)$-*sample-access-OI. Similarly, for any collection of subpopulations $\mathcal{C}$ and $\alpha > 0$, there exists a (closely related) class of distinguishers $\mathcal{A}_{\mathcal{C}}$ and $\varepsilon_{\alpha} > 0$, such that $(\mathcal{A}_{\mathcal{C}}, \varepsilon_{\alpha})$-sample-access-OI implies $\mathcal{C}$-multi-calibration.*

Importantly, the relation between the class of distinguishers and collection of subpopulations preserves most natural measures of complexity; in other words, if we take $\mathcal{A}$ to be a class of efficient distinguishers, then evaluating set membership for the populations in $\mathcal{C}$ will be efficient (and vice versa). No-access-OI is similarly equivalent to the weaker notion of multi-accuracy, which requires accurate expectations for each $S \in \mathcal{C}$, rather than calibration.

Feasibility and Constructions. We consider the question of whether efficient OI predictors always exist. In particular, we ask, *Can we bound the complexity of OI predictors, independently of the complexity of Nature's distribution?* The picture we uncover is subtle; we will see that Outcome Indistinguishability differs qualitatively from prior notions of indistinguishability.

First off, leveraging feasibility results for the fairness notions from [HKRR18], we can obtain efficient predictors satisfying no-access-OI or sample-access-OI, by reduction to multi-accuracy and multi-calibration. Informally, for each of these levels, we can obtain OI predictors whose complexity scales linearly in the complexity of $\mathcal{A}$ and inverse polynomially in the desired distinguishing advantage ε. The result is quite generic; for concreteness, we state the theorem using circuit size as the complexity measure.

Theorem 2. *Let $\mathcal{A}$ be a class of distinguishers implemented by size-s circuits. For any $\mathcal{D}^*$ and $\varepsilon > 0$, there exists a predictor $\tilde{p} : \mathcal{X} \to [0, 1]$ satisfying $(\mathcal{A}, \varepsilon)$-sample-access-OI (similarly, no-access-OI) implemented by a circuit of size $O(s/\varepsilon^2)$.*

Turning now to oracle-access-OI and code-access-OI predictors, we obtain a general-purpose algorithm for constructing OI predictors, even when the distinguishers are allowed arbitrary access to the predictor in question. This construction extends the learning algorithm for multi-calibration of [HKRR18] to the more general setting of OI. When we allow such powerful distinguishers, the learned predictor $\tilde{p}$ is quantitatively less efficient than in the weaker notions of OI. In this introduction, we state the bound informally, assuming the distinguishers are implemented by circuits with oracle gates (a formal treatment appears in Section 5). As an example, if we let $\mathcal{A}$ be the set of oracle-circuits of some fixed polynomial size (in the dimension d of individual's representations), and allow arbitrary oracle queries, then $\tilde{p}$ will be of size $d^{O(1/\varepsilon^2)}$.

Theorem 3 (Informal)**.** *Let $\mathcal{A}$ be a class of oracle-circuit distinguishers implemented by size-s circuits that make at most q oracle calls to the predictor in question. For any $\mathcal{D}^*$ and $\varepsilon > 0$, there exists a predictor $\tilde{p} : \mathcal{X} \to [0, 1]$ satisfying $(\mathcal{A}, \varepsilon)$-oracle-access-OI implemented by a (non-oracle) circuit of size $s \cdot q^{O(1/\varepsilon^2)}$.*

Intuitively, code-access-OI can implement any of the prior levels through simulation: given the code for $\tilde{p}$, the distinguishers can execute oracle calls (or calls to $\tilde{p}_i$) whenever needed.[5] At the extreme of efficient OI, we consider code-access-OI with respect to the class of polynomial-sized

---

[5]There is some subtlety in making this intuition formal, related to encoding and decoding the description of the predictor, but the overhead is mild; see Section 5.

distinguishers. Importantly, we allow the complexity of these distinguishers to grow as a (fixed) polynomial in both the dimension of individuals d and the length of the description of the predictor $\tilde{p}$, which we denote by n. For this most general version of OI, the complexity may scale doubly exponentially in poly($1/\varepsilon$); nevertheless, the bound is independent of the complexity of p^*.

Theorem 4 (Informal). *For some $d \in \mathbb{N}$, let $\mathcal{X} \subseteq \{0,1\}^d$ be represented by d-bit strings. Suppose for some $k \in \mathbb{N}$, $\mathcal{A}$ is a class of distinguishers implemented by circuits of size $(d+n)^k$, on inputs $i \in \mathcal{X}$ and descriptions of predictors in $\{0,1\}^n$. For any $\mathcal{D}^*$ and $\varepsilon > 0$, there exists a predictor $\tilde{p} : \mathcal{X} \to [0,1]$ satisfying $(\mathcal{A}, \varepsilon)$-code-access-OI implemented by a circuit of size $d^{2^{O(1/\varepsilon^2)}}$.*

Hardness via Fine-Grained Complexity.   We establish a connection between the fine-grained complexity of well-studied problems and the complexity of achieving oracle-access-OI. Under the assumption that the (randomized) complexity of counting k-cliques in n-vertex graphs is $n^{\Omega(k)}$, we demonstrate that the construction of Theorem 3 is optimal up to polynomial factors. Specifically, we rule out (under this assumption) the possibility that the complexity of a oracle-access-OI predictor can be a fixed polynomial in the complexity of the distinguishers in $\mathcal{A}$ and in the distinguishing advantage ε. This hardness result holds for constant distinguishing advantage ε and for an efficiently-sampleable distribution $\mathcal{D}^*$. This hardness results are in stark contrast to the state of affairs for sample-access-OI (see Theorem 2). Concretely, in the parameters of the upper bound, the result based on the hardness of clique-counting rules out any predictor $\tilde{p}$ satisfying oracle-access-OI of (uniform) size significantly smaller than $d^{\Omega(1/\epsilon)}$.

Theorem 5 (Informal). *For $k \in \mathbb{N}$, assume there exist $\alpha > 0$ s.t. there is no $o(n^{\alpha \cdot k})$-time randomized algorithm for counting k-cliques. Then, there exist: $\mathcal{X} \subseteq \{0,1\}^{d^2}$, an efficiently-sampleable distribution $\mathcal{D}^*$, and a class $\mathcal{A}$ of distinguishers that run in time $\tilde{O}(d^3)$ and make $\tilde{O}(d)$ queries, s.t. for $\varepsilon = \frac{1}{100k}$, no predictor $\tilde{p}$ that runs in time $(d^{\alpha \cdot k} \cdot \log^{-\omega(1)}(d))$ can satisfy $(\mathcal{A}, \varepsilon)$-oracle-access-OI.*

See Corollary 6.9 for a formal statement. This lower bound is robust to the computational model: assuming that clique-counting requires $n^{\Omega(k)}$-sized circuits implies a similar lower bound on the circuit size of oracle-access-OI predictors. The complexity of clique counting has been widely studied and related to other problems in the fine-grained and parameterized complexity literatures, see the discussion in Section 6.2. We note that, under the plausible assumption that the fine-grained complexity of known clique counting algorithms is tight, our construction shows that obtaining oracle-access-OI is as hard, up to sub-polynomial factors, as computing p^*. We emphasize that this is the case even though the running time of the distinguishers can be arbitrarily small compared to the running time of p^*.

Hardness via BPP $\neq$ PSPACE.   We also show that, under the (milder) assumption that BPP $\neq$ PSPACE, there exists a polynomial collection of distinguishers and a distribution $\mathcal{D}^*$, for which no polynomial-time predictor $\tilde{p}$ can be OI. The distinction from the fine-grained result (beyond the difference in the assumptions) is that here $\mathcal{D}^*$ is not efficiently sampleable, and the distinguishing advantage for which OI is hard is much smaller.

Theorem 6 (Informal). *Assume that BPP $\neq$ PSPACE. Then, there exist: $\mathcal{X} \subseteq \{0,1\}^d$, a distribution $\mathcal{D}^*$ (which can be sampled in $\exp(\mathrm{poly}(d))$ time), and a class $\mathcal{A}$ of $\mathrm{poly}(d)$ distinguishers*

that run in time poly(*d*), *s.t. for* $\varepsilon = \frac{1}{\text{poly}(d)}$, *no predictor* $\tilde{p}$ *that runs in time* poly(*d*) *can satisfy* $(\mathcal{A}, \varepsilon)$-*oracle-access-OI.*

Discussion. We highlight a few possible interpretations and insights that stem from our technical results. The ability to construct predictors that satisfy outcome indistinguishability can be viewed both positively and negatively. On the one hand, the feasibility results demonstrate the possibility of learning generative models of observed phenomena that withstand very powerful scrutiny, even given the complete description of the model. On the other hand, OI does not guarantee statistical closeness to Nature (it need not be the case that $p^* \approx \tilde{p}$). Thus, the feasibility results demonstrate the ability to learn an *incorrect* model that cannot be detected by efficient inspection.

More generally, the computational perspective of OI underscores an inherent limitation in trying to recover the exact laws governing Nature from observational data alone. We illustrate this perspective through a comparison to pseudorandomness. Traditionally in pseudorandomness, our object of desire is *random* (e.g., a large string of random bits fed to a BPP algorithm), and we show that a simple *deterministic* object suffices to "fool" efficient observers. In outcome indistinguishability, our object of desire is a model of Nature, which may obey highly-complex *deterministic* laws. In this work, we show that a simple *random* model of Nature—namely, $\mathcal{D}(\tilde{p})$ for an OI predictor $\tilde{p}$—suffices to "fool" efficient observers. In this sense, attempting to recover the "true" model of Nature based on real-world observations is futile: no efficient analyst can falsify the outcomes of the random model defined by $\tilde{p}$, agnostic to the "true" laws of Nature.

The most surprising (and potentially-disturbing) aspect of our results may be the complexity of achieving oracle-access-OI and code-access-OI. In particular, for these levels, we show strong evidence that there exist p^* and $\mathcal{A}$ that do not admit efficient OI predictors $\tilde{p}$, *even when $\mathcal{A}$ is a class of efficient distinguishers!* That is, there are choices of Nature that cannot be modeled simply, even if all we care about is passing simple tests. This stands in stark contrast to the existing literature on indistinguishability, where the complexity of the indistinguishable object is usually polynomial in the distinguishers' complexity and distinguishing advantage, regardless of the complexity of the object we are trying to imitate.

The increased distinguishing power of oracle access to the predictor in oracle-access-OI seems to have practical implications. Currently, there are many conversations about the appropriate usage of algorithms when making high-stakes judgments about members of society, for instance in the context of the criminal justice system. Much of the discussion revolves around the idea of *auditing* the predictions, for accuracy and fairness. The separation between oracle-access-OI and sample-access-OI provides a rigorous foundation for the argument that auditors should at the very least have query access to the prediction algorithms they are auditing: given a fixed computational bound, the auditors with oracle-access may perform significantly stronger tests than those who only receive sample access.

1.3 Technical Overview

Next, we give a technical overview of the main results. Our goal is to convey the intuition for our findings, deferring the technical details to subsequent sections.

Relating OI and multi-calibration. To build intuition for the equivalence, as described informally in Theorem 1 and formally in Section 4, we begin by describing the construction that establishes the lower level of the equivalence, between multi-accuracy and no-access-OI (distinguishers that do not directly observe $\tilde{p}_i$). Informally, for a collection of subpopulations $\mathcal{C}$, multi-accuracy guarantees that the expectations of p_i^* and $\tilde{p}_i$ are approximately the same, even when conditioning on the event that $i \in S$ (simultaneously for every $S \in \mathcal{C}$).

Given a subpopulation S, we define the multi-accuracy violation $\nabla_S(\tilde{p})$ to be the absolute value of the above difference in conditional expectations. This can be viewed as a direct analogue of the distinguishing advantage $\Delta_A(\tilde{p})$ (the absolute difference between the acceptance probability of A on a sample from $\mathcal{D}^*$ vs $\mathcal{D}(\tilde{p})$). To translate between the notions, we define two mappings (subpopulations to distinguishers, and distinguishers to subpopulations) that allow us to upper-bound one quantity in terms of the other. Specifically, given a collection of subpopulations $\mathcal{C}$, we define a family of distinguishers $\mathcal{A} = \{A_S\}$, where for all $S \in \mathcal{C}$, $A_S(i, b) = \mathbf{1}[\, i \in S \wedge b = 1 \,]$.

Note that the probability that each A_S accepts on a sample $(i, \tilde{o}_i) \sim \mathcal{D}(\tilde{p})$ is the joint probability that $i \in S$ and $\tilde{o}_i = 1$, which can be directly related to the expectation of $\tilde{p}_i$ conditioned on $i \in S$. This allows us to express the multi-accuracy violation in terms of the distinguishing advantage, as $\nabla_S(\tilde{p}) = \frac{\Delta_{A_S}(\tilde{p})}{\mathbf{Pr}_{i \sim \mathcal{D}_\mathcal{X}}[i \in S]}$. By taking the accuracy parameter in no-access-OI to be sufficiently small, we can guarantee that no-access-OI implies multi-accuracy. Similarly, to implement outcome indistinguishability from multi-accuracy, we define two sets $S_{(A,0)}$ and $S_{(A,1)}$ for every distinguisher $A \in \mathcal{A}$ and $b \in \{0, 1\}$, $S_{(A,b)} = \{i \in \mathcal{X} : A(i, b) = 1\}$.

Using similar arguments, we show that $\Delta_A(\tilde{p})$ can be upper-bounded by $\nabla_{S_{(A,0)}}(\tilde{p}) + \nabla_{S_{(A,1)}}(\tilde{p})$, for which multi-accuracy (w.r.t the constructed family of subpopulations) provides a bound. Note that the constructions are very closely related; in fact, repeating the translation twice reaches a "stable point". That is, given $\mathcal{C}$ (or given $\mathcal{A}$), we can construct a canonical pair $(\mathcal{C}', \mathcal{A}')$ such that $\mathcal{C}'$-multi-accuracy implies $\mathcal{A}'$-no-access-OI, and vice versa. Importantly, $\mathcal{C}'$ is (essentially) of the same complexity as $\mathcal{C}$ (resp., $\mathcal{A}'$ compared to $\mathcal{A}$), and the degradation in the accuracy parameters only results from the fact that multi-accuracy is defined for a collection of arbitrarily small sets.

Showing a similar equivalence for multi-calibration follows the same general construction, but requires more care. We begin with the observation that for a fixed predictor $\tilde{p}$, $\mathcal{C}$-multi-calibration can be viewed as $\tilde{\mathcal{C}}$-multi-accuracy, where each subpopulation in $\tilde{\mathcal{C}}$ is obtained as the intersection of some subpopulation $S \in \mathcal{C}$ and "level-set" of $\tilde{p}$: $\{i \in S \wedge \tilde{p}_i = v\}$. Thus, at an intuitive level, we can model the constructions similarly in terms of the sets in $\tilde{\mathcal{C}}$. A number of technical subtleties arise due to the precise notion of approximate calibration from [HKRR18], which is necessary to provide sufficiently strong fairness guarantees; we discuss these issues in more detail in Section 4.

Constructing OI predictors. We establish the complexity of OI predictors (as in Theorems 3 and 4) by describing a learning algorithm that, given a class of distinguishers $\mathcal{A}$, an approximation parameter ε, and samples from Nature's distribution $(i, o_i^*) \sim \mathcal{D}^*$, constructs a predictor satisfying outcome indistinguishability, for any level of the OI hierarchy. To start, inspired by the approach of [HKRR18], we consider a reduction from the task of constructing an OI predictor to auditing for OI. Specifically, the auditing problem receives a candidate predictor p, and must determine whether for all $A \in \mathcal{A}$, the distinguishing advantage $\Delta_A(p) < \varepsilon$ is small; if there is an $A \in \mathcal{A}$ that has nontrivial advantage in distinguishing $\mathcal{D}^*$ from $\mathcal{D}(p)$, then the auditor must return such

8

a distinguisher. Naively, the auditor can be implemented by exhaustive search: for each $A \in \mathcal{A}$, the auditor—using the samples from $\mathcal{D}^*$ as well as generated samples from $\mathcal{D}(p)$—evaluates the advantage of $\Delta_A(p)$, returning A if $\Delta_A(p) > \varepsilon$.[6]

Suppose we're given some candidate predictor p; by iteratively auditing and updating, we aim to construct a circuit computing a predictor $\tilde{p}$ that satisfies OI. To start, given the predictor p, if the auditor certifies that p passes $(\mathcal{A}, \varepsilon)$-OI, then trivially we have succeeded in our construction. If, however, there is a distinguisher $A \in \mathcal{A}$ with a nontrivial advantage, then p fails the audit; in this case, the successful distinguisher $A \in \mathcal{A}$ witnesses some "direction" along which p fails to satisfy OI. If we can update along this direction, a standard potential argument (akin to that of boosting or gradient descent) demonstrates that the updated predictor has made "progress" towards satisfying OI. Then, we can recurse, calling the auditor on the updated predictor. We argue, the process must terminate with an OI predictor after not too many rounds of auditing and updating.

Thus, the crux of the construction is to solve the following problem: given a circuit computing a predictor p and a distinguisher $A \in \mathcal{A}$ that witnesses $\Delta_A(p) > \varepsilon$, derive a new circuit computing an updated predictor p' that has made progress towards OI. A subtle issue arises when making this intuition rigorous for oracle-access-OI and code-access-OI. At these levels of OI, the distinguishers may access the predictor in question, so there seems to be some circularity in the construction: to obtain the OI predictor $\tilde{p}$, we need to call the distinguishers $A \in \mathcal{A}$; but to evaluate the distinguishers $A \in \mathcal{A}$, we may need to access $\tilde{p}$. We argue that, in fact, there is no issue; to avoid the circularity, in each iteration, we can use the current predictor $p^{(t)}$ as the "oracle" for the distinguishers in $\mathcal{A}$. If $p^{(t)}$ passes auditing by oracle distinguishers $A^{p^{(t)}}$, then this predictor satisfies oracle-access-OI. If $p^{(t)}$ fails auditing, then we can still use the distinguisher $A^{p^{(t)}}$ to derive an update that we argue makes progress towards Nature's predictor p^*. Of course, because $\mathcal{D}^* = \mathcal{D}(p^*)$, p^* satisfies OI. Thus, the potential argument still works, and we guarantee termination after a bounded number of updates.[7]

To finish the construction, we leverage the concrete assumptions about the model of distinguishers to build up the circuit computing $\tilde{p}$. We focus on obtaining oracle-access-OI for size s oracle circuits that make at most q queries to $\tilde{p}$. The argument above ensures that in the t-th iteration, we can implement each oracle distinguisher using (non-oracle) circuits, where each of the q oracles calls is replaced with a copy of the current predictor $p^{(t)}$ hard-coded in place of the oracle gates. Then, we can derive an updated circuit $p^{(t+1)}$ by combining the circuits computing $p^{(t)}$ and computing $A^{p^{(t)}}$ (taking an addition of the outputs, with appropriate scaling). This recursive construction—where we build the circuit computing $p^{(t+1)}$ by incorporating multiple copies of $p^{(t)}$—suggests a recurrence relation characterizing an upper bound on the eventual circuit size. Intuitively, with a base circuit size of s, and q oracle calls (determining the branching factor), the size of $p^{(t)}$ grows roughly as $s \cdot q^t$. Leveraging an upper bound on the number of iterations $T = O(1/\varepsilon^2)$, the claimed bound follows.

Establishing the upper bound on the complexity of code-access-OI follows by a similar high-level argument, but there are some additional complications. Briefly, because the distinguishers take, as input, the description of a circuit computing the predictor in question, we need to work with

---

[6]For the sake of the overview of the construction, we ignore algorithmic issues of sample and time complexity in certifying that the estimate of each $\Delta_A(p)$ is sufficiently close.

[7]A similar argument holds for code-access-OI, using the description of $p^{(t)}$ as input.

a class of distinguishers that *grows with the complexity of the predictor itself.* We deal with the technicalities needed to encode and decode predictors so that we can simulate the lower levels within code-access-OI. We discuss these issues in full detail in Section 5.

Lower bounds for oracle-access-OI. To relate the complexity of evaluating oracle-access-OI predictors to complexity-theoretic assumptions such as the hardness of clique counting or PSPACE-complete problems (as done in the informal results stated in Theorems 5 and 6), we consider the task of constructing an OI predictor that needs to withstand the scrutiny of a distinguisher that can make oracle queries. Suppose we guarantee that any such predictor must compute a moderately hard function f on at least part of its domain. Then, a distinguisher could use oracle access to the predictor (on that part of the domain) to perform expensive computations of f at unit cost, while scrutinizing other parts of the domain. As we'll see, pushing this intuition, we show how efficient oracle distinguishers can perform surprisingly powerful tests to distinguish $\tilde{p}$ from p^*.

With this intuition in mind, we divide the domain $\mathcal{X}$ into m disjoint subsets $\mathcal{X}_1, \ldots, \mathcal{X}_m$. As a first step, we want to make sure that it is moderately hard to achieve OI when the distribution $\mathcal{D}^*$ is restricted to $\mathcal{X}_1$: for $i \in \mathcal{X}_1$, we set $o_i^* = f_1(i)$, where f_1 is a moderately hard Boolean function. We will guarantee that any oracle-access-OI predictor $\tilde{p}$ needs to compute f_1 on inputs in $\mathcal{X}_1$ by adding a distinguisher A_1 that verifies, for inputs $i \in \mathcal{X}_1$, that $o_i = f_1(i)$. At this point, it isn't clear that anything interesting is happening: the complexity of achieving OI (i.e., computing f_1) is not yet higher than the complexity of the distinguisher (which also needs to compute f_1). However, we can now add a distinguisher A_2 that verifies, for inputs in $\mathcal{X}_2$, that $\tilde{p}$ also correctly computes a harder function f_2. The key point is for the distinguisher to verify that $o_i = f_2(i)$ *without computing f_2 itself!* To achieve this, the distinguisher can use its oracle access to $\tilde{p}$. In particular, assuming that $\tilde{p}$ correctly computes f_1 on inputs $\mathcal{X}_1$, we can use a *downwards self-reduction* from computing f_2 on inputs in $\mathcal{X}_2$ to computing f_1 on inputs in $\mathcal{X}_1$.

The construction proceeds along these lines, using a sequence of functions $\{f_j\}_{j \in [m]}$, where for every $j \in [m]$ and $i \in \mathcal{X}_j$, we set $o_i^* = f_j(i)$. Now, for every $j \in [2, \ldots, m]$, we want the function f_j to be harder to compute than f_{j-1}, and we want a downwards self-reduction from computing f_j to computing f_{j-1}. The distinguisher A_j uses the given predictor $\tilde{p}$ as an oracle to f_{j-1}, and verifies that for $i \in \mathcal{X}_j$, $o_i = f_j(i)$. We emphasize that the complexity of the oracle distinguisher A_j is proportional to the cost of the downwards self-reduction, which can (and will) be significantly smaller than the complexity of computing f_j.

While intuitively appealing, the discussion above ignores an important point: OI only provides an approximate guarantee on the real-valued predictions, not exact recovery of the sequence of fucntions $\{f_j\}$. Starting at the first level of functions, an $(\{A_1\}, \varepsilon)$-oracle-access-OI predictor $\tilde{p}$ only has to correctly compute f_1 in a limited sense. First, $\tilde{p}$ only needs to be correct *on average* for random inputs; it can err completely on some inputs. Second, while we will choose f_1 (and all the f_j's) to be a Boolean function, the predictor $\tilde{p}$ itself need not be Boolean. Nonetheless, for any input i, the distinguisher A_1 accepts the input $(i, \tilde{o}_i)$ only when $\tilde{o}_i = f_1(i)$, so taking ε small enough guarantees that for any $(\{A_1\}, \varepsilon)$-oracle-access-OI predictor $\tilde{p}$, with all but small probability over a draw of i from $\mathcal{D}_\mathcal{X}$ restricted to $\mathcal{X}_1$, rounding $\tilde{p}(i)$ gives the correct value of f_1. The probability of an error raises a new problem: the distinguisher A_2, which uses oracle calls to $\tilde{p}$ to compute f_1, might receive incorrect answers! Indeed, we expect the downwards self-reduction from f_2 to

f_1 to make multiple queries (since f_2 is harder to compute), and so the error probability will be amplified. To handle this difficulty, we also want a *worst-case to average-case reduction* for f_1: from computing f_1 on worst-case inputs in $\mathcal{X}_1$, to computing f_1 w.h.p. over random inputs drawn from $\mathcal{D}_\mathcal{X}$ restricted to $\mathcal{X}_1$. Indeed, we'll want a similar reduction for each of the functions in the hierarchy. For each $j \in [2, \ldots, k]$, the distinguisher A_j will use the downwards self reduction, to f_{j-1}, using the worst-case to average-case reduction for f_{j-1} to reduce the error probability of $\tilde{p}$ before answering the downward self-reduction's oracle queries.

We can now make an inductive argument: assume that any $\tilde{p}$ that is OI for the distinguishers $\{A_1, \ldots, A_{j-1}\}$ must compute f_{j-1} correctly w.h.p. over inputs drawn from $\mathcal{D}_\mathcal{X}$ restricted to $\mathcal{X}_{j-1}$. Then $\tilde{p}$ is a very useful oracle for the j-th distinguisher A_j, which uses the downwards self-reduction and worst-case to average-case reductions, together with its oracle access to $\tilde{p}$, to compute f_j (and verify that $o_i = f_j(i)$). The key point is that A_j can do this (via oracle access to $\tilde{p}$), even though its running time is much smaller than the time needed to compute f_j. In turn, we conclude that any $\tilde{p}$ that is OI for the distinguishers $\{A_1, \ldots, A_j\}$ must compute f_j correctly w.h.p. over inputs drawn from $\mathcal{D}_\mathcal{X}$ restricted to $\mathcal{X}_j$. At the top (m-th) level of the induction, we conclude that a predictor that is OI for the entire collection of distinguishers must compute f_m correctly w.h.p. over random inputs. Finally, since we have a worst-case to average-case reduction for f_m, this implies that achieving OI is almost as hard as worst-case (randomized) computation of f_m.

To instantiate this framework, we need a collection of functions $\{f_j : \{0,1\}^n \to \{0,1\}\}_{j=1}^m$ with three properties: (1) "Scalable hardness": The complexity of computing f_j should increase with j. A natural goal is $n^{\Theta(j)}$ time complexity, where the lower bound should apply for randomized (BPP) algorithms; (2) Downwards self-reduction: A reduction from computing f_j to computing f_{j-1}, with fixed polynomial running time and query complexity (ideally $\tilde{O}(n)$, though we will use a collection where the complexity is a larger fixed polynomial); (3) Worst-case to average-case reduction: A reduction from computing f_j in the worst case, to computing f_j w.h.p. over a distribution D_j.

The clique counting problem presents a natural candidate, where f_{j-2} counts the number of j-cliques in an unweighted input graph with n vertices.[8] The complexity of this well-studied problem is believed to be $n^{\Theta(j)}$. Goldreich and Rothblum [GR18] recently showed a worst-case to average-case reduction for this problem, where the reduction runs in $\tilde{O}(n^2)$ times and makes $\text{poly}(\log n)$ queries. The problem also has a downwards self-reduction from counting cliques of size j to counting cliques of size $(j-1)$, which runs in time $O(n^3)$ and makes n oracle queries (on inputs of size $O(n^2)$). The above construction utilized a sequence of Boolean functions, whereas the output of the clique-counting function is an integer in $[n^j]$. We use the Goldreich-Levin hardcore predicate [GL89] to derive a Boolean function that is as hard to compute as clique counting. The full details are in Section 6.

The above framework can be also be instantiated using the algebraic variants of fine-grained complexity problems studied in the work of Ball, Rosen, Sabin, and Vasudevan [BRSV17, BRSV18]. Interestingly, downwards self-reducability also comes up in their work [BRSV18], where it is used to argue hardness for batch evaluation of many instances. Their algebraic variants of the k-orthogonal-vectors and k-SUM problems seem directly suited to our construction. We focus on clique counting because of the tightness of the upper and lower bounds that have been suggested and studied in the literature.

---

[8]Clique counting is trivial for cliques of size 1 or 2, and begins being interesting for 3-cliques, or triangle counting.

For PSPACE hardness of oracle-access-OI, we use a PSPACE-complete problem that is both downwards self-reducible and random self-reducible, due to Trevisan and Vadhan [TV07]. The details are in Section 6.3. The permanent [Lip89] (or scaled-down variants thereof, see [GR18]) seems to be another promising candidate for our construction. In a different direction, it is interesting to ask whether moderately hard cryptographic assumptions, as suggested by Dwork and Naor [DN92], could also provide candidates.

1.4 Broader Context and Related Notions

A Socio-Technical Path of Progress. A sufficiently rich representation of real-life individuals implies a mapping from individuals to their representation as input to the predictor that has no collisions. In other words, given enough bits of information in the representation, each individual will have a unique representation. Still, this richness does *not* mean that the representation contains the right information to determine the values of the p_i^*, even information-theoretically. For example, modulo identical twins, individuals' genomes suffice to uniquely represent every person, but sequences of DNA are insufficient to determine an individual's ability to repay a loan. Alternatively, the necessary information may be present, but its interpretation may be computationally infeasible.

Generally, we assume that the representation of individuals is fixed and informative. Our analysis demonstrates that OI is feasible by using a potential argument. Specifically, we describe an algorithm that iteratively looks for updates to the current set of predictions that will step closer towards indistinguishability. We guarantee that the algorithm terminates in a bounded number of steps by arguing that after sufficiently many updates, the predictor we hold is essentially p^*.

In practice, however, it may be the case that our features will be insufficiently rich to capture p^*. Given a simple representation, even if we require a predictor $\tilde{p}$ that satisfies OI using a very computationally-powerful set of distinguishers (*e.g.*, polynomial-sized circuits), there will be an inherent, information-theoretic limitation that prevents $\tilde{p}$ from converging to p^*. While, given this representation of individuals, it may be impossible to distinguish Nature's outputs o^* from $\tilde{o}$ drawn according to $\tilde{p}$, it may be possible to distinguish the outputs if we obtain an enriched representation of individuals. Moreover, obtaining an enriched representation may even be easy, in that it can be accomplished (by a human) in time polynomial in the size of the original representation!

The OI framework can be extended naturally, allowing for the representation of individuals to be augmented throughout time. Given such an enriched representation, we can continue iteratively updating $\tilde{p}$, based on the new representation. Specifically, we can obtain new training data, concatenate the old and enriched representations to form a new representation, initialize a new predictor to equal $\tilde{p}$, and enrich the collection of distinguishers to operate on the new representation. The new class of distinguishers retains and adds to the old distinguishing power, so $\tilde{p}$ likely will no longer satisfy OI; thus, we can apply our algorithm, starting at $\tilde{p}$, updating until the predictor fools the new class of distinguishers. By applying the same potential argument, we can can guarantee that this process of augmenting the representations cannot happen too many times. Any augmentation that significantly improves the distinguishing advantage between p^* and $\tilde{p}$ must result in new updates that allow for significant progress towards p^*.

Prediction Indistinguishability   In this work, we also investigate a notion we call *Prediction Indistinguishability (PI)*. In PI we require the stronger condition that (p^*, o^*) be indistinghishable from $(\tilde{p}, o^*)$. While Prediction Indistinguishability is intuitively appealing, there are very simple distinguishers that show it is too much to ask for: a predictor $\tilde{p}$ is PI with respect to these distinguishers if and only if $\tilde{p}$ is statistically close to p^*. But indistinguishability as a concept in computational complexity theory is interesting precisely when coming up with $\tilde{p}$ that is statistically close to p^* is impossible. Moreover, since we never see the values p^*—we don't even know if randomness exists!—we cannot hope to have indistinguishability of the $\tilde{p}$ from the "true" probabilities and it is strange even to pose such a criterion. Nonetheless, we show that PI and OI are equivalent when indistinguishability is with respect to tests that are passed by all natural histories with high probability (Section 3), which we discuss in more detail next.

Tests vs. Distinguishers   The framing of outcome indistinguishability draws directly from the notion of computational indistinguishability, studied extensively in the literature on cryptography, pseudorandomness, and complexity theory (see, e.g., [Gol06, Gol09, Vad12, Gol08] and references therein). A comparison to the extensive literature on online forecasting clarifies the semantic distinction between two concepts: *tests* (in the forecasting literature) and *distinguishers* (in the complexity literature).

The forecasting literature focuses on an online setting where there are two players, Nature and the Algorithm. Nature controls the data generating process (e.g., the weather patterns), while the Algorithm tries to assess, on each Day $t - 1$, the probability of an event on Day t (e.g., will it rain tomorrow?). There are many possibilities for Nature; by definition, in this literature, Nature calls the shots, in the following sense: On Day $t - 1$, Nature assigns a probability p_t^* that governs whether it will rain or not on Day t. Note that Nature is free to select $p_t^* \in \{0, 1\}$, in which case the outcomes are deterministic, $o_t^* = p_t^*$

In the early 1980s, [Daw82] proposed that, at the very least, forecasts should be calibrated. More stringent requirements were obtained by considering large (countable) numbers of sets of days, such as odd days, even days, prime-numbered days, days on which it has rained for exactly 40 preceding days and nights, and so on, and requiring calibration for each of these sets simultaneously [SSV03]. This is the "moral equivalent" of multi-calibration in the world of infinite sequences of online forecasting.

A signal result in the forecasting literature, due to Sandroni [San03] applies to a more general set of tests than calibration tests. Consider infinite histories, that is, sequences of (prediction, outcome) pairs. We say a history is *natural* if it is a sequence $((p_1, o_1), (p_2, o_2), \dots)$ where $\forall t$ we have $o_t \sim \text{Ber}(p_t)$. Note that certain natural histories may have no connection to any real-life weather phenomena, instead corresponding to a valid but unrealistic choice of p. A *test* takes as input a (not necessarily natural) history and outputs a bit. The test is usually thought of as trying to assess whether an algorithm's predictions are "reasonably accurate" with respect to the actual observations. This implicitly focuses attention on tests that natural histories pass with high probability (over the draws from the Bernoulli distributions), and indeed, calibration tests fall into this category. The goal of the Algorithm, then, is to generate predictions $\tilde{p}$ for which the histories $((\tilde{p}_1, o_1^*), (\tilde{p}_2, o_2^*), \dots)$ pass the test. Here $\tilde{p}_i$ is the Algorithm's forecasted probability of rain for Day i and o_i^* is the Boolean outcome, rain or not, that actually occured on Day i.

Sandroni's powerful result [San03], proves, non-constructively[9], the existence of an Algorithm that, given any test T, yields a history which passes T with probability at least as great as the minimum probability with which any natural history $((p_1, o \sim \text{Ber}(p_1)), (p_2, o \sim \text{Ber}(p_2)) \dots)$ passes T (again, the probability is over the draws from the $\text{Ber}(p_i)$, $i \geq 1$).

There are two major differences between tests in the forecasting literature and distinguishers in the complexity-theoretic literature.[10] The first is that tests have semantics—you want to pass the test, and the higher the probability of passing a test the better. In contrast, distinguishers output 0 or 1 with no semantics, and our goal is to produce an object such that the distinguisher outputs 1 with the same probability as the objects that we are imitating. In this case, getting the distinguisher to output 1 with higher probability may be worse. The second difference is that in the forecasting setting we want natural histories to pass the test with high probability: if natural histories fail the test, how can we interpret the Algorithm's inability to pass the test as an indication that the Algorithm is inaccurate? As a result, the Algorithm does not compete with the actual Nature p^*, but only with the hypothetical Nature that passes the test with the lowest probability.

To highlight these differences, consider a distinguisher that considers two sets of individuals, S and T. For each set, the distinguisher estimates the outcome probabilities α_S and α_T—that is, averaged over the individuals $i \in S$ (respectively, T), the probability that $o_i^* = 1$—and outputs 0 with probability $|\alpha_S - \alpha_T|$ and 1 otherwise. Note that for some Natures the distinguisher will output 1 with very small probability. Nevertheless, in cases where Nature treats S and T equally, the distinguisher will output 1 with high probability and, OI guarantees that $\tilde{p}$ must also treat S and T equally. Many properties of samples from a distribution are quite naturally and directly specified through the language of distinguishers, and not obviously through the language of tests. In light of this discussion, the connection between sample-access-OI and multi-calibration is very interesting: it shows how to reduce a collection of distinguishers into a collection of tests, and even more specifically to a collection of calibration tests.

Algorithmic fairness. Tests are also implicit in the literature on algorithmic fairness, where they are sometimes referred to as *auditors*. One line of work, the *evidence-based fairness* framework—initially studied in [HKRR18, KGZ19, DKR+19]—relates directly to outcome indistinguishability and centers around tests that Nature always passes. Broadly, the framework takes the perspective that, first and foremost, predictors should reflect the "evidence" at hand—typically specified through historical outcome data—as well as the statistical and computational resources allow.

Central to evidence-based fairness is the notion of multi-calibration [HKRR18], which was also studied in the context of rankings in [DKR+19]. Recently, [JLP+20] provide algorithms for achieving an extension of multi-calibration that ensures calibration of higher moments of a scoring function, and show how it can be used to provide credible prediction intervals. [SCM20] study multi-calibration from a sample-complexity perspective. In a similar vein, [ZME20] study a notion of individualized calibration and show it can be obtained by randomized forecasters.

Evidence-based fairness is part of a more general paradigm for defining fairness notions, sometimes referred to as "multi-group" notions, which has received considerable interest in recent

---

[9]The result leverages Fan's minimax theorem.

[10]Unfortunately, and confusingly, the literature on indistinguishability often conflates the notions, referring to distinguishers as tests.

years [HKRR18, KNRW18, KRR18, KGZ19, KNRW19, DKR$^+$19, SCM20, BL19, JLP$^+$20]. This approach to fairness aims to strengthen the guarantees of notoriously-weak group fairness notions, while maintaining their practical appeal. For instance, [KRR18, KNRW18, KNRW19] give notions of multi-group fairness based on parity notions studied in [DHP$^+$12] and [HPS16]. [BL19] extend this idea to the online setting. Other approaches to fairness adopt a different perspective, and intentionally audit for properties that Nature does not necessarily pass. Notable examples are group-based notions of parity [HPS16, KMR16, KNRW18, KNRW19].

Computational and Statistical Learning. Prediction tasks have also been studied extensively in the theoretical computer science and machine learning communities, both in the offline PAC model [Val84], as well as in the online model [LW94, FS97]; see [SSBD14] and references therein. Relatedly, [BM07] also show a multi-calibration-style guarantee in the online "sleeping experts" setting. More broadly, our work is also in conversation with more applied approaches to learning distributions and generative models including GANs [GPAM$^+$14] or auto-encoders [KW14]. The perspective of generating (statistically) indistinguishabile samples was also recently considered in a work introducing the problem of "sample amplification" [AGSV19].

Organization. The remainder of the manuscript is structured as follows.

- Section 2 defines OI formally and shows a number of propositions relating the various notions of OI to one another.

- Section 3 introduces the notion of prediction indistinguishability, and investigates the relationship of OI distinguishers to forecasting-style tests.

- Section 4 demonstrates the connections between the first two levels of the OI hierarchy to multi-accuracy and multi-calibration.

- Section 5 describes our construction of OI predictors, establishing the feasibility of the final two levels.

- Finally, Section 6 describes the construction establishing a conditional lower bound against the final levels.

2   Outcome Indistinguishability

Throughout this work, we study how to obtain predictors that generate outcomes that cannot be distinguished from natural outcomes. Specifically, we model Nature as a joint distribution over individuals and outcomes, denoted $\mathcal{D}^*$. Individuals come from a discrete domain $\mathcal{X}$; throughout, we will assume that each $i \in \mathcal{X}$ can be resolved to some d-dimensional boolean string $i \in \{0, 1\}^d$, which represents the "features" of the individual. In this work, we focus on boolean outcomes $\mathcal{Y} = \{0, 1\}$. Thus, $\mathcal{D}^*$ is supported on $\mathcal{X} \times \mathcal{Y} \subseteq \{0, 1\}^d \times \{0, 1\}$. We use $i \sim \mathcal{D}_{\mathcal{X}}$ to denote a sample from Nature's marginal distribution over individuals.

We say that a *predictor* is a function $p : \mathcal{X} \rightarrow [0, 1]$ that maps individuals to an estimate of the conditional probability of the individual's outcome being 1. For ease of notation, we use

$p_i = p(i)$ to denote a predictor's estimate for individual i. Note that the marginal distribution over individuals $\mathcal{D}_{\mathcal{X}}$ paired with a predictor induce a joint distribution over $\mathcal{X} \times \mathcal{Y}$. Given a predictor p, we use $(i, o_i) \sim \mathcal{D}(p)$ to denote an individual-outcome pair, where $i \sim \mathcal{D}_{\mathcal{X}}$ is sampled from Nature's distribution over individuals, and the outcome $o_i \sim \text{Ber}(p_i)$ is sampled—conditional on i—according to the Bernoulli distribution with parameter p_i.

With this basic setup in place, we are ready to introduce the main notion of this work—outcome indistinguishability (OI). Intuitively, when developing a prediction model, a natural goal would be to learn a predictor $\tilde{p} : \mathcal{X} \to [0, 1]$ whose outcomes "look like" Nature's distribution $\mathcal{D}^*$. Outcome indistinguishability formalizes this intuition, and is parameterized by a family of distinguisher algorithms $\mathcal{A}$. In the most basic form of OI, each $A \in \mathcal{A}$ receives as input a labeled sample from one of two distributions, Nature's distribution $\mathcal{D}^*$ or the predictor's distribution $\mathcal{D}(\tilde{p})$.

$$\underbrace{(i, o_i^*) \sim \mathcal{D}^*}_{\text{Nature's distribution}} \qquad\qquad \underbrace{(i, \tilde{o}_i) \sim \mathcal{D}(\tilde{p})}_{\text{Predictor's distribution}}$$

In other words, in each distribution the individual i is sampled according to nature's marginal distribution on inputs, denoted $i \sim \mathcal{D}_{\mathcal{X}}$. The distribution over outcomes, however, varies: conditioned on the individual i, the distinguisher either receives the corresponding natural outcome o_i^*, or receives an outcome sampled as $\tilde{o}_i \sim \text{Ber}(\tilde{p}_i)$. In its most basic form, a predictor $\tilde{p}$ satisfies OI over the family $\mathcal{A}$ if for all $A \in \mathcal{A}$, the probability that A accepts the sample (i, o_i) is (nearly) the same for Nature's distribution and the predictor's distribution. In addition to the sample from $\mathcal{D}^*$ versus $\mathcal{D}(\tilde{p})$, we can also allow the distinguishers to access the predictor $\tilde{p}$ itself. This setup allows us to define a prototype for a notion of OI.

Definition 2.1 (Outcome Indistinguishability). *Fix Nature's distribution $\mathcal{D}^*$. For a class of distinguishers $\mathcal{A}$ and $\varepsilon > 0$, a predictor $\tilde{p} : \mathcal{X} \to [0, 1]$ satisfies $(\mathcal{A}, \varepsilon)$-outcome indistinguishability (OI) if for every $A \in \mathcal{A}$,*

$$\left| \Pr_{(i, o_i^*) \sim \mathcal{D}^*} [\, A(i, o_i^*; \tilde{p}) = 1 \,] - \Pr_{(i, \tilde{o}_i) \sim \mathcal{D}(\tilde{p})} [\, A(i, \tilde{o}_i; \tilde{p}) = 1 \,] \right| \leq \varepsilon.$$

The subsequent sections introduce multiple variants of outcome indistinguishability, highlighting four distinct access patterns to $\tilde{p}$. By allowing the distinguishers increasingly liberal access to the predicitve model $\tilde{p}$, the indistinguishability guarantee becomes increasingly strong.

Remark on nature. We primarily model Nature $\mathcal{D}^*$ as a fixed and unknown joint distribution over $\mathcal{X} \times \mathcal{Y}$. The presentation of some result benefits from an equivalent view, based on the agnostic PAC framework [Hau92, KS94, KSS94]. In this view, we imagine that individuals $i \sim \mathcal{D}_{\mathcal{X}}$ are sampled from the marginal distribution over $\mathcal{X}$, and then Nature selects outcomes conditioned on i. Throughout, we will use $p^* : \mathcal{X} \to [0, 1]$ to denote the function that maps individuals to the true conditional probability of outcomes given the individual. That is, for all $i \in \mathcal{X}$:

$$p_i^* = \Pr_{(i, o_i^*) \sim \mathcal{D}^*} [\, o_i^* = 1 \mid i \,].$$

In our notation, we can imagine that Nature specifies the distribution over individuals $i \sim \mathcal{D}_{\mathcal{X}}$, then specifies the "natural predictor" p^* and samples the outcome for an individual i as $o_i^* \sim \text{Ber}(p_i^*)$;

in other words, $\mathcal{D}^* = \mathcal{D}(p^*)$. We emphasize that this view—of Nature selecting a predictor—is an abstraction that is sometimes instructive in our analysis of OI. Nevertheless, we make no assumptions about p^* other than it defines a valid conditional probability distribution for each $i \in \mathcal{X}$. In particular, p^* need not come from any realizable or learnable class of functions.

Expectations and norms. We often will take expectations over Nature's marginal distribution over individuals, possibly conditioned on membership in particular subpopulations $S \subseteq \mathcal{X}$. A simple but important observation is that for any subpopulation $S \subseteq \mathcal{X}$, the expectation of the natural outcome is equal to the expectation of p^*.

$$\Pr_{(i,o_i^*)\sim\mathcal{D}^*} [\ o_i^* = 1 \mid i \in S\ ] = \mathop{\mathbf{E}}_{(i,o_i^*)\sim\mathcal{D}^*} [\ o_i^* \mid i \in S\ ] = \mathop{\mathbf{E}}_{i\sim\mathcal{D}_\mathcal{X}} [\ p_i^* \mid i \in S\ ]$$

Similarly, we may compare predictors over the distribution of individuals. For any two predictors $p, p' : \mathcal{X} \to [0,1]$, we use the following ℓ_1-distance to measure the statistical distance between their outcome distributions $\mathcal{D}(p)$ and $\mathcal{D}(p')$.

$$\left\| p - p' \right\|_1 = \mathop{\mathbf{E}}_{i\sim\mathcal{D}_\mathcal{X}} [\ |p_i - p_i'|\ ]$$

We only use the $\|\cdot\|_1$ notation when the distribution on individuals $\mathcal{D}_\mathcal{X}$ is clear from context.

Supported predictions. In many definitions, we can reason about predictors as arbitrary functions $p : \mathcal{X} \to [0,1]$, but to be an effective definition, we need to discuss functions that are implemented by a realizable model of computation. Importantly, this means we will think of predictors as mapping individuals $i \in \mathcal{X}$ to a range of values p_i that live on a discrete subset of $[0,1]$. We assume for any predictor $p : \mathcal{X} \to [0,1]$, the predictor's support is a discrete set of values in $[0,1]$ that receive positive probability over $\mathcal{D}_\mathcal{X}$. For any subpopulation $S \subseteq \mathcal{X}$, we denote the support of p on S as

$$\text{supp}_S(p) = \left\{ v \in [0,1] : \Pr_{i\sim\mathcal{D}_\mathcal{X}} [\ p_i = v \mid i \in S\ ] > 0 \right\}$$

In this way, for any $v \in \text{supp}(p)$, the conditional distribution over individuals $i \sim \mathcal{D}_\mathcal{X}$ where we condition on the event $p_i = v$ is well-defined.

When possible, we obtain results agnostic to the exact choice of discretization. Sometimes, we need to reason about the discretization explicitly and map the values of $\text{supp}(p)$ onto a known grid with fixed precision; we introduce additional technical details as needed in the subsequent sections.

Distinguishers and subpopulations. The notion of outcome indistinguishability is parameterized by a family of distinguishing algorithms, which we denote as $\mathcal{A}$. To instantiate a concrete notion of OI (at any of the four levels we define), we must specify $\mathcal{A}$ within a fixed realizable model of computation. In practice, it may make sense to use a class of learning-theoretic distinguishers, (e.g., decision-trees, halfspaces, neural networks). In this work, we focus on more abstract models of distinguishers. When our proofs allow, we will treat $\mathcal{A}$ as an arbitrary class of computations, but for certain results, it will be easier to assume something about the model of computation in which $A \in \mathcal{A}$ are implemented (e.g., time-bounded uniform, size-bounded non-uniform).

Recall, we assume the domain of individuals $\mathcal{X} \subseteq \{0,1\}^d$ can be represented as d-dimensional boolean vectors for $d \in \mathbb{N}$, and that the distinguishing algorithms $A \in \mathcal{A}$ take as input an individual $i \in \mathcal{X}$ and an outcome $o_i \in \{0,1\}$. Often, we will think of the dimension d as fixed. In this case, we can think of $\mathcal{A}$ as a fixed class of distinguishers of concrete complexity: for example, if the class $\mathcal{A}$ is implemented by circuits, then we can reason about their size as $s(d) = s$ for some fixed $s \in \mathbb{N}$. When we think of the dimension as growing, then we need to consider ensembles of distinguishing families, where the family is parameterized by $d \in \mathbb{N}$.

The same issues arise when we discuss multi-calibration, which is parameterized by a collection of "efficiently-identifiable" subpopulations $\mathcal{C} \subseteq \{0,1\}^{\mathcal{X}}$. Here, efficiently-identifiable refers to the fact that we assume for each $S \in \mathcal{C}$, there exists some efficient computational model that given $i \in \mathcal{X}$, can compute the predicate $\mathbf{1}[\, i \in S \,]$ (i.e., the characteristic function of S). Again, whenever possible, our treatment does not depend on the exact model of computation.

Circuits. As suggested above, some results are most naturally stated with a concrete model of computation in mind. In these cases, we will describe the distinguishers and subpopulations as computed by a collection of circuits. Fixing such a model of circuits will be useful when relating the complexity of a class of distinguishers $\mathcal{A}$ to that of a class of subpopulations $\mathcal{C}$, as well as showing the feasibility of obtaining circuits that implement OI predictors. Analogous results could be proved instead for uniform classes.

Throughout, we say that a family of distinguishers $\mathcal{A}$ (resp., subpopulations $\mathcal{C}$) for $\mathcal{X} \subseteq \{0,1\}^d$ is implemented by a family of circuits of size $s(d)$, if for each $A \in \mathcal{A}$ (resp., $S \in \mathcal{C}$), there exists a bounded fan-in circuit over the complete boolean basis c_A that computes the distinguisher A on all inputs, with at most $s(d)$ gates (or equivalently, by bounded fan-in, $\Theta(s(d))$ wires).

Specifying the model of computation for the most stringent levels of OI requires some care. The third level—oracle-access-OI—allows the distinguishers oracle-access to the predictor in question. For each $A \in \mathcal{A}$, we denote the oracle distinguisher as $A^{\tilde{p}}$, which has random access to $\tilde{p}_i$ for any $i \in \mathcal{X}$. The fourth level—code-access-OI—allows the distinguishers direct access to the description of the predictor in question, denoted $\langle \tilde{p} \rangle$. In this case, it makes sense to allow the ensemble of distinguishers to be parameterized by the length of the description $n = |\langle \tilde{p} \rangle|$ in addition to the dimension d. We discuss the specific assumptions about the implementation of these notions in subsequent sections.

2.1 Defining the Levels of Outcome Indistinguishability

With the general framework and preliminaries in place, we are ready to define the various levels of outcome indistinguishability. In this section, we focus on the definitions of each notion—no-access-OI, sample-access-OI, oracle-access-OI, and code-access-OI. Along the way, we show some relations between the notions, but defer most of our investigation of the notions to subsequent sections. We begin introducing each notion in the single-sample setting, and discuss OI for distinguishers that receive multiple samples in Section 2.2.

2.1.1 No-Access-OI

The weakest model of distinguisher receives no direct access to the predictive model $\tilde{p}$, and must make its judgments based only on the observed sample. In this framework, the only access to the predictor is indirect, through the sampled outcomes.

Definition 2.2 (No-Access-OI). *Fix Nature's distribution $\mathcal{D}^*$. Let $\mathcal{Z} = \mathcal{X} \times \{0, 1\}$. For a class of distinguishers $\mathcal{A} \subseteq \{\mathcal{Z} \to \{0, 1\}\}$ and $\varepsilon > 0$, a predictor $\tilde{p} : \mathcal{X} \to [0, 1]$ is $(\mathcal{A}, \varepsilon)$-no-access-OI if for every $A \in \mathcal{A}$,*

$$\left| \Pr_{(i, o_i^*) \sim \mathcal{D}^*} [\ A(i, o_i^*) = 1\ ] - \Pr_{(i, \tilde{o}_i) \sim \mathcal{D}(\tilde{p})} [\ A(i, \tilde{o}_i) = 1\ ] \right| \leq \varepsilon.$$

We remark that from a statistical perspective, no-access-OI already defines a strong framework for indistinguishability. Even at this baseline level, if we allow a computationally-inefficient class of distinguishers, no-access-OI can be used to require closeness in statistical distance. For instance, consider a family of "subset" distinguishers, where for a subset $S \subseteq \mathcal{X}$, the distinguisher A_S is defined as follows.

$$A_S(i, o_i) = \begin{cases} o_i & \text{if } i \in S \\ 0 & \text{o.w.} \end{cases}$$

If we take $\mathcal{A} = \{A_S : S \subseteq \mathcal{X}\}$ to be the family of all subset distinguishers, then the only predictors $\tilde{p}$ that satisfy no-access-OI will be statistically close to Nature's predictor p^*. Of course, this class of distinguishers includes inefficient tests (necessary to certify $\|p^* - \tilde{p}\|_1$ is small). Our interest will be on the guarantees afforded by OI when we take $\mathcal{A}$ to be a class of efficient distinguishers.

2.1.2 Sample-Access-OI

To strengthen the distinguishing power, we define sample-access-OI, which allows distinguishers to observe the prediction for the individual in question. Specifically, in addition to the sampled individual $i \sim \mathcal{D}$ and outcome o_i (drawn according to nature or the predictor $\tilde{p}$), the distinguisher receives the prediction $\tilde{p}_i$.

Definition 2.3 (Sample-Access-OI). *Fix Nature's distribution $\mathcal{D}^*$. Let $\mathcal{Z} = \mathcal{X} \times \{0, 1\} \times [0, 1]$. For a class of distinguishers $\mathcal{A} \subseteq \{\mathcal{Z} \to \{0, 1\}\}$ and $\varepsilon > 0$, a predictor $\tilde{p} : \mathcal{X} \to [0, 1]$ is $(\mathcal{A}, \varepsilon)$-sample-access-OI if for every $A \in \mathcal{A}$,*

$$\left| \Pr_{(i, o_i^*) \sim \mathcal{D}^*} [\ A(i, o_i^*, \tilde{p}_i) = 1\ ] - \Pr_{(i, \tilde{o}_i) \sim \mathcal{D}(\tilde{p})} [\ A(i, \tilde{o}_i, \tilde{p}_i) = 1\ ] \right| \leq \varepsilon.$$

Sample-access-OI is a strengthening of no-access-OI: for any no-access-OI distinguisher, on input $(i, o_i, \tilde{p}_i)$, a sample-access-OI distinguisher can simply ignore the prediction $\tilde{p}_i$, and simulate the original no-access-OI distinguisher.

2.1.3 Oracle-Access-OI

The next strengthening of OI allows distinguishers to make queries to $\tilde{p}$, not just on the sampled individual $i \sim \mathcal{D}$, but also on any other $j \in \mathcal{X}$. Such a query model needs to be formalized; at a high level, we assume the distinguishers in the class $A \in \mathcal{A}$ are augmented with oracle access to $\tilde{p}$, denoted as $A^{\tilde{p}}$.

Definition 2.4 (Oracle-Access-OI). *Fix Nature's distribution $\mathcal{D}^*$. Let $\mathcal{Z} = \mathcal{X} \times \{0,1\}$. For a class of oracle distinguishers $\mathcal{A} \subseteq \{\mathcal{Z} \to \{0,1\}\}$ and $\varepsilon > 0$, a predictor $\tilde{p} : \mathcal{X} \to [0,1]$ is $(\mathcal{A}, \varepsilon)$-oracle-access-OI if for every $A^{\tilde{p}} \in \mathcal{A}$,*

$$\left| \Pr_{(i,o_i^*) \sim \mathcal{D}^*} \left[A^{\tilde{p}}(i, o_i^*) = 1 \right] - \Pr_{(i,\tilde{o}_i) \sim \mathcal{D}(\tilde{p})} \left[A^{\tilde{p}}(i, \tilde{o}_i) = 1 \right] \right| \leq \varepsilon.$$

The exact formulation of such oracle distinguishers will vary based on the model of computation in which $\mathcal{A}$ is defined. In Section 5, we give one concrete formalization for distinguishers that are implemented efficiently as boolean circuits. Independent of the exact model, oracle-access-OI can implement sample-access-OI: on input (i, o_i), the oracle-access-OI distinguisher can access $\tilde{p}_i$ using a single query and then simulate the sample-access-OI distinguisher.

Lunchtime-OI and Sample-Access-OI. Oracle-access-OI generally defines a stronger notion of indistinguishability than sample-access-OI, but we show that if the oracle-access-OI distinguishers are non-adaptive—asking only pre-processing queries—then they can be simulated by a family of (non-uniform) sample-access-OI distinguishers. This result demonstrates that the power of oracle-access distinguishers over sample-access distinguishers derives from the ability to query $\tilde{p}$ *adaptively*, based on the sample in question. In particular, in Section 6 we show that oracle-access-OI is strictly more powerful than sample-access-OI. The construction follows by exploiting correlations within $\mathcal{D}^*$ across different $i, j \in \mathcal{X}$, which can be tested efficiently by querying $\tilde{p}$ adaptively.

In fact, this collapse from oracle-access-OI to sample-access-OI will hold for an even more powerful class of distinguishers, which are allowed "lunchtime attack" style pre-processing on $\tilde{p}$. Consider the following model of pre-processing analysis. For some $t \in \mathbb{N}$, given a family of distinguishers $\mathcal{A}$, suppose that for each $A \in \mathcal{A}$, there exists a pre-processing algorithm $R_A^{\tilde{p}} : 1^d \to \{0,1\}^t$ with oracle access to $\tilde{p}$. Given access to $\tilde{p}$ for input domain $\mathcal{X} \subseteq \{0,1\}^d$, the pre-processing algorithm $R_A^{\tilde{p}}(1^d)$ produces an advice string $a \in \{0,1\}^t$. Then, oracle access to $\tilde{p}$ is revoked, and the distinguisher A receives a individual-outcome-prediction sample $(i, o_i, \tilde{p}_i)$ from one of the two distributions, given access to a.[11] That is, the lunchtime variant of $(\mathcal{A}, \varepsilon)$-oracle-access-OI holds if for every $A \in \mathcal{A}$ and $a = R_A^{\tilde{p}}(1^d)$,

$$\left| \Pr_{(i,o_i^*) \sim \mathcal{D}^*} \left[A^a(i, o_i^*, \tilde{p}_i) = 1 \right] - \Pr_{(i,\tilde{o}_i) \sim \mathcal{D}(\tilde{p})} \left[A^a(i, \tilde{o}_i, \tilde{p}_i) = 1 \right] \right| \leq \varepsilon.$$

Note that computing $R_A^{\tilde{p}}(1^d)$ need not be efficient, but importantly, its analysis of $\tilde{p}$ must be summarized into t bits. For this variant of oracle-access-OI, we show the following inclusion.

---

[11]Note that, as in sample-access-OI, we additionally give $\tilde{p}_i$ as input to the lunchtime distinguisher. We exclude the prediction $\tilde{p}_i$ as input in Definition 2.4 because, in general, an adaptive oracle-access-OI distinguisher can query $\tilde{p}_i$ as desired. Without feeding the prediction as input, lunchtime-OI actually collapses to no-access-OI.

Proposition 2.5. *Fix Nature's distribution $\mathcal{D}^*$. Let $\mathcal{Z} = \mathcal{X} \times \{0,1\} \times [0,1]$. Suppose $\mathcal{A} \subseteq \{\mathcal{Z} \to \{0,1\}\}$ is a class of lunchtime distinguishers implemented by size-s circuits. Then, there exists a class of sample-access distinguishers $\mathcal{A}'$ implemented by size-s circuits, such that any predictor $\tilde{p} : \mathcal{X} \to [0,1]$ that satisfies $(\mathcal{A}', \varepsilon)$-sample-access-OI must also satisfy $(\mathcal{A}, \varepsilon)$-oracle-access-OI.*

Proof. Given a class of lunchtime distinguishers $\mathcal{A}$, we define a new class $\mathcal{A}'$ of sample-access distinguishers as follows.

$$\mathcal{A}' = \left\{ A'_a : A \in \mathcal{A}, \ a \in \{0,1\}^t \right\}$$

where A'_a is defined as

$$A'_a(i, o_i, p_i) = A^a(i, o_i, p_i)$$

for all $i \in \mathcal{X}$ and $o_i \in \{0,1\}$ and $p_i \in [0,1]$. In other words, for each $A \in \mathcal{A}$, we introduce 2^t fixed distinguishers that have the possible output of $R^{\tilde{p}}_A(1^d)$ hard-coded. If A is implemented by a circuit with access to the advice string output by $R^{\tilde{p}}_A(1^d)$, then for any $a \in \{0,1\}^t$, A'_a can be implemented by circuits with the same number of wires. We argue that $(\mathcal{A}', \varepsilon)$-sample-access-OI implies $(\mathcal{A}, \varepsilon)$-oracle-access-OI.

Suppose there is some $A \in \mathcal{A}$ such that A^a distinguishes between the natural and modeled distribution. Then, by construction, there exists some $A'_a \in \mathcal{A}'$ that also distinguishes the distributions with the same advantage. Thus, by contrapositive, if a predictor $\tilde{p}$ satisfies $(\mathcal{A}', \varepsilon)$-sample-access-OI, then it also satisfies $(\mathcal{A}, \varepsilon)$-oracle-access-OI. $\qquad\square$

Note that we state and prove Proposition 2.5 for distinguishers implemented by circuits, but the construction is quite generic. This style of hard-coding works very naturally for any non-uniform class model of distinguishers. Even if we work with a uniform model of distinguishers, if the length of the advice string $t \in \mathbb{N}$ is a constant (independent of d the dimension of individuals $\mathcal{X}$), then for each $A \in \mathcal{A}$ we can define a TM that has $a \in \{0,1\}^t$ hard-coded as part of its description. The number of distinguishers in $\mathcal{A}'$ grows by a factor of 2^t.

2.1.4   Code-Access-OI

The strongest notion of distinguishers we consider receive—as part of their input—the description $\langle \tilde{p} \rangle$ of a circuit that computes $\tilde{p}$. In this model, which we call code-access-OI, the distinguishers can accept or reject their sample based on nontrivial analysis of the circuit computing $\tilde{p}$, not just its evaluation on domain elements. We assume that $|\langle \tilde{p} \rangle| = n$ for some $n \in \mathbb{N}$.

Definition 2.6 (Code-Access-OI). *Fix Nature's distribution $\mathcal{D}^*$. Let $Z = \mathcal{X} \times \{0,1\} \times \{0,1\}^n$ for $n \in \mathbb{N}$. For a class of distinguishers $\mathcal{A} \subseteq \{\mathcal{Z} \to \{0,1\}\}$ and $\varepsilon > 0$, a predictor $\tilde{p} : \mathcal{X} \to [0,1]$ is $(\mathcal{A}, \varepsilon)$-code-access-OI if for every $A \in \mathcal{A}$,*

$$\left| \Pr_{(i,o_i^*)\sim\mathcal{D}^*} [\ A(i, o_i^*, \langle \tilde{p} \rangle) = 1\ ] - \Pr_{(i,\tilde{o}_i)\sim\mathcal{D}(\tilde{p})} [\ A(i, \tilde{o}_i, \langle \tilde{p} \rangle) = 1\ ] \right| \le \varepsilon.$$

There are a number of subtle technicalities in how we define code-access-OI, relating to how we encode $\langle \tilde{p} \rangle$. In particular, if we want to be able to simulate the prior notions of OI within code-access-OI, then we need to allow the complexity of the distinguishers in $\mathcal{A}$ to scale with the complexity of $\tilde{p}$. Even evaluating $\tilde{p}$ on a single domain element requires that $\mathcal{A}$ can compute circuit evaluation. This technicality sets code-access-OI apart from the prior notions, where it sufficed to think of the domain as fixed in dimension, and thus think of the distinguishers' complexity as fixed as well. We discuss these issues in detail in Section 5, where we discuss the feasibility of OI.

2.2 Multiple sample OI

Throughout this work, we focus on distinguishers that receive a single sample from nature or the modeled distribution, with varying levels of access to $\tilde{p}$. A natural generalization of this model allows distinguishers to access multiple samples. We define the generic variant as follows (where each of no-access-OI, sample-access-OI, oracle-access-OI, and code-access-OI follow by allowing distinguishers the analogous degree of access to $\tilde{p}$).

Definition 2.7. *Fix Nature's distribution $\mathcal{D}^*$. Let $m \in \mathbb{N}$ and $\mathcal{Z} = (\mathcal{X} \times \{0,1\})$. For a class of multi-sample distinguishers $\mathcal{A}_m \subseteq \{\mathcal{Z}^m \to \{0,1\}\}$ and $\varepsilon > 0$, a predictor $\tilde{p} : \mathcal{X} \to [0,1]$ is $(\mathcal{A}_m, \varepsilon)$-OI if for every $A_m \in \mathcal{A}_m$,*

$$
\left| \Pr_{(i_1, o_{i_1}^*), \ldots, (i_m, o_{i_m}^*) \sim (\mathcal{D}^*)^m} \left[A_m \left((i_1, o_{i_1}^*), \ldots, (i_m, o_{i_m}^*); \tilde{p} \right) = 1 \right] \right.
$$

$$
\left. - \Pr_{(i_1, \tilde{o}_{i_1}), \ldots, (i_m, \tilde{o}_{i_m}) \sim \mathcal{D}(\tilde{p})^m} \left[A_m \left((i_1, \tilde{o}_{i_1}), \ldots, (i_m, \tilde{o}_{i_m}); \tilde{p} \right) = 1 \right] \right| \leq \varepsilon.
$$

We leave full exploration of multi-sample-OI to future work, but make the following observation. If the class of distinguishers we use admits a hybrid argument, then the multi-sample distinguishers' advantage can be bounded generically in terms of the single-sample advantage. As an example, we show the following proposition for oracle-access-OI.

Proposition 2.8. *Fix Nature's distribution $\mathcal{D}^*$. Let $\mathcal{A}$ be the class of size-s single-sample distinguishers, and for $m \in \mathbb{N}$ let $\mathcal{A}_m$ be the class of size-s m-sample distinguishers. Suppose we allow $\mathcal{A}$ pre-processing samples from $\mathcal{D}^*$ and oracle-access to $\tilde{p}$. For $\varepsilon > 0$, if a predictor $\tilde{p} : \mathcal{X} \to [0,1]$ is $(\mathcal{A}, \varepsilon/m)$-oracle-access-OI, then it is $(\mathcal{A}_m, \varepsilon)$-oracle-access-OI.*

Proof. Suppose there exists some m-sample distinguisher $A_m \in \mathcal{A}_m$ that distinguishes between nature and the model $\tilde{p}$ with advantage at least ε. We show that there is a single-sample randomized distinguisher $A \in \mathcal{A}$ that distinguishes between nature and the model with advantage at least ε/m. By contrapositive, if $\tilde{p}$ is $(\mathcal{A}, \varepsilon/m)$-oracle-access-OI, then it must be $(\mathcal{A}_m, \varepsilon)$-oracle-access-OI.

Consider the following sequence of hybrid distributions over m samples, $(i_1, o_{i_1}), \ldots, (i_m, o_{i_m})$, where $\mathcal{D}_k = (\mathcal{D}^*)^{m-k} \times \mathcal{D}(\tilde{p})^k$ is a product distribution of $m - k$ independent samples from nature and k samples from the model. Note that assuming pre-processing access to samples from $\mathcal{D}^*$ and oracle access to $\tilde{p}$, each $\mathcal{D}_k$ is sampleable. Specifically, to obtain a sample from $\mathcal{D}_k$, we will draw m samples from $\mathcal{D}^*$, and then for each $j \in \{m - k + 1, \ldots, m\}$, we resample the outcome by evaluating $\tilde{p}_{i_j}$ and then randomly drawing $\tilde{o}_{i_j} \sim \mathrm{Ber}(\tilde{p}_{i_j})$.

Observing that $\mathcal{D}_0 = (\mathcal{D}^*)^m$ and $\mathcal{D}_m = \mathcal{D}(\tilde{p})^m$, we can write the distinguishing probability of A_m as a telescoping sum over distinguishing probabilities over the hybrid distributions.

$$\Pr_{(i_1, o_{i_1}^*),\ldots,(i_m, o_{i_m}^*) \sim (\mathcal{D}^*)^m} \left[A_m^{\tilde{p}} \left((i_1, o_{i_1}^*), \ldots, (i_m, o_{i_m}^*) \right) = 1 \right]$$

$$- \Pr_{(i_1, \tilde{o}_{i_1}),\ldots,(i_m, \tilde{o}_{i_m}) \sim \mathcal{D}(\tilde{p})^m} \left[A_m^{\tilde{p}} \left((i_1, \tilde{o}_{i_1}), \ldots, (i_m, \tilde{o}_{i_m}) \right) = 1 \right]$$

$$= \sum_{j=1}^{m} \left(\Pr_{(I,O) \sim \mathcal{D}_{j-1}} \left[A_m^{\tilde{p}}(I, O) = 1 \right] - \Pr_{(I,O) \sim \mathcal{D}_j} \left[A_m^{\tilde{p}}(I, O) = 1 \right] \right)$$

$$\geq \varepsilon$$

Thus, the following randomized single-sample oracle-access-OI distinguisher succeeds with advantage at least ε/m: as pre-processing, sample a random index $j \sim [m]$ and draw a sample from the hybrid distribution $\mathcal{D}_j$; on input (i, o_i), replace the jth sample with the input (i, o_i), and run A_m on the resulting m-sample input. If the input is drawn from nature, then the resulting sample is drawn from $\mathcal{D}_{j-1}$, whereas if the input is from the model, then the resulting sample is drawn from $\mathcal{D}_j$. Thus, the distinguishing advantage of A is the average distinguishing advantage between $\mathcal{D}_{j-1}$ and $\mathcal{D}_j$, or ε/m. $\qquad\square$

We state Proposition 2.8 for oracle-access-OI (and thus, by simulation, code-access-OI), due to the ease of running the hybrid argument with oracle access to $\tilde{p}$. Note that we use circuit-size as the complexity measure for concreteness, but the argument will go through for most complexity measures of $\mathcal{A}_m$. Similar hybrid arguments can also be made for no-access-OI and sample-access-OI, provided the model of computation of the distinguishers admits "hard-coding" the outcome values $\{\tilde{o}_{i_{m-k+1}}, \ldots, \tilde{o}_{i_m}\}$, and $\{\tilde{p}_{i_1}, \ldots, \tilde{p}_{i_{m-k}}\}$ if needed (for sample-access-OI). In particular, for any non-uniform class of multi-sample distinguishers $\mathcal{A}$, there exists a class $\mathcal{A}'$ of single-sample distinguishers that simulates the distinguishers in $\mathcal{A}$ with the choices hard-coded.

3 Prediction Indistinguishability

Before discussing the properties of OI further, we turn our attention to an idealized notion of indistinguishability, which we refer to as prediction indistinguishability (PI). Describing this notion requires the equivalent view on Nature's distribution $\mathcal{D}^*$, where we imagine that Nature selects a marginal distribution over individuals and a true outcome predictor p^*. Distinguishers receive as input an individual-outcome pair $(i, o_i^*) \sim \mathcal{D}^*$ from Nature's distribution, and either Nature's prediction p_i^* or the model's estimate of the parameter $\tilde{p}_i$.

We show that achieving PI may require learning Nature's predictor p^* very precisely, even when $\mathcal{A}$ is a very simple class of distinguishers. This result shows that PI is generally infeasibille due to the ability to access p_i^* directly: even computationally-weak PI distinguishers are incredibly powerful at distinguishing between p^* and $\tilde{p}$. In a sense, the hardness of PI motivates our focus on OI.

Statistical closeness through PI. Prediction indistinguishability requires that the joint distribution of such individual-outcome-prediction triples cannot be significantly distinguished by a family of algorithms $\mathcal{A}$.

Definition 3.1 (Prediction Indistinguishability). *Fix Nature's distribution $\mathcal{D}^*$. Let $\mathcal{Z} = \mathcal{X} \times \{0,1\} \times [0,1]$. For a class of distinguishers $\mathcal{A} : \mathcal{Z} \to [0,1]$ and $\varepsilon > 0$, a predictor $\tilde{p} : \mathcal{X} \to [0,1]$ satisfies $(\mathcal{A}, \varepsilon)$-Prediction Indistinguishability (PI) if for every $A \in \mathcal{A}$,*

$$\left| \Pr_{(i,o_i^*)\sim\mathcal{D}^*} [\ A(i, o_i^*, p_i^*) = 1\ ] - \Pr_{(i,o_i^*)\sim\mathcal{D}^*} [\ A(i, o_i^*, \tilde{p}_i) = 1\ ] \right| \leq \varepsilon.$$

We emphasize that prediction indistinguishability departs from outcome indistinguishability in an essential way, by assuming the distinguisher may receive direct access to Nature's prediction p_i^*.[12]

We show that prediction indistinguishability is too strong a notion of indistinguishability to be broadly useful. Specifically, we show that using a very simple distinguisher, we can test for statistical closeness between nature's predictor p^* and the model's predictor $\tilde{p}$. Given the hardness of recovering individual-level predictions in statistical distance (both information-theoretic and computational), this reduction allows us to conclude that, in general, prediction indistinguishability is infeasible.

Consider the randomized distinguisher A_{ℓ_1} defined as follows.

$$A_{\ell_1}(i, o_i, p_i) = \begin{cases} 0 & \text{w.p. } |o_i - p_i| \\ 1 & \text{o.w.} \end{cases}$$

We argue that if a candidate $\tilde{p}$ passes this single PI-distinguisher, it must have small statistical distance to p^*.

Proposition 3.2. *Fix Nature's distribution $\mathcal{D}^*$ and constant $\varepsilon, \tau \geq 0$; suppose Nature's predictor $p^* : \mathcal{X} \to [0,1]$ is such that $p^* = f + \delta$ for a boolean function $f : \mathcal{X} \to \{0,1\}$ and $\delta : \mathcal{X} \to [-1,1]$ where $\|\delta\|_1 \leq \tau$. Then any $(\{A_{\ell_1}\}, \varepsilon)$-PI predictor $\tilde{p} : \mathcal{X} \to [0,1]$ is statistically close to p^*, satisfying*

$$\|p^* - \tilde{p}\|_1 \leq 4\tau + \varepsilon.$$

Proof. Consider the difference in probabilities of acceptance under that natural and modeled distributions.

$$\Pr_{(i,o_i^*)\sim\mathcal{D}^*} [\ A_{\ell_1}(i, o_i^*, p_i^*) = 1\ ] - \Pr_{(i,o_i^*)\sim\mathcal{D}^*} [\ A_{\ell_1}(i, o_i^*, \tilde{p}_i) = 1\ ]$$

$$= \mathbb{E}_{(i,o_i^*)\sim\mathcal{D}^*} [\ 1 - |o_i^* - p_i^*|\ ] - \mathbb{E}_{(i,o_i^*)\sim\mathcal{D}^*} [\ 1 - |o_i^* - \tilde{p}_i|\ ]$$

$$= \mathbb{E}_{i\sim\mathcal{D}_{\mathcal{X}}} [\ p_i^* \cdot (p_i^* - \tilde{p}_i) + (1 - p_i^*) \cdot (\tilde{p}_i - p_i^*)\ ] \tag{1}$$

---

[12]The assumption that p^* meaningfully exists such that p_i^* can be given as input to a distinguisher breaks the abstraction of $\mathcal{D}^*$, but is a common assumption in the forecasting literature. Still, this is another sense in which PI is an idealized variant of OI, because we can never actually generate individual-outcome-prediction samples from $\mathcal{D}^*$.

Assuming that $\tilde{p}$ is $(\{A_{\ell_1}\}, \varepsilon)$-PI, we can upper bound this quantity by ε. Under the assumption that $p^* = f + \delta$ for boolean f, we will lower bound the quantity in terms of $\|p^* - \tilde{p}\|$ and τ.

$$
\begin{aligned}
(1) &= \mathop{\mathbf{E}}_{i \sim \mathcal{D}_{\mathcal{X}}} \left[\, (f_i + \delta_i) \cdot (f_i + \delta_i - \tilde{p}_i) + (1 - f_i - \delta_i) \cdot (\tilde{p}_i - f_i - \delta_i) \, \right] \\
&\geq \mathop{\mathbf{E}}_{i \sim \mathcal{D}_{\mathcal{X}}} \left[\, f_i \cdot (f_i + \delta_i - \tilde{p}_i) + (1 - f_i) \cdot (\tilde{p}_i - f_i - \delta_i) \, \right] - 2 \, \|\delta\|_1 \\
&\geq \mathop{\mathbf{E}}_{i \sim \mathcal{D}_{\mathcal{X}}} \left[\, f_i \cdot (f_i - \tilde{p}_i) + (1 - f_i) \cdot (\tilde{p}_i - f_i) \, \right] - 3 \, \|\delta\|_1 \\
&= \mathop{\mathbf{E}}_{i \sim \mathcal{D}_{\mathcal{X}}} \left[\, |f_i - \tilde{p}_i| \, \right] - 3 \, \|\delta\|_1 \\
&\geq \|f - \tilde{p}\|_1 - 3\tau \\
&\geq \|p^* - \tilde{p}\|_1 + 4\tau
\end{aligned}
$$

Thus, in combination, we can conclude

$$
\|p^* - \tilde{p}\|_1 - 4\tau \leq (1) \leq \varepsilon
$$

and the proposition follows. $\qquad \square$

We can therefore port any hardness results for recovering p^* in statistical distance to obtaining prediction indistinguishability. For example, if we take p^* to be a random boolean function, then ℓ_1-recovery is information-theoretically impossible unless we observe the outcome o_i^* for a $1 - O(\varepsilon)$ fraction of inputs $i \in \mathcal{X}$. If we restrict ourselves to relatively simple functions, ℓ_1-recovery may be information-theoretically feasible, but computationally infeasible: for instance, if p^* is a pseudorandom function, then any computationally-efficient estimate of $\tilde{p}$ will fail $(\{A_{\ell_1}\}, \varepsilon)$-PI.

PI when distinguishers don't reject natural histories. While in full generality, PI may be infeasible, we show that for a certain broad class of distinguishers —derived from tests used in the forecasting literature—OI and PI are actually equivalent. Recall, in [San03], the goal is to learn a predictive model that passes tests as well as nature, where the tests may be over histories of outcomes. Importantly, in the forecasting literature, the predictor is compared to *every* possible choice of nature, and only needs to perform as well as the "worst" nature.

We show that for tests that accept natural histories with high probability, PI and OI are essentially the same condition. This requirement—that every possible nature must pass the tests—turns out to be very restrictive. We formulate these tests as a restricted family of multi-sample distinguishers.

Definition 3.3 (Tests that rarely reject nature). *Let $m \in \mathbb{N}$ and $\varepsilon \geq 0$, and suppose $\mathcal{A}$ is a family of m-sample PI distinguishers. We say that $\mathcal{A}$ rejects nature with probability at most ε if for every choice of Nature's distribution $\mathcal{D}^* = \mathcal{D}(p^*)$ for predictor $p^* : \mathcal{X} \to [0, 1]$, for every $A \in \mathcal{A}$,*

$$
\mathop{\mathbf{Pr}}_{(i_1, o_{i_1}^*), \ldots, (i_m, o_{i_m}^*) \sim \mathcal{D}(p^*)^m} \left[\, A((i_1, o_{i_1}^*, p_{i_1}^*), \ldots, (i_m, o_{i_m}^*, p_{i_m}^*)) = 1 \, \right] \geq 1 - \varepsilon.
$$

Despite the fact that these PI distinguishers receive access to the true generating parameters of the model, we show the tests are very limited in their distinguishing abilities. In fact, we will show that the equivalence between PI and OI holds for distinguishers satisfying an even weaker property,

which we call the *valid-model* property. Specifically, we will assume that for all $A \in \mathcal{A}$, there exists an acceptance probability q_A, such that for all $p : \mathcal{X} \to [0, 1]$,

$$\left| \Pr_{(i_1, o_{i_1}), \ldots, (i_m, o_{i_m}) \sim \mathcal{D}(p)^m} [\, A\left((i_1, o_{i_1}, p_{i_1}), \ldots, (i_m, o_{i_m}, p_{i_m})\right) = 1 \,] - q_A \right| \leq \varepsilon.$$

Essentially, such valid-model distinguishers may test whether the outcomes o_{i_j} are actually sampled from the model $\mathrm{Ber}(p_{i_j})$ over the collection of samples they receive, but cannot depend on the value of p_{i_j} (independently of o_{i_j}). Tests that rarely reject nature satisfy the valid-model property with $q_A = 1$ for all $A \in \mathcal{A}$.

Proposition 3.4. *Fix Nature's distribution $\mathcal{D}^*$. Suppose for $\varepsilon \geq 0$, $\mathcal{A}$ is a family of distinguishers satisfying the valid-model property. Then, PI and sample-access-OI over $\mathcal{A}$ are equivalent; specifically,*

- *if $\tilde{p}$ is $(\mathcal{A}, \varepsilon)$-sample-access-OI, it is $(\mathcal{A}, 3\varepsilon)$-PI*

- *if $\tilde{p}$ is $(\mathcal{A}, \varepsilon)$-PI, it is $(\mathcal{A}, 3\varepsilon)$-sample-access-OI.*

Proof. Suppose that $\mathcal{A}$ is a family satisfying the valid-model property. First, we show that for all $A \in \mathcal{A}$, by the valid-model property, the acceptance probabilities over $\mathcal{D}(\tilde{p})$ and $\mathcal{D}^* = \mathcal{D}(p^*)$, when we feed A the correct generating probabilities, must be similar.

$$\Pr_{(i_1, \tilde{o}_{i_1}), \ldots, (i_m, \tilde{o}_{i_m}) \sim \mathcal{D}(\tilde{p})^m} [\, A\left((i_1, \tilde{o}_{i_1}, \tilde{p}_{i_1}), \ldots, (i_m, \tilde{o}_{i_m}, \tilde{p}_{i_m})\right) = 1 \,] \tag{2}$$

$$\leq q_A + \varepsilon \tag{3}$$

$$\leq \Pr_{(i_1, o_{i_1}^*), \ldots, (i_m, o_{i_m}^*) \sim \mathcal{D}(p^*)^m} [\, A\left((i_1, o_{i_1}^*, p_{i_1}^*), \ldots, (i_m, o_{i_m}^*, p_{i_m}^*)\right) = 1 \,] + 2\varepsilon \tag{4}$$

These inequalities (and their reverse) follow directly from the valid-model assumption on $\mathcal{A}$. Using this closeness, we can show both directions of the equivalence.

First, suppose the OI distinguishing advantage is bounded by ε.

$$\left| \Pr_{(i_1, o_{i_1}^*), \ldots, (i_m, o_{i_m}^*) \sim (\mathcal{D}^*)^m} [\, A\left((i_1, o_{i_1}^*, \tilde{p}_{i_1}), \ldots, (i_m, o_{i_m}^*, \tilde{p}_{i_m})\right) = 1 \,] \right.$$

$$\left. - \Pr_{(i_1, \tilde{o}_{i_1}), \ldots, (i_m, \tilde{o}_{i_m}) \sim \mathcal{D}(\tilde{p})^m} [\, A\left((i_1, \tilde{o}_{i_1}, \tilde{p}_{i_1}), \ldots, (i_m, \tilde{o}_{i_m}, \tilde{p}_{i_m})\right) = 1 \,] \right| \leq \varepsilon \tag{5}$$

Applying the valid-model inequalities (from (5)→(2)→(3)→(4)), we can bound the PI distinguishing advantage by 3ε.

Next, suppose the PI distinguishing advantage is bounded by ε.

$$\left| \Pr_{(i_1, o_{i_1}^*), \ldots, (i_m, o_{i_m}^*) \sim (\mathcal{D}^*)^m} [\, A\left((i_1, o_{i_1}^*, \tilde{p}_{i_1}), \ldots, (i_m, o_{i_m}^*, \tilde{p}_{i_m})\right) = 1 \,] \right.$$

$$\left. - \Pr_{(i_1, o_{i_1}^*), \ldots, (i_m, o_{i_m}^*) \sim (\mathcal{D}^*)^m} [\, A\left((i_1, o_{i_1}^*, p_{i_1}^*), \ldots, (i_m, o_{i_m}^*, p_{i_m}^*)\right) = 1 \,] \right| \leq \varepsilon \tag{6}$$

Applying the valid-model inequalities (from (2)→(3)→(4)→(6)), we can bound the OI distinguishing advantage by 3ε. □

4 OI and Evidence-Based Fairness

With the formal definitions of the variants of OI in place, we turn to understanding the guarantees implied by OI. In this section, we show a tight connection between the notion of outcome indistinguishability and notions of *evidence-based fairness*.

Evidence-based fairness notions. We review two related notions—*multi-accuracy* and *multi-calibration*—which require a predictor $\tilde{p} : \mathcal{X} \to [0, 1]$ to display "consistency" with p^*, not simply on the population $\mathcal{X}$ as a whole, but also on structured subpopulations of $S \subseteq \mathcal{X}$. The notions are parameterized by a class of subpopulations $\mathcal{C} \subseteq \{0, 1\}^{\mathcal{X}}$, which controls the strength of the guarantee: the richer the class of subpopulations, the stronger the consistency requirement.

For conceptual and algorithmic reasons, it is natural to define the collection $\mathcal{C}$ in terms of a concept class (e.g., conjunctions, halfspaces, decision trees), where each function in the class $c : \mathcal{X} \to \{0, 1\}$ defines the characteristic function for a subpopulation $S_c \in \mathcal{C}$. We review the definitions of multi-accuracy and multi-calibration, and refer the reader to [HKRR18,Kim20] for more in-depth coverage of the definitions and algorithms for achieving the notions.

We start with the notion of multi-accuracy. Given a collection of subpopulations $\mathcal{C}$, multi-accuracy requires that a predictor $\tilde{p}$ reflect the expectations of p^* correctly over each subpopulation $S \in \mathcal{C}$.

Definition 4.1 (Multi-Accuracy [HKRR18]). *Fix a distribution over individuals $\mathcal{D}_{\mathcal{X}}$ and Nature's predictor $p^* : \mathcal{X} \to [0, 1]$. For a collection of sets $\mathcal{C} \subseteq \{0, 1\}^{\mathcal{X}}$ and $\alpha \geq 0$, a predictor $\tilde{p} : \mathcal{X} \to [0, 1]$ satisfies $(\mathcal{C}, \alpha)$-multi-accuracy if for every $S \in \mathcal{C}$,*

$$\left| \underset{i \sim \mathcal{D}_{\mathcal{X}}}{\mathbf{E}} [\, p_i^* \mid i \in S \,] - \underset{i \sim \mathcal{D}_{\mathcal{X}}}{\mathbf{E}} [\, \tilde{p}_i \mid i \in S \,] \right| \leq \alpha \tag{7}$$

Predictors that satisfy multi-accuracy cannot introduce significant bias in their predictions over any subpopulation defined within $\mathcal{C}$. Still, because we average over S, multi-accuracy may allow the predictor to make distinctions between individuals with similar p^* values. For instance, if every individual $i \in \mathcal{X}$ has $p_i^* = 1/2$, then a random boolean function that predicts $\tilde{p}_i \in \{0, 1\}$ for each $i \in \mathcal{X}$ will satisfy multi-accuracy for $\mathcal{C}$ of bounded complexity.

Multi-calibration prevents this type of disparate treatment of similar individuals by requiring the predictor $\tilde{p}$ to be calibrated with respect to p^* over each $S \in \mathcal{C}$. Here, a set of predictions is calibrated if amongst the individuals $i \in \mathcal{X}$ who receive prediction $\tilde{p}_i = v$, their actual expectation is v. Intuitively, calibration goes further than accuracy in expectation and requires that the predictions can be meaningfully interpreted as conditional probabilities.

Technically, to reason about approximate calibration in a way that provides strong guarantees and is practically realizable, we need to work with discretized predictors. Recall that for a predictor $\tilde{p}$, we use $\mathrm{supp}_S(\tilde{p}) = \{v \in [0, 1] : \mathbf{Pr}_{i \sim \mathcal{D}_{\mathcal{X}}} [\, \tilde{p}_i = v \mid i \in S \,] > 0\}$ to denote the support of $\tilde{p}$ on $S \subseteq \mathcal{X}$.

Definition 4.2 (Multi-Calibration [HKRR18]). *Fix a distribution over individuals $\mathcal{D}_\mathcal{X}$ and Nature's predictor $p^* : \mathcal{X} \to [0,1]$. For a collection of sets $\mathcal{C} \subseteq \{0,1\}^\mathcal{X}$ and parameters $\alpha > 0$, a predictor $\tilde{p} : \mathcal{X} \to [0,1]$ satisfies $(\mathcal{C}, \alpha)$-multi-calibration if for every set $S \in \mathcal{C}$ and supported value $v \in \mathrm{supp}_S(\tilde{p})$ such that $\mathbf{Pr}_{i \sim \mathcal{D}_\mathcal{X}}[\ \tilde{p}_i = v \mid i \in S\ ] \geq \alpha / |\mathrm{supp}_S(\tilde{p})|$,*

$$\left| \ \mathbf{E}_{i \sim \mathcal{D}}[\ p_i^* \mid \tilde{p}_i = v \wedge i \in S\ ] - v \ \right| \leq \alpha. \tag{8}$$

Intuitively, by conditioning on the value of $\tilde{p}$ in addition to membership in S, multi-calibration prevents a predictor from treating similar individuals differently. If a multi-calibrated predictor says that the value $\tilde{p}_i = v$, then the "probability" of their outcome being 1 can be thought of as v, where the probability is taken over the randomness in the outcome and the random selection of a member of any $S \in \mathcal{C}$ for which $i \in S$.

We include two technical remarks about Definition 4.2:

- Observe that the definition of multi-calibration excludes from consideration level sets where $\mathbf{Pr}_{i \sim \mathcal{D}_\mathcal{X}}[\ \tilde{p}_i = v \mid i \in S\ ] < \alpha / |\mathrm{supp}_S(\tilde{p})|$. This exclusion is a technicality needed in order to learn multi-calibrated predictors from a small set of training data. Because the tolerance is scaled by the support size of $\tilde{p}$, the total fraction of any subpopulation $S \in \mathcal{C}$ that can be excluded is α. Effectively, this means that predictors have to be confident in their predictions—a more refined predictor with many supported values must satisfy the multi-calibration constraints on smaller slices.

- The notion of approximate calibration used in Definition 4.2 enforces that the expectation of $\tilde{p}$ and p^*, conditioned on $i \in S$ and $\tilde{p}_i = v$, are the same up to a tolerance of α. This relative error guarantee (which becomes more stringent for smaller sets) is deliberate and best captures the intuition that calibrated predictions should "mean what they say." [13]

Relating OI to multi-accuracy and multi-calibration. In the next sections, we demonstrate a tight connection between no-access-OI and multi-accuracy, and sample-access-OI and multi-calibration. We will show that we can implement the OI notion as an instance of multi-group fairness, and vice versa. In particular, given any distinguisher class $\mathcal{A}$, we can construct a corresponding class of subpopulations $\mathcal{C}_\mathcal{A}$, such that multi-calibration over $\mathcal{C}_\mathcal{A}$ implies sample-access-OI over $\mathcal{A}$; and given any class of subpopulations $\mathcal{C}$, we can construct a corresponding distinguisher class $\mathcal{A}_\mathcal{C}$ such that sample-access-OI over $\mathcal{A}_\mathcal{C}$ implies multi-calibration over $\mathcal{C}$ (with small loss in the approximation parameters). In fact, the transformation between distinguishing algorithms $\mathcal{A}$ and subpopulations $\mathcal{C}$ is tight enough that if we repeat the construction a constant number of times, the classes hit a "fixed point" (e.g., $\mathcal{C}_{\mathcal{A}_\mathcal{C}} = \mathcal{C}$).

An important quantity in the analysis of outcome indistinguishability is the advantage of a distinguisher A with respect to a particular predictor p. For example, in sample-access-OI:

---

[13] A natural alternative notion of approximation would allow for *absolute* error, where the slack on the conditional expectation in (8) is scaled as $\alpha / \mathbf{Pr}_{i \sim \mathcal{D}_\mathcal{X}}[\ \tilde{p}_i = v \wedge i \in S\ ]$. While easier to work with algorithmically, this constraint provides minimal guarantees of consistency with p^*. For instance, the predictor that randomly assigns individuals one of $1/\alpha$ values will, with high probability, have $\mathbf{Pr}_{i \sim \mathcal{D}_\mathcal{X}}[\ \tilde{p}_i = v \wedge i \in S\ ] \approx \alpha$ for all $S \in \mathcal{C}$, and thus, each of the constraints would be vacuously satisfied.

$$\Delta_A(p) \triangleq \left| \Pr_{(i,o_i^*)\sim\mathcal{D}^*} [\ A(i, o_i^*, p_i) = 1\ ] - \Pr_{(i,o_i)\sim\mathcal{D}(p)} [\ A(i, o_i, p_i) = 1\ ] \right|$$

It will be natural to define analogous quantities for the evidence-based fairness notions we mentioned above; we refer to these as the *multi-accuracy violation* and *multi-calibration violation*, respectively. Given a set S and possibly a value $v \in [0, 1]$, these are defined as follows:

$$\nabla_S^{\mathbf{MA}}(p) \triangleq \left| \mathop{\mathbf{E}}_{i\sim\mathcal{D}_\mathcal{X}} [\ p_i^* \mid i \in S\ ] - \mathop{\mathbf{E}}_{i\sim\mathcal{D}_\mathcal{X}} [\ p_i \mid i \in S\ ] \right|$$

$$= \left| \Pr_{(i,o_i^*)\sim\mathcal{D}^*} [\ o_i^* = 1 \mid i \in S\ ] - \Pr_{(i,o_i)\sim\mathcal{D}(p)} [\ o_i = 1 \mid i \in S\ ] \right|$$

$$\nabla_{S,v}^{\mathbf{MC}}(p) \triangleq \left| \mathop{\mathbf{E}}_{i\sim\mathcal{D}_\mathcal{X}} [\ p_i^* \mid p_i = v \wedge i \in S\ ] - v \right|$$

$$= \left| \Pr_{(i,o_i^*)\sim\mathcal{D}^*} [\ o_i^* = 1 \mid p_i = v \wedge i \in S\ ] - v \right|$$

Since the subscript allows us to distinguish between the two violations, we will typically just write $\nabla(p)$, and the corresponding fairness notion will be clear from context: ∇_S for the multi-accuracy violation, and $\nabla_{v,S}$ for the multi-calibration violation.

4.1 No-Access-OI and Multi-Accuracy

In this section we show that no-access-OI and multi-accuracy are very closely related. In particular, we prove that each can be implemented using the other, by an appropriate choice of the distinguisher class (resp., of the collection of sets) and accuracy parameters.

Theorem 4.3. *Fix Nature's distribution $\mathcal{D}^*$. Let $\mathcal{Z} = \mathcal{X} \times \{0, 1\}$.*

- Implementing multi-accuracy using OI: *Suppose $\mathcal{C} \subseteq \{0,1\}^\mathcal{X}$ is a collection of subpopulations, and let $\gamma_\mathcal{C} \geq \min_{S\in\mathcal{C}} \Pr_{i\sim\mathcal{D}_\mathcal{X}} [\ i \in S\ ]$. Then there exists a family of distinguishers $\mathcal{A}_\mathcal{C} \subseteq \{\mathcal{Z} \to \{0,1\}\}$ such that for any $\alpha > 0$, if a predictor $\tilde{p} : \mathcal{X} \to [0,1]$ satisfies $(\mathcal{A}_\mathcal{C}, \alpha \cdot \gamma_\mathcal{C})$-no-access-OI, then $\tilde{p}$ satisfies $(\mathcal{C}, \alpha)$-multi-accuracy.*

- Implementing OI using multi-accuracy: *Suppose $\mathcal{A} \subseteq \{\mathcal{Z} \to \{0,1\}\}$ is a family of deterministic distinguishers. Then there exists a collection of subpopulations $\mathcal{C}_\mathcal{A} \subseteq \{0,1\}^\mathcal{X}$ such that for any $\varepsilon > 0$, if a predictor $\tilde{p}$ satisfies $(\mathcal{C}_\mathcal{A}, \varepsilon/2)$-multi-accuracy, then $\tilde{p}$ satisfies $(\mathcal{A}, \varepsilon)$-no-access-OI.*

Note that proof reveals the additional property that the complexity of $\mathcal{C}$ and $\mathcal{A}$ are essentially equivalent. That is, for each $S \in \mathcal{C}$, we construct a distinguisher $A_S \in \mathcal{A}_\mathcal{C}$ such that the complexity of evaluating set membership $\mathbf{1}[\ i \in S\ ]$ is nearly identical to that of computing $A_S(i, b)$ for $b \in \{0, 1\}$.

Proof. We prove each direction separately.

Implementing multi-accuracy using OI. Fix a collection of sets $\mathcal{C}$. We define a collection of $|\mathcal{C}|$ distinguishers $\mathcal{A}_{\mathcal{C}} = \{A_S : S \in \mathcal{C}\}$, where for each $S \in \mathcal{C}$, the distinguisher A_S is defined as:

$$A_S(i, b) = \mathbf{1}\left[\, i \in S \wedge b = 1 \,\right]$$

For any predictor $\tilde{p}$ and set $S \in \mathcal{C}$, we can relate the multi-accuracy violation $\nabla_S(\tilde{p})$ to the distinguishing advantage $\Delta_{A_S}(\tilde{p})$, as follows.

$$
\begin{aligned}
\nabla_S(\tilde{p}) &= \left| \Pr_{(i,o_i^*) \sim \mathcal{D}^*}\left[\, o_i^* = 1 \mid i \in S \,\right] - \Pr_{(i,\tilde{o}_i) \sim \mathcal{D}(\tilde{p})}\left[\, \tilde{o}_i = 1 \mid i \in S \,\right] \right| \\
&= \frac{1}{\Pr_{i \sim \mathcal{D}_{\mathcal{X}}}[i \in S]} \cdot \left| \Pr_{(i,o_i^*) \sim \mathcal{D}^*}\left[\, o_i^* = 1 \wedge i \in S \,\right] - \Pr_{(i,\tilde{o}_i) \sim \mathcal{D}(\tilde{p})}\left[\, \tilde{o}_i = 1 \wedge i \in S \,\right] \right| \\
&= \frac{1}{\Pr_{i \sim \mathcal{D}_{\mathcal{X}}}[i \in S]} \cdot \left| \Pr_{(i,o_i^*) \sim \mathcal{D}^*}\left[\, A_S(i, o_i^*) = 1 \,\right] - \Pr_{(i,\tilde{o}_i) \sim \mathcal{D}(\tilde{p})}\left[\, A_S(i, \tilde{o}_i) = 1 \,\right] \right| \\
&= \frac{1}{\Pr_{i \sim \mathcal{D}_{\mathcal{X}}}[i \in S]} \cdot \Delta_{A_S}(\tilde{p})
\end{aligned}
$$

Therefore, if $\tilde{p}$ satisfies $(\mathcal{A}_{\mathcal{C}}, \alpha \cdot \gamma_{\mathcal{C}})$-no-access-OI, we derive the following string of inequalities.

$$\nabla_S(\tilde{p}) = \frac{\Delta_{A_S}(\tilde{p})}{\Pr_{i \sim \mathcal{D}_{\mathcal{X}}}[i \in S]} \leq \frac{\Delta_{A_S}(\tilde{p})}{\gamma_{\mathcal{C}}} \leq \frac{\alpha \cdot \gamma_{\mathcal{C}}}{\gamma_{\mathcal{C}}} = \alpha$$

Thus, for every $S \in \mathcal{C}$ we have $\nabla_S(\tilde{p}) \leq \alpha$, and we conclude that $\tilde{p}$ satisfies $(\mathcal{C}, \alpha)$-multi-accuracy.

Implementing OI using multi-accuracy. Fix a collection of deterministic distinguishers $\mathcal{A}$. We define a collection of $2 \cdot |\mathcal{A}|$ sets $\mathcal{C}_{\mathcal{A}} = \{S_{(A,b)} : A \in \mathcal{A}, b \in \{0,1\}\}$, where for each $A \in \mathcal{A}$ and $b \in \{0,1\}$, the subpopulation $S_{(A,b)}$ is defined as

$$S_{(A,b)} = \{i \in \mathcal{X} : A(i,b) = 1\}$$

For any predictor $\tilde{p}$ and distinguisher $A \in \mathcal{A}$, we can relate the distinguishing advantage $\Delta_A(p)$ to

the multi-accuracy violations $\nabla_{S_{(A,1)}}$ and $\nabla_{S_{(A,0)}}$, as follows.

$$
\Delta_A(\tilde{p}) = \left| \Pr_{(i,o_i^*)\sim\mathcal{D}^*} [\, A(i, o_i^*) = 1 \,] - \Pr_{(i,\tilde{o}_i)\sim\mathcal{D}(\tilde{p})} [\, A(i, \tilde{o}_i) = 1 \,] \right|
$$

$$
= \left| \sum_{b\in\{0,1\}} \left(\Pr_{(i,o_i^*)\sim\mathcal{D}^*} [\, A(i, b) = 1 \wedge o_i^* = b \,] - \Pr_{(i,\tilde{o}_i)\sim\mathcal{D}(\tilde{p})} [\, A(i, b) = 1 \wedge \tilde{o}_i = b \,] \right) \right|
$$

$$
\leq \sum_{b\in\{0,1\}} \left| \Pr_{(i,o_i^*)\sim\mathcal{D}^*} [\, A(i, b) = 1 \wedge o_i^* = b \,] - \Pr_{(i,\tilde{o}_i)\sim\mathcal{D}(\tilde{p})} [\, A(i, b) = 1 \wedge \tilde{o}_i = b \,] \right|
$$

$$
= \sum_{b\in\{0,1\}} \left| \Pr_{(i,o_i^*)\sim\mathcal{D}^*} [\, i \in S_{(A,b)} \wedge o_i^* = b \,] - \Pr_{(i,\tilde{o}_i)\sim\mathcal{D}(\tilde{p})} [\, i \in S_{(A,b)} \wedge \tilde{o}_i = b \,] \right|
$$

$$
= \sum_{b\in\{0,1\}} \Pr_{i\sim\mathcal{D}_\mathcal{X}} [\, i \in S_{(A,b)} \,] \cdot \left| \Pr_{(i,o_i^*)\sim\mathcal{D}^*} [\, o_i^* = b \mid i \in S_{(A,b)} \,] - \Pr_{(i,\tilde{o}_i)\sim\mathcal{D}(\tilde{p})} [\, \tilde{o}_i = b \mid i \in S_{(A,b)} \,] \right|
$$

$$
= \sum_{b\in\{0,1\}} \Pr_{i\sim\mathcal{D}_\mathcal{X}} [\, i \in S_{(A,b)} \,] \cdot \nabla_{S_{(A,b)}}(\tilde{p})
$$

$$
\leq \nabla_{S_{(A,0)}}(\tilde{p}) + \nabla_{S_{(A,1)}}(\tilde{p})
$$

Therefore, if $\tilde{p}$ satisfies $(\mathcal{C}_\mathcal{A}, \varepsilon/2)$-multi-accuracy, then we have $\Delta_A(\tilde{p}) \leq \varepsilon$ for every $A \in \mathcal{A}$; we conclude that $\tilde{p}$ satisfies $(\mathcal{A}, \varepsilon)$-no-access-OI. $\qquad\square$

4.2 Sample-Access-OI and Multi-Calibration

In this section we establish a similar relationship, this time proving that sample-access-OI and multi-calibration can each be implemented using the other. One technical subtlety is that unlike the previous level, we need to work with some fixed discretization of the predictor in question. Formally, for any $m \in \mathbb{N}$, we use $\tilde{p}^{1/m}$ to denote the "rounded" version of a predictor $p : \mathcal{X} \to [0,1]$, with respect to the grid $G_m = \left\{ \frac{1}{2m}, \frac{2}{2m}, \dots, \frac{2m-1}{2m} \right\}$ that partitions the interval $[0,1]$ into m "bins" of width $1/m$:

$$
\forall i \in \mathcal{X} : \qquad \tilde{p}_i^{1/m} \equiv u(p_i)
$$

where for $v \in [0,1]$, $u(v) \in G_m$ is the closest value to v that is on the grid: $u(v) \equiv \operatorname{argmin}_{u \in G_m} |u - v|$. Note that by definition, for every $i \in \mathcal{X}$, $\left| \tilde{p}_i^{1/m} - p_i \right| \equiv |u(p_i) - p_i| \leq 1/2m$.

Theorem 4.4. *Fix nature's distribution $\mathcal{D}^*$. Let $\mathcal{Z} = \mathcal{X} \times \{0,1\} \times [0,1]$.*

1. Implementing multi-calibration using OI: *Suppose $\mathcal{C} \subseteq \{0,1\}^\mathcal{X}$ is a collection of subpopulations, and let $\gamma_C \geq \min_{S\in\mathcal{C}} \Pr_{i\sim\mathcal{D}_\mathcal{X}} [\, i \in S \,]$. Then for every $m \in \mathbb{N}$, there exists a family of distinguishers $\mathcal{A}_{\mathcal{C},m} \subseteq \{\mathcal{Z} \to \{0,1\}\}$ such that for any $\alpha > 0$ and for every predictor $p : \mathcal{X} \to [0,1]$, if $\tilde{p}^{1/m}$ satisfies $(\mathcal{A}_{\mathcal{C},m}, \alpha^2 \cdot \gamma_\mathcal{C}/m)$-sample-access-OI, then $\tilde{p}^{1/m}$ satisfies $(\mathcal{C}, \alpha)$-multi-calibration.*

2. Implementing OI using multi-calibration: *Suppose $\mathcal{A} \subseteq \{\mathcal{Z} \to \{0,1\}\}$ is a family of deterministic distinguishers. Then for every $m \in \mathbb{N}$, there exists a collection of subpopulations*

31

$\mathcal{C}_{\mathcal{A},m} \subseteq \{0,1\}^{\mathcal{X}}$ such that for any $\varepsilon > 0$ and for every predictor $\tilde{p} : \mathcal{X} \to [0,1]$, if $\tilde{p}^{1/m}$ satisfies $(\mathcal{C}_{\mathcal{A},m}, \varepsilon/4)$-multi-calibration, then $\tilde{p}^{1/m}$ satisfies $(\mathcal{A}, \varepsilon)$-sample-access-OI.

Proof. We prove each direction separately.

Implementing multi-calibration using OI. Fix $\mathcal{C}$ and $m \in \mathbb{N}$. We define a family of $|\mathcal{C}| \cdot m$ distinguishers, $\mathcal{A}_{\mathcal{C},m} = \{A_{S,u} \mid u \in G_m, S \in \mathcal{C}\}$, where:

$$A_{u,S}(i,b,v) = \mathbf{1}\left[\, i \in S \wedge b = 1 \wedge |u - v| \le 1/2m \,\right]$$

For any subpopulation $S \in \mathcal{C}$ and value $u \in G_m$, we can relate the distinguishing advantage $\Delta_{A_{u,S}}(\tilde{p}^{1/m})$ to the multi-calibration violation $\nabla_{u,S}(\tilde{p}^{1/m})$, as follows:

$$\Delta_{A_{u,S}}(\tilde{p}^{1/m})$$

$$= \left| \Pr_{(i,o_i^*) \sim \mathcal{D}^*}\left[\, A_{u,S}(i, o_i^*, \tilde{p}_i^{1/m}) = 1 \,\right] - \Pr_{(i,\tilde{o}_i) \sim \mathcal{D}(\tilde{p}^{1/m})}\left[\, A_{u,S}(i, \tilde{o}_i, \tilde{p}_i^{1/m}) = 1 \,\right] \right|$$

$$= \left| \Pr_{(i,o_i^*) \sim \mathcal{D}^*}\left[\, i \in S \wedge o_i^* = 1 \wedge \left|u - \tilde{p}_i^{1/m}\right| \le 1/2m \,\right] - \Pr_{(i,\tilde{o}_i) \sim \mathcal{D}(\tilde{p}^{1/m})}\left[\, i \in S \wedge \tilde{o}_i = 1 \wedge \left|u - \tilde{p}_i^{1/m}\right| \le 1/2m \,\right] \right|$$

$$= \left| \Pr_{(i,o_i^*) \sim \mathcal{D}^*}\left[\, i \in S \wedge o_i^* = 1 \wedge \tilde{p}_i^{1/m} = u \,\right] - \Pr_{(i,\tilde{o}_i) \sim \mathcal{D}(\tilde{p}^{1/m})}\left[\, i \in S \wedge \tilde{o}_i = 1 \wedge \tilde{p}_i^{1/m} = u \,\right] \right|$$

$$= \Pr_{i \sim \mathcal{D}_{\mathcal{X}}}[\tilde{p}_i^{1/m} = u \wedge i \in S] \cdot \left| \Pr_{(i,o_i^*) \sim \mathcal{D}^*}\left[\, o_i^* = 1 \,\Big|\, \tilde{p}_i^{1/m} = u \wedge i \in S \,\right] - \Pr_{(i,\tilde{o}_i) \sim \mathcal{D}(\tilde{p}^{1/m})}\left[\, \tilde{o}_i = 1 \,\Big|\, \tilde{p}_i^{1/m} = u \wedge i \in S \,\right] \right|$$

$$= \Pr_{i \sim \mathcal{D}_{\mathcal{X}}}[\tilde{p}_i^{1/m} = u \wedge i \in S] \cdot \nabla_{u,S}(\tilde{p}^{1/m})$$

Suppose $\tilde{p}^{1/m}$ satisfies $(\mathcal{A}_{\mathcal{C},m}, \varepsilon)$-sample-access-OI for $\varepsilon = \frac{\alpha^2 \cdot \gamma_{\mathcal{C}}}{m}$. To prove $\tilde{p}^{1/m}$ satisfies multi-calibration, we must bound the multi-calibration violation $\nabla_{u,S}(\tilde{p}^{1/m})$ whenever u, S are such that $\Pr_{i \sim \mathcal{D}_{\mathcal{X}}}\left[\, \tilde{p}_i^{1/m} = u \,\Big|\, i \in S \,\right] \ge \alpha/\operatorname{supp}_S(\tilde{p}^{1/m})$.

Suppose u, S satisfy $\Pr_{i \sim \mathcal{D}_{\mathcal{X}}}\left[\, \tilde{p}_i^{1/m} = u \,\Big|\, i \in S \,\right] \ge \alpha/\operatorname{supp}_S(\tilde{p}^{1/m}) \ge \alpha/m$. Note that this implies

$$\Pr_{i \sim \mathcal{D}_{\mathcal{X}}}\left[\, \tilde{p}_i^{1/m} = u \wedge i \in S \,\right] = \Pr_{i \sim \mathcal{D}_{\mathcal{X}}}\left[\, \tilde{p}_i^{1/m} = u \,\Big|\, i \in S \,\right] \cdot \Pr_{i \sim \mathcal{D}_{\mathcal{X}}}\left[\, i \in S \,\right] \ge \frac{\alpha \cdot \gamma_{\mathcal{C}}}{m}$$

Using the above, we obtain:

$$\nabla_{u,S}(\tilde{p}^{1/m}) = \frac{\Delta_{A_{u,S}}(\tilde{p})}{\Pr_{i \sim \mathcal{D}_{\mathcal{X}}}[\tilde{p}_i^{1/m} = u \wedge i \in S]} \le \frac{\alpha^2 \cdot \gamma_{\mathcal{C}}/m}{\alpha \cdot \gamma_{\mathcal{C}}/m} = \alpha$$

Which guarantees that $\tilde{p}^{1/m}$ satisfies $(\mathcal{C}, \alpha)$-multi-calibration, as required.

Implementing OI using multi-calibration. Fix a collection of deterministic distinguishers $\mathcal{A}$ and $m \in \mathbb{N}$. We define a family of $2m \cdot |\mathcal{A}|$ sets, $\mathcal{C}_{\mathcal{A},m} = \{S_{A,b,u} \mid A \in \mathcal{A}, b \in \{0,1\}, u \in G_m\}$, where:

$$S_{A,b,u} = \{i \in \mathcal{X} : \ A(i,b,u) = 1\}$$

To simplify notation, in what follows we denote

$$\gamma_{A,b,u} \triangleq \Pr_{i \sim \mathcal{D}_{\mathcal{X}}} [\tilde{p}_i^{1/m} = u \land i \in S_{A,b,u}]$$

For any distinguisher $A \in \mathcal{A}$, we can relate the distinguishing advantage $\Delta_A(\tilde{p}^{1/m})$ to the multi-calibration violations $\nabla_{S_{A,b,u}}(\tilde{p}^{1/m})$ for $b \in \{0,1\}$ and $u \in G_m$, as follows:

$$\Delta_A(\tilde{p}^{1/m}) =$$

$$= \left| \Pr_{(i,o_i^*) \sim \mathcal{D}^*} \left[A(i, o_i^*, \tilde{p}_i^{1/m}) = 1 \right] - \Pr_{(i,\tilde{o}_i) \sim \mathcal{D}(\tilde{p}^{1/m})} \left[A(i, \tilde{o}_i, \tilde{p}_i^{1/m}) = 1 \right] \right|$$

$$= \left| \sum_{u,b} \left[\Pr_{(i,o_i^*) \sim \mathcal{D}^*} \left[o^* = b \land \tilde{p}_i^{1/m} = u \land A(i,b,u) = 1 \right] - \Pr_{(i,\tilde{o}_i) \sim \mathcal{D}(\tilde{p}^{1/m})} \left[\tilde{o}_i = b \land \tilde{p}_i^{1/m} = u \land A(i,b,u) = 1 \right] \right] \right|$$

$$= \left| \sum_{u,b} \gamma_{A,b,u} \cdot \left[\Pr_{(i,o_i^*) \sim \mathcal{D}^*} \left[o^* = b \ \middle| \ \tilde{p}_i^{1/m} = u \land i \in S_{A,b,u} \right] - \Pr_{(i,\tilde{o}_i) \sim \mathcal{D}(\tilde{p}^{1/m})} \left[\tilde{o}_i = b \ \middle| \ \tilde{p}_i^{1/m} = u \land i \in S_{A,b,u} \right] \right] \right|$$

$$\leq \sum_{u,b} \gamma_{A,b,u} \cdot \left| \left[\Pr_{(i,o_i^*) \sim \mathcal{D}^*} \left[o^* = b \ \middle| \ \tilde{p}_i^{1/m} = u \land i \in S_{A,b,u} \right] - \Pr_{(i,\tilde{o}_i) \sim \mathcal{D}(\tilde{p}^{1/m})} \left[\tilde{o}_i = b \ \middle| \ \tilde{p}_i^{1/m} = u \land i \in S_{A,b,u} \right] \right] \right|$$

$$= \sum_{u,b} \gamma_{A,b,u} \cdot \left| \left[\Pr_{(i,o_i^*) \sim \mathcal{D}^*} \left[o^* = 1 \ \middle| \ \tilde{p}_i^{1/m} = u \land i \in S_{A,b,u} \right] - \Pr_{(i,\tilde{o}_i) \sim \mathcal{D}(\tilde{p}^{1/m})} \left[\tilde{o}_i = 1 \ \middle| \ \tilde{p}_i^{1/m} = u \land i \in S_{A,b,u} \right] \right] \right|$$

$$= \sum_{u,b} \gamma_{A,b,u} \cdot \nabla_{S_{A,b,u}}(\tilde{p}^{1/m})$$

Suppose $\tilde{p}^{1/m}$ satisfies $(\mathcal{C}, \alpha)$-multi-calibration, for some $\alpha > 0$. Recall that, by definition, multi-calibration provides a bound on the multi-calibration violation $\nabla_{S_{A,b,u}}(\tilde{p}^{1/m})$ whenever A, b, u are such that $\Pr_{i \sim \mathcal{D}_{\mathcal{X}}} \left[\tilde{p}_i^{1/m} = u \ \middle| \ i \in S_{A,b,u} \right] < \alpha/\mathrm{supp}_{S_{A,b,u}}(\tilde{p}^{1/m})$. We therefore proceed by splitting the above sum into two parts, which we denote by I_{small} and I_{large}. In particular, for $u \in G_m$ and $b \in \{0,1\}$:

$$(u,b) \in I_{\mathrm{small}} \iff \Pr_{i \sim \mathcal{D}_{\mathcal{X}}} \left[\tilde{p}_i^{1/m} = u \ \middle| \ i \in S_{A,b,u} \right] < \alpha/m$$

And we let $I_{\mathrm{large}} = G_m \times \{0,1\} - I_{\mathrm{small}}$. As mentioned, for these "large" sets, multi-calibration guarantees an upper bound of α on the multi-calibration violation:

$$\sum_{(u,b) \in I_{\text{large}}} \gamma_{A,b,u} \cdot \nabla_{S_{A,b,u}}(\tilde{p}^{1/m}) = \sum_{u : (u,0) \in I_{\text{large}}} \gamma_{A,0,u} \cdot \nabla_{S_{A,0,u}}(\tilde{p}^{1/m}) + \sum_{u : (u,1) \in I_{\text{large}}} \gamma_{A,1,u} \cdot \nabla_{S_{A,1,u}}(\tilde{p}^{1/m})$$

$$\leq \max_{u : (u,0) \in I_{\text{large}}} \nabla_{S_{A,0,u}}(\tilde{p}^{1/m}) + \max_{u : (u,1) \in I_{\text{large}}} \nabla_{S_{A,1,u}}(\tilde{p}^{1/m})$$

$$\leq 2\alpha$$

Where the above uses the fact that $\sum_i a_i x_i \leq \max_i x_i$ whenever $\sum_i a_i \leq 1$.

For the "small sets" we use a trivial upper bound on the multi-calibration violation, but exploit the fact that the overall mass there is small:

$$\sum_{(u,b) \in I_{\text{small}}} \gamma_{A,b,u} \cdot \nabla_{S_{A,b,u}}(\tilde{p}^{1/m}) \leq \sum_{(u,b) \in I_{\text{small}}} \gamma_{A,b,u}$$

$$= \sum_{(u,b) \in I_{\text{small}}} \Pr_{i \sim \mathcal{D}_{\mathcal{X}}} [\tilde{p}_i^{1/m} = u \wedge i \in S_{A,b,u}]$$

$$\leq \sum_{(u,b) \in I_{\text{small}}} \Pr_{i \sim \mathcal{D}_{\mathcal{X}}} \left[\tilde{p}_i^{1/m} = u \mid i \in S_{A,b,u} \right]$$

$$\leq \sum_{(u,b) \in I_{\text{small}}} \alpha/m$$

$$\leq 2\alpha$$

Put together, we obtain

$$\Delta_A(\tilde{p}^{1/m}) \leq \sum_{(u,b) \in I_{\text{small}}} \gamma_{A,b,u} \cdot \nabla_{S_{A,b,u}}(\tilde{p}^{1/m}) + \sum_{(u,b) \in I_{\text{large}}} \gamma_{A,b,u} \cdot \nabla_{S_{A,b,u}}(\tilde{p}^{1/m}) \leq 4\alpha$$

Taking $\alpha = \varepsilon/4$, we conclude that $\tilde{p}^{1/m}$ satisfying $(\mathcal{C}, \varepsilon/4)$ multi-calibration guarantees that for every $A \in \mathcal{A}$, $\Delta_A(\tilde{p}^{1/m}) \leq \varepsilon$. This means $\tilde{p}^{1/m}$ satisfies $(\mathcal{A}, \varepsilon)$-sample-access-OI.

$\square$

5 Outcome Indistinguishability beyond Multi-Calibration

In this section, we give a generic construction that shows how to construct predictors satisfying outcome indistinguishability (for any level of access to $\tilde{p}$). We analyze the resulting complexity of $\tilde{p}$ for each notion of OI. Informally, for no-access-OI and sample-access-OI, the construction establishes that the existence of OI predictors whose complexity scales linearly with the complexity of $\mathcal{A}$ and inverse polynomially on ε. Note that by the reductions of Section 4 and prior algorithmic results of [HKRR18], we could obtain predictors for no-access-OI and sample-access-OI with similar guarantees through reduction to obtaining multi-accuracy or multi-calibration, respectively.

Algorithm 1 CONSTRUCT OUTCOME INDISTINGUISHABILITY

$p^{(0)}(\cdot) \leftarrow 1/2$ //initialize to constant predictor

repeat for $t = 0, 1, \ldots$

   for all $A \in \mathcal{A}$.

     $\Delta_A^{(t)} \leftarrow \Pr_{(i,o_i^*) \sim \mathcal{D}^*}\big[\, A(i, o_i^*; p^{(t)}) = 1 \,\big] - \Pr_{(i,\tilde{o}_i) \sim \mathcal{D}(p^{(t)})}\big[\, A(i, \tilde{o}_i; p^{(t)}) = 1 \,\big]$

   end for   // distinguishing advantage of A

   if $\exists A \in \mathcal{A}$ s.t. $\left|\Delta_A^{(t)}\right| > \varepsilon$ **then**

     $\delta_A^{(t)}(\cdot) \leftarrow A(\cdot, 1; p^{(t)}) - A(\cdot, 0; p^{(t)})$ // update defined by delta

     $p^{(t+1)}(\cdot) \leftarrow \pi_{[0,1]}\left(p^{(t)}(\cdot) + \frac{\Delta_A^{(t)}}{2} \cdot \delta_A^{(t)}(\cdot)\right)$ // project onto interval [0,1]

     continue   // next iteration upon update

   end if

   return $\tilde{p} = p^{(t)}$ // return when no A distinguishes

---

For oracle-access-OI and code-access-OI, the construction establishes novel bounds on the complexity required to obtain each notion. We prove that, in each case, OI predictors $\tilde{p}$ exist in complexity that scales independently of the complexity of Nature's predictor p^*, but with considerably worse dependence on the approximation parameter, growing exponentially with $1/\varepsilon$ for oracle-access-OI and doubly-exponentially with $1/\varepsilon$ for the most general form of code-access-OI. In Section 6, we show evidence that some exponential dependence on $1/\varepsilon$ is necessary under plausible complexity conjectures.

Our proof is constructive: we describe an algorithm (Algorithm 1) that iteratively refines a candidate predictor, until no more distinguishers $A \in \mathcal{A}$ can distinguish between the distributions induced by the model and p^*. The resulting predictor, naturally, satisfies outcome indistinguishability. Thus, the remaining analysis bounds the number of iterations necessary to terminate, and how the complexity of $\tilde{p}$ scales with each additional iteration. We describe the algorithm in a generic manner that works for distinguishers with any degree of access to the candidate model.

The iterative algorithm can be viewed as a variant of gradient descent or boosting, following the works of [TTV09, HKRR18, KGZ19]. In this exposition, we focus on demonstrating the feasibility of the resulting OI predictors, not on the feasibility of implementing the algorithm efficiently (statistically or computationally) which is the focus of [HKRR18]. We discuss issues of sample complexity and running time briefly in Section 5.4.

Consider Algorithm 1. The procedure starts with an (arbitrary) initial guess of the constant function $p_i^{(0)} = 1/2$ for all $i \in \mathcal{X}$. Then, in each iteration t, we check if there is any $A \in \mathcal{A}$ that distinguishes Nature's distribution $\mathcal{D}^*$ and the predictor's distribution $\mathcal{D}(p^{(t)})$. If there is no such $A \in \mathcal{A}$, then the procedure terminates—by construction this $\tilde{p} = p^{(t)}$ is $(\mathcal{A}, \varepsilon)$-OI. If there is some $A \in \mathcal{A}$ for which $\left|\Delta_A^{(t)}\right| > \varepsilon$, then we perform an additive update to $p^{(t)}$ that is designed to bring $p^{(t+1)}$ closer to the optimal predictor p^*.

Bounding the iteration complexity. First, we show that the procedure is guaranteed to terminate after a finite number of updates.

Lemma 5.1. *Algorithm 1 terminates after at most $T \leq O(1/\varepsilon^2)$ iterations.*

Proof. Fixing Nature's distribution $\mathcal{D}^*$, recall we use $p^* : \mathcal{X} \to [0,1]$ to denote Nature's predictor (i.e., the true conditional probability distribution of outcomes). The iteration complexity follows by a potential argument. For a predictor $p : \mathcal{X} \to [0,1]$, we define a potential function $\phi(p)$ to track the mean squared error to p^*.

$$\phi(p) = \mathop{\mathbf{E}}_{i \sim \mathcal{D}_{\mathcal{X}}} \left[(p_i^* - p_i)^2 \right]$$

The potential is bounded $\phi(p) \leq 1$ for all predictors p (including the initial constant predictor). Thus, to bound the number of iterations by $T \leq O(1/\varepsilon^2)$, we show that after every update the potential drops significantly; specifically, $\phi(p^{(t)}) - \phi(p^{(t+1)}) \geq \Omega(\varepsilon^2)$.

$$\phi\left(p^{(t)}\right) - \phi\left(p^{(t+1)}\right) = \mathop{\mathbf{E}}_{i \sim \mathcal{D}_{\mathcal{X}}} \left[\left(p_i^* - p^{(t)}\right)^2 - \left(p_i^* - p^{(t+1)}\right)^2 \right]$$

$$\geq \mathop{\mathbf{E}}_{i \sim \mathcal{D}_{\mathcal{X}}} \left[\left(p_i^* - p^{(t)}\right)^2 - \left(p_i^* - p^{(t)} - \frac{\Delta_A^{(t)}}{2} \cdot \delta_A^{(t)}(i)\right)^2 \right]$$

$$= \Delta_A^{(t)} \cdot \mathop{\mathbf{E}}_{i \sim \mathcal{D}_{\mathcal{X}}} \left[\left(p_i^* - p_i^{(t)}\right) \cdot \delta_A^{(t)}(i) \right] - \frac{(\Delta_A^{(t)})^2}{4} \cdot \mathop{\mathbf{E}}_{i \sim \mathcal{D}_{\mathcal{X}}} \left[\delta_A^{(t)}(i)^2 \right]$$

$$\geq \Delta_A^{(t)} \cdot \mathop{\mathbf{E}}_{i \sim \mathcal{D}_{\mathcal{X}}} \left[\left(p_i^* - p_i^{(t)}\right) \cdot \delta_A^{(t)}(i) \right] - \frac{(\Delta_A^{(t)})^2}{4}$$

We complete the proof by showing that if the update in Algorithm 1 occurs based on $A \in \mathcal{A}$, then $\mathbf{E}_{i \sim \mathcal{D}_{\mathcal{X}}} \left[(p_i^* - p_i^{(t)}) \cdot \delta_A^{(t)}(i) \right] = \Delta_A^{(t)} \geq \varepsilon$, so the overall progress is at least $3\varepsilon^2/4$.

$$\Delta_A^{(t)} = \mathop{\mathbf{Pr}}_{(i,o_i^*) \sim \mathcal{D}^*} \left[A(i, o_i^*; p^{(t)}) = 1 \right] - \mathop{\mathbf{Pr}}_{(i,\tilde{o}_i) \sim \mathcal{D}(p^{(t)})} \left[A(i, \tilde{o}_i; p^{(t)}) = 1 \right]$$

$$= \sum_{b \in \{0,1\}} \left(\mathop{\mathbf{Pr}}_{(i,o_i^*) \sim \mathcal{D}^*} \left[A(i, b; p^{(t)}) = 1 \wedge o_i^* = b \right] - \mathop{\mathbf{Pr}}_{(i,\tilde{o}_i) \sim \mathcal{D}(p^{(t)})} \left[A(i, b; p^{(t)}) = 1 \wedge \tilde{o}_i = b \right] \right)$$

$$= \sum_{b \in \{0,1\}} \left(\mathop{\mathbf{E}}_{(i,o_i^*) \sim \mathcal{D}^*} \left[A(i, b; p^{(t)}) \cdot \mathbf{1}\left[o_i^* = b \right] \right] - \mathop{\mathbf{E}}_{(i,\tilde{o}_i) \sim \mathcal{D}(p^{(t)})} \left[A(i, b; p^{(t)}) \cdot \mathbf{1}\left[\tilde{o}_i = b \right] \right] \right)$$

$$= \sum_{b \in \{0,1\}} \mathop{\mathbf{E}}_{i \sim \mathcal{D}_{\mathcal{X}}} \left[A(i, b; p^{(t)}) \cdot \left(\mathop{\mathbf{Pr}}_{o_i^* \sim \mathrm{Ber}(p_i^*)} \left[o_i^* = b \right] - \mathop{\mathbf{Pr}}_{\tilde{o}_i \sim \mathrm{Ber}(p_i^{(t)})} \left[\tilde{o}_i = b \right] \right) \right]$$

$$= \mathop{\mathbf{E}}_{i \sim \mathcal{D}_{\mathcal{X}}} \left[A(i, 1; p^{(t)}) \cdot \left(p_i^* - p_i^{(t)}\right) + A(i, 0; p^{(t)}) \cdot \left((1 - p_i^*) - (1 - p_i^{(t)})\right) \right]$$

$$= \mathop{\mathbf{E}}_{i \sim \mathcal{D}_{\mathcal{X}}} \left[\left(p_i^* - p_i^{(t)}\right) \cdot \left(A(i, 1; p^{(t)}) - A(i, 0; p^{(t)})\right) \right]$$

$$= \mathop{\mathbf{E}}_{i \sim \mathcal{D}_{\mathcal{X}}} \left[\left(p_i^* - p_i^{(t)}\right) \cdot \delta_A^{(t)}(i) \right]$$

36

Thus, each iteration causes the potential to drop by at least $\Omega(\varepsilon^2)$, and the number of iterations is bounded by $T \le O(1/\varepsilon^2)$. $\qquad\square$

Bounding the predictor's complexity. In the remainder of this section, we discuss how to actually implement the updates to build up the circuit that computes $\tilde{p}$. At a high level, we maintain a circuit to compute $p^{(t)}$ in each iteration. Initially, $p^{(0)}$ computes a constant function. Then, at each iteration, we build $p^{(t+1)}$ from $p^{(t)}$ by constructing a new circuit that computes the addition of $p^{(t)}$ and (a scaling of) $\delta_A^{(t)}$. Importantly, $\delta_A^{(t)}$ can be computed using two evaluations of some $A \in \mathcal{A}$; hence, the size of the circuit computing the eventual $\tilde{p}$ will depend on the circuit complexity of the family of distinguishers $\mathcal{A}$.

For no-access-OI and sample-access-OI, the bounds on circuit complexity follow directly from this overview. We overview this analysis next. When considering oracle-access-OI and code-access-OI, the distinguishers at each iteration depend non-trivially on the current predictor in question. As such, in each iteration, we need to incorporate $p^{(t)}$, not only to maintain the existing predictions, but also to compute $A^{p^{(t)}}$ itself within the computation of $\delta_A^{(t)}$. For each variant, we describe how to implement these distinguishers in a way that we can bound the complexity of the eventual $\tilde{p}$ produced.

Concrete complexity assumptions. To discuss the complexity needed to represent predictors satisfying OI, we must fix a model of computation for the distinguishers. We analyze Algorithm 1 assuming that each $A \in \mathcal{A}$ is implemented by boolean circuits satisfying some minimal properties. While we analyze the complexity of $\tilde{p}$ in terms of circuit size, we could similarly analyze the time or space complexity needed to evaluate $\tilde{p}$ under a uniform distinguishing model.

Recall, we assume that each $i \in \mathcal{X}$ can be represented by a boolean vector in $\{0,1\}^d$. When relevant, we parameterize the class of distinguishers by the dimension d. We assume that the distinguisher class $\mathcal{A}$ can be implemented by bounded fan-in circuits of size $s(d)$. Typically, we think of $d \in \mathbb{N}$ as fixed by the prediction problem at hand; correspondingly, we can think of $s(d) = s$ to be a fixed bound on the circuit size for some $s \in \mathbb{N}$.

Finally, we make an observation and a few technical assumptions for convenience of analysis. First, we note that Algorithm 1 can be implemented to build $\tilde{p}$ using computations up to a fixed precision of $\Theta(\varepsilon)$, while maintaining the iteration complexity $T = O(1/\varepsilon^2)$ (i.e., with only a constant factor increase). Assuming a fixed precision, we make some additional assumptions about the complexity of implementing logic over values from $[0,1]$, needed to implement the circuit built by Algorithm 1. Specifically, the circuit we build requires operations to:

- fixed-precision addition over $[0,1]$;

- multiplication of fixed-precision values in $[0,1]$ by $b \in \{0,1\}$; and

- maintaining fixed-precision values in $[0,1]$ (i.e., projection)

We assume that each of these operations can be implemented for $\Theta(\varepsilon)$-precision in $w(\varepsilon)$ gates. Throughout, for technical simplicity, we assume that $s(d) \ge d \ge 3 \cdot w(\varepsilon)$.

5.1 Recovering bounds for No-Access-OI and Sample-Access-OI

To begin, we show that Algorithm 1 produces circuits for no-access-OI and sample-access-OI of complexity that scales linearly in the complexity of the distinguishers and inverse polynomially in the approximation parameter ε. The analysis will be identical for each notion, so we state the theorem for the stronger notion of sample-access-OI.

Theorem 5.2. *Let $\mathcal{X} \subseteq \{0,1\}^d$ for $d \leq s \in \mathbb{N}$, and suppose $\mathcal{A}$ is a class of distinguishers implemented by size-s circuits. For any choice of Nature's distribution $\mathcal{D}^*$ and $\varepsilon > 0$, there exists an $(\mathcal{A}, \varepsilon)$-sample-access-OI predictor $\tilde{p} : \mathcal{X} \to [0, 1]$ implemented by a circuit of size $O\left(s/\varepsilon^2\right)$.*

Proof. Initially, $p^{(0)}$ is a constant function, which Algorithm 1 knows and may access. In each iteration of Algorithm 1, $p^{(t+1)}$ is formed by combining a circuit for $p^{(t)}$ with a circuit computing $\delta_A^{(t)}$. Computing $\delta_A^{(t)}$ requires two calls to A. For sample-access-OI distinguishers, we must pass $p_i^{(t)}$ as input to each copy of the circuit computing A.[14] Leveraging the existing circuit for $p^{(t)}$, we can feed its output wires into the corresponding inputs to A. Thus to compute $p^{(t+1)}$, we require $2d$ wires from the output of $p^{(t)}$ into the distinguishers to compute $\delta_A^{(t)}$, which can be computed in $2s$ gates. Finally, we need to perform a scalar multiplication, a finite-precision addition, and projection onto $[0, 1]$, which by assumption can be handled in $3w \leq d$ gates. Thus, in total, each iteration adds $O(d + s) = O(s)$ new gates. By the bounded iteration complexity of $T = O(1/\varepsilon^2)$, the eventual circuit computing $\tilde{p}$ can be bounded in size by $O(s/\varepsilon^2)$. $\qquad\square$

5.2 Obtaining Oracle-Access-OI

In oracle-access-OI, we allow distinguishers to make oracle calls to the predictor in question $\tilde{p}$. We model these distinguishers by circuits that may include $\tilde{p}$-oracle-gates. We assume that such gates have d labeled input wires, which can be resolved to an $i \in \mathcal{X}$, and output $\tilde{p}_i$. We measure the complexity of such oracle circuits in terms of the size of the circuits and the number of oracle queries. Again, we assume the dimension $d \in \mathbb{N}$ is fixed, so drop the dependence on the d.

Definition 5.3 (Oracle circuits). *For $s, q \in \mathbb{N}$, a family of distinguishers $\mathcal{A}$ is implemented by (s, q)-oracle circuits if for each $A \in \mathcal{A}$, given a predictor $p : \mathcal{X} \to [0, 1]$, the distinguisher A^p is implemented by some size-s circuit that includes at most q p-oracle gates.*

Importantly, because Algorithm 1 builds up a circuit for $\tilde{p}$ iteratively, in each iteration $t \leq O(1/\varepsilon^2)$ the algorithm has access to a circuit computing $p^{(t)}$. Thus, in order to implement each oracle distinguisher $A^{p^{(t)}}$, we can hardwire copies of $p^{(t)}$ into A whenever it makes an oracle call. Overall, we can prove the following bound on the total size of the circuit that computes $\tilde{p}$.

Theorem 5.4. *Let $\mathcal{X} \subseteq \{0,1\}^d$ and $d \leq s, q \in \mathbb{N}$, and suppose $\mathcal{A}$ is a class of distinguishers implemented by (s, q)-oracle circuits. For any choice of Nature's distribution $\mathcal{D}^*$ and $\varepsilon > 0$, there exists an $(\mathcal{A}, \varepsilon)$-oracle-access-OI predictor $\tilde{p} : \mathcal{X} \to [0, 1]$ implemented by a circuit of size $s \cdot q^{O(1/\varepsilon^2)}$.*

---

[14] For no-access-OI, in each iteration, evaluating $\delta_A^{(t)}$ requires two calls to some $A \in \mathcal{A}$, but no calls to $p^{(t)}$. This property allows the circuit computing $\tilde{p}$ to make calls to each $\delta_A^{(t)}$ in parallel, so the depth of the eventual circuit can be bounded by the depth of the circuits computing $\mathcal{A}$ (with small overhead to compute the addition of all of the updates).

Proof. For any class of (s, q)-oracle distinguishers, we show how to construct $\tilde{p}$ as in Algorithm 1 (without oracle gates). This implementation leads to a recurrence relation for $s^{(t)}$ expressing the size of the circuit implementing $p^{(t)}$ for each of the t iterations. Solving the recurrence for $T = O(1/\varepsilon^2)$ yields the stated bound.

To begin, the initial predictor $p^{(0)}$, the constant $1/2$ function, requires a circuit of size $s^{(0)} \leq d \leq s$. Additionally, we note that the predictor $p^{(0)}$ is available to Algorithm 1 in iteration $t = 0$. These two observations form the basis of our recurrence.

Consider iteration $t + 1$ of Algorithm 1. By assumption each distinguisher is implemented by an (s, q)-oracle circuit with $p^{(t)}$-oracle circuits. Importantly, the algorithm has access to a size $s^{(t)}$ circuit implementing $p^{(t)}$. Thus, to evaluate $A^{p^{(t)}}$ for any $A \in \mathcal{A}$, we can implement oracle gates using the explicit circuit for $p^{(t)}$ in a total size of $s + q \cdot s^{(t)}$. If the predictor is updated (i.e., does not terminate) the new predictor $p^{(t+1)}$ requires two calls to the distinguishing circuit $A^{p^{(t)}}$, one call to $p^{(t)}$, a multiplication of $\delta_A^{(t)}$ by a scalar, an addition, and a projection onto $[0, 1]$. By our assumption that the necessary operations can each be computed in w gates for $3w \leq s$, we can express the size of the updated circuit as follows.

$$s^{(t+1)} \leq 2 \cdot \left(s + q \cdot s^{(t)} \right) + s^{(t)} + 3w$$
$$\leq 3s + (2q + 1) \cdot s^{(t)}$$

Solving the recurrence[15], we bound the size of the circuit after $T = O(1/\varepsilon^2)$ iterations needed to compute the $(\mathcal{A}, \varepsilon)$-oracle-access-OI predictor $\tilde{p}$.

$$s^{(T)} \leq 3s \cdot \frac{(2q + 1)^{T+1} - 1}{2q}$$
$$\leq s \cdot q^{O(T)}$$

By the bound on the number of iterations from Lemma 5.1, the bound of $s \cdot q^{O(1/\varepsilon^2)}$ on the circuit complexity of $\tilde{p}$ follows. $\qquad\square$

5.3 Obtaining Code-Access-OI

In all the previous levels of OI, we worked without loss of generality with a distinguisher class of fixed concrete complexity (e.g., size-s circuits). Because code-access-OI considers distinguishers that take the description of $\tilde{p}$ as input, it is most natural to consider a family whose complexity scales with that of $\tilde{p}$. We will consider a distinguishers implemented by a non-uniform family that is parameterized by both the dimension d and the length of the description of $\tilde{p}$, which we denote n. Concretely, we measure the complexity of the distinguishers and the resulting $\tilde{p}$ in terms of the size $s(d, n)$ of the circuits in $\mathcal{A}$.

For computations over the description of circuits to be meaningful, it is important to fix a representation of the description of $\tilde{p}$; we denote this canonical description by $\langle \tilde{p} \rangle$. The proposition follows by using the adjacency list representation of the circuit and applying any constant-stretch pairing function.

---

[15] The solution follows by the Taylor approximation $\frac{1}{1-x} = 1 + x + x^2 + \dots$ taking $x = 2q + 1$.

Proposition 5.5. *Suppose $k, s \in \mathbb{N}$ and $k \leq s$. For any size-s bounded fan-in circuit $c : \{0, 1\}^k \to \{0, 1\}$, there exists a canonical encoding $\langle c \rangle$ of the description of the circuit, and a canonical encoding $\langle \langle c \rangle, x \rangle$ of the description-input pairing, each of length $O(s \log(s))$ bits.*

Throughout our discussion of code-access-OI, we will still assume that the dimension d is fixed; thus, we will assume that the distinguishers are of fixed size $s(d)$ with respect to the dimension, but growing with respect to the length of the encoding n,

$$s(d, n) = s(d) + f(n)$$

for various functions f. As before, we will assume that the base distinguisher size s is at least on the order of the dimension d and cost of operations w, so that $d + w \leq O(s(d))$. Given this assumption, and the simplicity of the initial constant predictor $p^{(0)}$, for all classes of distinguishers we consider here, we may assume a base size $s^{(1)}$ of the predictor $p^{(1)}$ after one iteration of Algorithm 1 as

$$s^{(1)} \leq O(s(d)).$$

In addition to this base case, we can give a generic recurrence for the growth of $p^{(t)}$ in terms of the size of the distinguishers.

Lemma 5.6. *Suppose $\mathcal{A}$ is a class of code-access-OI distinguishers implemented by $s(d, n)$-sized circuits, and let $n(s) = O(s \log(s))$ upper bound the length of the encoding of size-s circuits. For any iteration t of Algorithm 1, let $p^{(t)}$ be the current predictor and $s^{(t)}$ denote the size of its circuit. Then, the predictor size can be recursively bounded as*

$$s^{(t+1)} \leq O\left(s\left(d, n(s^{(t)})\right)\right) + s^{(t)}.$$

Proof. As in the analysis for oracle-access-OI, Algorithm 1 updates $p^{(t)}$ to $p^{(t+1)}$ using two calls to the algorithm that distinguishes $p^{(t)}$ and p^*, one call $p^{(t)}$, and three operations of cost at most w. Thus, we can bound the growth rate of $s^{(t)}$ recursively as

$$s^{(t+1)} \leq 2 \cdot s\left(d, n(s^{(t)})\right) + s^{(t)} + 3w$$
$$\leq O\left(s\left(d, n(s^{(t)})\right)\right) + s^{(t)},$$

where the simplified inequality follows by the assumption that $w \leq O(s(d)) \leq O(s(d, n))$. $\qquad \square$

We can apply Lemma 5.6 to obtain bounds on $s^{(T)}$, under various concrete assumptions about $s(d, n)$. In particular, for different choices of $s(d, n)$, the different levels of OI can be implemented in the framework of code-access-OI. For each level, we recover the bounds on the resulting circuit size up to low order terms (e.g., polylogarithmic factors in the distinguisher size). This blow-up follows as a consequence of the generality of the framework of code-access-OI, and the need to encode $\langle \tilde{p} \rangle$ as input to the distinguisher and decode $\langle \tilde{p} \rangle$ if the distinguisher wants to evaluate $\tilde{p}$.

Sublinear distinguishers. To start, suppose we choose a class of distinguishers of complexity independent of $\tilde{p}$, $s(d, n) = s(d)$. In this case, as $\tilde{p}$ grows in complexity, the distinguishers' access to $\langle\tilde{p}\rangle$ becomes relatively limited. Such distinguishers can still implement no-access-OI tests by ignoring $\langle\tilde{p}\rangle$; we show that the complexity of such predictors grows similarly to the complexity needed to satisfy no-access-OI. In fact, we bound the complexity of a code-access-OI predictor even for distinguishers that have (strongly) sublinear access to $\tilde{p}$.

Theorem 5.7. *Let $\mathcal{X} \subseteq \{0, 1\}^d$ for $d \in \mathbb{N}$. For any fixed constant $\delta \in [0, 1)$, suppose $\mathcal{A}$ is a class of distinguishers implemented by size-$s(d, n)$ circuits for*

$$s(d, n) \leq s(d) + \left(\frac{n}{\log(n)}\right)^{\delta}.$$

For any choice of Nature's distribution $\mathcal{D}^*$ and $\varepsilon > 0$, there exists an $(\mathcal{A}, \varepsilon)$-code-access-OI predictor $\tilde{p} : \mathcal{X} \to [0, 1]$ implemented by a circuit of size $O\left(s(d) \cdot \varepsilon^{-\frac{2}{1-\delta}}\right)$.

Proof. We bound the size of the predictor $\tilde{p}$ produced by Algorithm 1 when using a class of distinguishers with sublinear access to $\langle\tilde{p}\rangle$. Applying Lemma 5.6, we can derive a recurrence for the growth rate of the predictor at each iteration,

$$s^{(t+1)} \leq O\left(s(d) + \left(\frac{n(s^{(t)})}{\log(n(s^{(t)}))}\right)^{\delta}\right) + s^{(t)}$$

$$\leq O(s(d)) + s^{(t)} + b \cdot \left(s^{(t)}\right)^{\delta} \tag{9}$$

for some constant $b > 0$, where (9) follows by Proposition 5.5 bounding the length of the description $\langle p^{(t)}\rangle$ in terms of $s^{(t)}$. For a sufficiently large constant $B > 0$, this recurrence can be bounded by $s^{(T)} \leq B \cdot s(d) \cdot T^{\frac{1}{1-\delta}}$. First, we know $s^{(1)} \leq O(s(d))$. Then, inductively, assume that the claimed recurrence holds for all $t' \leq t$.

$$s^{(t+1)} \leq O(s(d)) + B \cdot s(d) \cdot t^{\frac{1}{1-\delta}} + b \cdot \left(B \cdot s(d) \cdot t^{\frac{1}{1-\delta}}\right)^{\delta}$$

$$\leq O(s(d)) + B \cdot s(d) \cdot t^{\frac{1}{1-\delta}} + b \cdot B^{\delta} \cdot s(d)^{\delta} \cdot t^{\frac{\delta}{1-\delta}}$$

$$\leq B \cdot s(d) \cdot \left(t^{\frac{1}{1-\delta}} + t^{\frac{1}{1-\delta}-1}\right)$$

$$\leq B \cdot s(d) \cdot (t+1)^{\frac{1}{1-\delta}}$$

Recalling that Algorithm 1 terminates after $T = O(1/\varepsilon^2)$ iterations, the resulting circuit for $\tilde{p}$ is bounded by $s^{(T)} \leq O\left(s(d) \cdot \varepsilon^{-\frac{2}{1-\delta}}\right)$ for fixed constant δ. $\square$

Quasilinear distinguishers. Next, we handle distinguishers whose complexity may grow quasilinearly in the size of the description $\langle\tilde{p}\rangle$. Our focus on quasilinear distinguishers is due to the fact that the circuit evaluation problem has quasilinear circuits; thus, using such distinguishers we can hope to simulate sample-access-OI and oracle-access-OI. Specifically, we rely upon the existence of universal circuits of the following form.

41

Theorem 5.8 (Corollary of Theorem 3.1 from [LW12]). *There exists a (polylog-time uniform) universal circuit family $\{C_n\}$ of size $O(n \cdot \operatorname{poly} \log(n))$, such that for any bounded fan-in circuit $c : \{0,1\}^k \to \{0,1\}$ and input $x \in \{0,1\}^k$, where $|\langle\langle c\rangle, x\rangle| = n$,*

$$C_n\left(\langle\langle c\rangle, x\rangle\right) = c(x).$$

By pairing this result with Proposition 5.5, we can bound the size needed to evaluate $\tilde{p}$. Recall that for a circuit p of size s, the description $|\langle p\rangle|$ can be written in $O(s \log(s))$ bits. Thus, in combination, we will assume that given the description $\langle p\rangle$, the circuit p can be evaluated by a distinguisher using $s \cdot (\log(s))^c$ gates for some $c \in \mathbb{N}$.

We analyze the growth rate of the circuit produced by Algorithm 1 if we allow distinguishers that can make q calls to the current model predictor. Specifically, we consider distinguishers of size

$$s(d, n(s)) \leq s(d) + s \cdot (\log(s))^c$$

This growth rate bounds the size of the predictors needed to implement oracle-access-OI within the framework of code-access-OI. As we'll see, the dependence on q and the base distinguisher size $s(d)$ remains roughly the same, with overhead due to encoding and decoding $p^{(t)}$ at each iteration, rather than wiring it directly.

Theorem 5.9. *Let $\mathcal{X} \subseteq \{0,1\}^d$ for $d \in \mathbb{N}$. For a constant $c > 0$, suppose $\mathcal{A}$ is a class of distinguishers implemented by size-$s(d, n)$ circuits for*

$$s(d, n(s)) \leq s(d) + s \cdot (\log(s))^c.$$

For any choice of Nature's distribution $\mathcal{D}^*$ and $\varepsilon \in (0,1)$, there exists an $(\mathcal{A}, \varepsilon)$-code-access-OI predictor $\tilde{p} : \mathcal{X} \to [0,1]$ implemented by a circuit of size $s(d) \cdot q^{O(1/\varepsilon^2)} \cdot (1/\varepsilon)^{O(1/\varepsilon^2)} \cdot (\log(s(d) \cdot q))^{O(1/\varepsilon^2)}$.

Proof. By Lemma 5.6, we can bound the growth rate at the tth iteration as follows

$$s^{(t+1)} \leq O\left(s(d) + q \cdot s^{(t)} \cdot \left(\log(s^{(t)})\right)^c\right) + s^{(t)}$$
$$\leq b \cdot \left(s(d) + q \cdot s^{(t)} \cdot \left(\log(s^{(t)})\right)^c\right)$$

for some constant $b > 0$. For notational convenience, let $\Xi = s(d) \cdot (2c)^{2c}bq$. We will show that the recurrence can be bounded as follows.

$$s^{(T)} \leq s(d) \cdot ((2c)^{2c}bq)^T \cdot ((T-1)!)^{2c} \cdot \left(\log(s(d) \cdot (2c)^{2c}bq)\right)^{cT}$$
$$= s(d) \cdot ((2c)^{2c}bq)^T \cdot ((T-1)!)^{2c} \cdot (\log(\Xi))^{cT}$$

Again, we know that $s^{(1)} \leq O(s(d))$. To see the recurrence, consider the expression for $s^{(t+1)}$

assuming the inductive hypothesis for $s^{(t)}$.

$$s^{(t+1)} \leq b \cdot s(d) + bq \cdot s^{(t)} \cdot \left(\log(s^{(t)})\right)^c$$

$$\leq b \cdot s(d) + bq \cdot \left(s(d) \cdot \left((2c)^{2c}bq\right)^t \cdot ((t-1)!)^{2c} \cdot (\log(\Xi))^{ct}\right)$$

$$\cdot \left(\log\left(s(d) \cdot \left((2c)^{2c}bq\right)\right)^t \cdot ((t-1)!)^{2c} \cdot (\log(\Xi))^{ct}\right)\right)^c$$

$$\leq s(d) \cdot (2c)^{2ct} \cdot (bq)^{t+1} \cdot ((t-1)!)^{2c} \cdot (\log(\Xi))^{ct}$$

$$\cdot (t \cdot \log(\Xi) + 2ct \cdot \log(t) + ct \cdot \log\log(\Xi))^c$$

$$\leq s(d) \cdot (2c)^{2ct} \cdot (bq)^{t+1} \cdot ((t-1)!)^{2c} \cdot (\log(\Xi))^{ct} \cdot (2ct)^{2c} \cdot \log(\Xi)^c$$

$$= s(d) \cdot \left((2c)^{2c}bq\right)^{t+1} \cdot (t!)^{2c} \cdot \log(\Xi)^{c\cdot(t+1)}$$

Taking b and c to be constants, the bound on the recurrence can be simplified and bounded as follows,

$$s^{(T)} \leq s(d) \cdot q^{O(1/\varepsilon^2)} \cdot (1/\varepsilon)^{O(1/\varepsilon^2)} \cdot (\log(s(d) \cdot q))^{O(1/\varepsilon^2)}$$

where $T \leq O(1/\varepsilon^2)$ follows by the iteration complexity of Algorithm 1. $\qquad\square$

Polynomial distinguishers. In the limits of efficient distinguishers, we may allow the computation over $\langle \tilde{p} \rangle$ to grow polynomially. Such distinguishers, in principle, could run nontrivial tests on the description $\langle \tilde{p} \rangle$ itself beyond simply evaluating $\tilde{p}$. We show that predictors satisfying this strong notion of OI exist, in complexity independent of that of p^*, albeit doubly exponential in the number of iterations of Algorithm 1.

Theorem 5.10. *Let $\mathcal{X} \subseteq \{0,1\}^d$ for $d \in \mathbb{N}$. For any constant $k \in \mathbb{N}$, suppose $\mathcal{A}$ is a class of distinguishers implemented by size-$s(d,n)$ circuits for*

$$s(d,n) \leq (s(d) + n)^k.$$

For any choice of Nature's distribution $\mathcal{D}^*$ and $\varepsilon > 0$, there exists an $(\mathcal{A}, \varepsilon)$-code-access-OI predictor $\tilde{p} : \mathcal{X} \to [0,1]$ implemented by a circuit of size $O\left(s(d)2^{O(1/\varepsilon^2)}\right)$.

Proof. Applying Lemma 5.6, we can bound the growth rate of the size of $p^{(t)}$ for all $t \geq 1$ loosely through the following recurrence.

$$s^{(t+1)} \leq \left(s(d) + n(s^{(t)})\right)^k + s^{(t)}$$

$$\leq \left(3 \cdot s^{(t)} \log(s^{(t)})\right)^k$$

$$\leq \left(3 \cdot s^{(t)}\right)^{2k}$$

This recurrence can be bounded as

$$s^{(T)} \leq 3^{((2k)^{T+1}-1)/(2k-1)} \cdot (s(d))^{(2k)^T}$$

for a sufficiently large constant B. To start, $s^{(1)} \leq O(s(d))$. Then, inductively, we bound the growth as follows.

$$s^{(t+1)} \leq 3^{2k} \cdot \left(3^{((2k)^{t+1}-1)/(2k-1)} \cdot (s(d))^{(2k)^t} \right)^{2k}$$
$$\leq 3^{((2k)^{t+2}-1)/(2k-1)} \cdot (s(d))^{(2k)^{t+1}}$$

Again, by the bound on the number of iterations $T = O(1/\varepsilon^2)$ and by the assumption that $k = O(1)$ is a constant, we can simplify the expression for the size of $\tilde{p}$ to $s^{(T)} \leq s(d)^{2^{O(1/\varepsilon^2)}}$. □

5.4 Learning Outcome Indistinguishability Predictors from Samples

While we describe Algorithm 1 primarily to establish the complexity of representing predictors $\tilde{p}$ that satisfy oracle-access-OI and code-access-OI, in principle, it could be implemented as a learning algorithm that works over labeled samples $\{(i, o_i^*)\}$ drawn from nature. We remark on the statistical and computational complexity of learning such predictors.

Bounding the sample complexity. The main statistical cost for implementing Algorithm 1 comes from the need to estimate $\Delta_A^{(t)}$ for each $A \in \mathcal{A}$ at each iteration $t \leq O(1/\varepsilon^2)$. Using standard concentration inequalities and union bounding, we can upper bound the number of samples per fixed iteration needed to estimate $\mathbf{Pr}_{(i,o_i^*) \sim \mathcal{D}^*} \left[A(i, o_i^*; p^{(t)} = 1 \right]$ to accuracy $\Theta(\varepsilon)$. For instance, applying Hoeffding's inequality, we can see that $m_t \geq \Omega\left(\frac{\log(|\mathcal{A}|/\delta)}{\varepsilon^2} \right)$ labeled samples suffice to estimate $\Delta_A^{(t)}$ for each $A \in \mathcal{A}$ with probability at least $1 - \delta$.

Naively, we can take a fresh set of labeled samples for each iteration, so by iteration complexity $T \leq O(1/\varepsilon^2)$, then in total, $m_{1:T} \geq \Omega\left(\frac{\log(|\mathcal{A}|/\delta\varepsilon)}{\varepsilon^4} \right)$ labeled samples suffice. Importantly, we cannot simply union bound over the iterations because the statistics we need to estimate at the tth iteration depend on $p^{(t)}$, and thus, depend on the estimates from earlier iterations. More sophisticated approaches to bounding the sample complexity are possible. For instance, to beat the approach of naive resampling, [HKRR18] apply the machinery for proving generalization in adaptive data analysis via differential privacy [DMNS06, DFH+15a, DFH+15b, DFH+15c, BNS+16, JLP+20], resulting in improved sample complexity.[16]

Bounding the computational complexity. While we bound the iteration complexity efficiently, naively, each iteration requires at least $\Omega(|\mathcal{A}|)$ time to iterate through each $A \in \mathcal{A}$. If we take $\mathcal{A}$ to be an arbitrary class of distinguishers, this complexity bound cannot be improved significantly under natural complexity-theoretic assumptions, even for no-access-OI. In particular, in the context of learning multi-accurate and multi-calibrated predictors, [HKRR18] demonstrated that the search problem for a violated constraint requires agnostic learning, a notoriously hard problem (e.g., [Reg10, KMV08, Fel10]).

On the positive side, [HKRR18] also showed a reverse direction: if $\mathcal{C}$ is taken to be a class of (agnostically) learnable functions, then $(\mathcal{C}, \alpha)$-multi-calibrated predictors are also learnable (in complexity

---

[16] [Kim20] covers the approach of [HKRR18] for establishing generalization in detail.

that scales with the complexity of learning $\mathcal{C}$). By analogy, if we take $\mathcal{A}$ to be a structured class of distinguishers there is hope that the problem of searching for a distinguisher $A \in \mathcal{A}$ with significant advantage may be reducible to an efficient learning problem. Such a reduction would follow from techniques similar to those discussed in [HKRR18, Kim20].

6 Barriers to Efficient Outcome Indistinguishability

We define ensembles of scalably hard functions, and show that they imply the hardness of oracle-access-OI. In Section 6.2 we show a candidate construction of scalably hard functions from clique counting and deduce that hardness of clique counting implies hardness of oracle-access-OI. In Section 6.3 we show results based on the existence of hard functions in PSPACE.

See Section 1.3 for a technical overview of these results, and for a discussion of further candidates based on fine-grained complexity assumptions.

6.1 Scalable Hardness and Hardness of OI

Definition 6.1 (Ensembles of functions, of distributions and of collections thereof). *An ensemble of functions $\{f_n : X_n \to Y_n\}_{n \in \mathcal{N}}$ is* computable *in time $t(n)$ if there exists a Turing machine that, on inputs of the form $\{(1^n, x) : x \in X_n\}$, runs in time $t(n)$ and correctly computes $f_n(x)$. Similarly, an ensemble of distributions $\{D_n\}_{n \in \mathcal{N}}$ is* sampleable *in time $t(n)$ if there exists a (probabilistic) Turing machine that, on input 1^n, runs in time $t(n)$, and whose output distribution is D_n.*

An ensemble of collections *of functions $\mathcal{F} = \{\{f_{i,n} : X_n \to Y_n\}_{i=1}^{m(n)}\}_{n \in \mathcal{N}}$, where the n-th collection is of size $m(n)$, is* computable *in time $t(n)$ if there exists a Turing machine that, on inputs of the form $\{(1^n, i, x) : i \in [1, \ldots, m(n)], x \in X_n\}$, runs in time $t(n)$ and outputs $f_{i,n}(x)$. We say that computing the ensemble* requires randomized *time $\ell(n)$ if it is not computable in randomized time $o(\ell(n))$ (i.e. there is no Turing machine running in this time that computes the function correctly w.h.p. over its coins). Similarly, an ensemble of* collections *of distributions $\mathcal{D} = \{\{D_{i,n}\}_{i=1}^{m(n)}\}_{n \in \mathcal{N}}$, is* sampleable *in time $t(n)$ if there exists a (probabilistic) Turing machine that, on inputs of the form $\{(1^n, i) : i \in [1, \ldots, m(n)]\}$, runs in time $t(n)$, and whose output distribution is $D_{i,n}$.*

Definition 6.2 (Scalable hardness). *Let $\mathcal{F} = \{\{f_{i,n} : X_n \to Y_n\}_{i=1}^{m(n)}\}$ be an ensemble of collections of functions, where $\{X_n, Y_n \subseteq \{0,1\}^{\mathrm{poly}(n)}\}$ and $m(n) = \mathrm{poly}(n)$. We say that the ensemble $\mathcal{F}$ has* scalable hardness *if the following conditions hold:*

- **Complexity bounds.** *The ensemble $\mathcal{F}$ can be computed in time $t_f(n)$, and requires randomized time $\ell_f(n)$.*

- **Downwards self-reduction.** *There is an oracle Turing Machine Q s.t.:*

$$\forall n \in \mathcal{N}, i \in [1, \ldots, m(n)], x \in X_n : Q^{f_{i-1,n}}(1^n, i, x) = f_{i,n}(x),$$

where we define $f_{0,n}$ to be the all-zero function $\bar{0}$ (i.e. for $i = 1$ the oracle does not help Q in its computation). Let $t_Q(n)$ and $q_Q(n)$, which we refer to as the runtime *and* query complexity *(respectively) of $\mathcal{F}$'s downward self-reduction, bound the (worst case) running time and query complexity (respectively) of Q on inputs of the form $(1^n, \cdot, \cdot)$.*

- **Random self-reduction.** *There is an oracle Turing Machine R, an error rate $\mu : \mathcal{N} \to [0,1]$, and an ensemble of collections of distributions $\mathcal{D} = \{\{D_{i,n}\}_{i \in [1,\ldots,m(n)]}\}_{n \in \mathcal{N}}$ that can be sampled in time $t_D(n)$, where each $D_{i,n}$ is over X_n, s.t. for every $n \in \mathcal{N}, i \in [1,\ldots,m(n)]$ and every function $f^* : X_n \to \{0,1\}$, if it holds that $\mathbf{Pr}_{x \sim D_{i,n}}[\ f^*(x) \neq f_{i,n}(x)\ ] < \mu(n)$, then:*

$$\forall x \in X_n : \mathbf{Pr}\left[\ R^{f^*}(1^n, i, x) = f_{i,n}(x)\ \right] \geq 2/3,$$

  where this latter probability is only over the coin tosses of R.

  Let $t_R(n)$ and $q_R(n)$, which we refer to as the runtime and query complexity *of $\mathcal{F}$'s random self-reduction (respectively), bound the (worst case) running time and query complexity (respectively) of R on inputs of the form $(1^n, \cdot, \cdot)$. We refer to $\mu(n)$ as $\mathcal{F}$'s* error rate *and we refer to $t_D(n)$ as $\mathcal{F}$'s* sampling time.

Non-Boolean to Boolean scalable hardness. We will be particularly interested in scalably hard *Boolean* functions, where the range is $\{0,1\}$ (for all $n \in \mathcal{N}$). We use the Goldreich-Levin hardcore predicate [GL89] to transform non-Boolean scalably hard functions into Boolean ones by taking an inner product of the output with a random input string. The degradation in the parameters is polynomial in the output length of the original functions:

Proposition 6.3 (Non-Boolean to Boolean scalable hardness)**.** *Let $\mathcal{F} = \{f_{i,n} : X_n \to \{0,1\}^{y(n)}\}$ be an ensemble of collections of functions with scalable hardness (see Definition 6.2) where each collection in the ensemble is of size $m(n)$ and the range of the functions is $\{0,1\}^{y(n)}$. Let $t_f(n)$ and $\ell_f(n)$ be $\mathcal{F}$'s complexity upper and lower bounds (respectively), let $t_Q(n)$ and $q_Q(n)$ be $\mathcal{F}$'s downward self-reduction's runtime and query complexity (respectively), let $t_R(n)$ and $q_R(n)$ be $\mathcal{F}$'s random self-reduction's runtime and query complexity (respectively), let $t_D(n)$ be $\mathcal{F}$'s sampling time and let $\mu(n)$ be $\mathcal{F}$'s error rate.*

Then there exists an ensemble $\mathcal{G} = \{g_{i,n} : (X_n \times \{0,1\}^{y(n)}) \to \{0,1\}\}$ of collections of Boolean functions with scalable hardness, where each collection is of the same size $m(n)$ and where the following hold:

- *The complexity upper and lower bounds (respectively) of $\mathcal{G}$ are $t_g(n) = O(t_f(n))$ and $\ell_g(n) = \Omega(\ell_f(n)/y(n) \cdot \log(y(n)))$.*

- *The runtime and query complexity of $\mathcal{G}$'s downwards self-reduction are (respectively) $t'_Q(n) = O(t_Q(n) + (q_Q(n) \cdot y^2(n)))$ and $q'_Q(n) = O(q_Q(n) \cdot y(n))$.*

- *The runtime and query complexity of $\mathcal{G}$'s random self-reduction are (respectively) $t'_R(n) = \tilde{O}(t_R(n) + (q_R(n) \cdot y^2(n)))$ and $q'_R(n) = \tilde{O}(q_R(n) \cdot y(n))$. The sampling time of $\mathcal{G}$ is $t'_D(n) = O(t_D(n) + y(n))$ and the error rate is $\mu'(n) = ((1/4 - \alpha) \cdot \mu(n))$ for an arbitrarily small constant $\alpha > 0$.*

Proof. The collection $\mathcal{G}$ is defined by $g_{i,n}(x, r) = \langle x, r \rangle$, where $x \in X_n, r \in \{0,1\}^{y(n)}$ and $\langle \cdot, \cdot \rangle$ is the inner product over GF[2]. The complexity upper bound follows by construction. The lower bound follows because computing $f_{i,n}$ on an input x can be reduced to computing each of its $y(n)$ output bits, and each output bit be computed by a call to $g_{i,n}$ (where we take the majority of $\log(y(n))$ such calls to $g_{i,n}$ to reduce any probability of error).

The downwards self-reduction for $g_{i,n}$ on input (x, r) uses the downwards self-reduction for $f_{i,n}$ on input x. Each call to $f_{i-1,n}$ can be simulated as described above, using $y(n)$ calls to $g_{i-1,n}$. Thus, the increase in the query complexity is a multiplicative $y(n)$ factor, the increase in the runtime is due to the time needed to make these additional queries.

The random self-reduction for $g_{i,n}$ is to a distribution D' obtained by sampling x from $\mathcal{F}$'s "hard" distribution $D_{i,n}$ and appending a uniformly random Boolean string r of length $y(n)$. The sampling time follows by construction. The random self-reduction R' for $g_{i,n}$ gets access to an oracle g^* that computes $g_{i,n}$ correctly with high probability over an input drawn from D'. On input (x, r), R' computes $f_{i,n}$ on x and takes the inner product with r. Computing $f_{i,n}(x)$ is done by applying the random self-reduction for $f_{i,n}$ on input x. The latter reduction needs access to an oracle f^* with error rate at most μ (over inputs drawn from $D_{i,n}$). We use the oracle to g^* to simulate such an oracle. Let $\alpha > 0$ be a constant, and assume that for input $x \in X_n$:

$$\Pr_{r \sim U_{y(n)}} [g^*(x, r) = g_{i,n}(x, r)] \geq 3/4 + \alpha.$$

In this case, we can reconstruct the j-th bit of $f_{i,n}(x)$ with probability greater than half by XORing the values $g^*(x, r)$ and $g^*(x, r \oplus e^{(j)})$ (where r is uniformly random and $e^{(j)}$ is the j-th unit vector). By repeating this $O(\log(y(n)) + \log(q_R(n)))$ times and taking the majority, we reconstruct the j-th bit of $f_{i,n}(x)$ with all but $1/(100q_R(n) \cdot y(n))$ probability. Repeating this for each of the bits and taking a union bound, we recover $f_{i,n}(x)$ with all but $1/100q_R(n)$ probability.

Now, since the error rate of g^* is at most $\mu(1/4 - \alpha)$, we get that:

$$\Pr_{x \sim D_{i,n}} \left[\Pr_{r \sim U_{y(n)}} [g^*(x, r) \neq g_{i,n}(x, r)] > 1/4 - \alpha \right] < \mu.$$

Thus, when R'^{g^*} invokes $f_{i,n}$'s random self-reduction R, it is simulating an oracle f^* that, for all but a μ-fraction of the inputs drawn from $D_{i,n}$, computes $f_{i,n}$ correctly with all but $1/100q_R(n)$ probability over its own coins. By the guarantee of $f_{i,n}$'s random self-reduction, and taking a union bound over the error probability in answering all of R's queries, we conclude that the invocation of R will return the correct answer with probability at least $(2/3 - 1/100)$. The error probability can be reduced to $2/3$ by repeating (a constant number of times).

The number of queries made by the reduction R' is (by construction) $O(q_R(n) \cdot y(n) \cdot (\log(y(n)) + \log(q_R(n))))$. The running time of R' is $O(t_R(n) + (q_R(n) \cdot (\log(y^2(n)) + \log(q_R(n)))))$. □

Hardness of OI from scalable hardness. We show that ensembles of scalably hard Boolean functions imply that OI is hard. Note that by Proposition 6.3 we derive a similar conclusion for ensembles of scalably hard non-Boolean functions. The complexity upper and lower bounds follow by construction.

Theorem 6.4. *Suppose $\mathcal{F}$ is an ensemble of collections of functions with scalable hardness (see Definition 6.2), where the functions are all Boolean (i.e. $Y_n = \{0, 1\}$ for all n), and each collection in the ensemble is of size $m(n)$. Let $t_f(n)$ and $\ell_f(n)$ be $\mathcal{F}$'s complexity upper and lower bounds (respectively), let $t_Q(n)$ and $q_Q(n)$ be $\mathcal{F}$'s downward self-reduction's runtime and query complexity (respectively), let $t_R(n)$ and $q_R(n)$ be $\mathcal{F}$'s random self-reduction's runtime and query complexity*

(respectively), and let $t_D(n)$ be $\mathcal{F}$'s sampling time. Suppose further that $\mathcal{F}$'s error rate $\mu(n)$ is at most 0.06.

Then there exist:

- *an ensemble of predictors $p^* = \{p_n^* : Z_n \to \{0,1\}\}$, where $Z_n \subseteq ([1,\ldots,m(n)] \times X_n)$, that is computable in time $O(t_f(n))$,*

- *an ensemble of distributions $H = \{H_n\}$, which can be sampled in time $O(t_D(n))$, where each H_n is over Z_n,*

- *and an ensemble $\{\mathcal{A}_n\}$ of collections of distinguishers, where the n-th collection is of size $m(n)$, that is computable in time $O\left(t_Q(n) + (t_R(n) \cdot q_Q(n) \cdot \log(q_Q(n)))\right)$,*

*such that for every ensemble of predictors $\{\tilde{p}_n : Z_n \to [0,1]\}$ computable in time $o\left(\frac{\ell_f(n)}{q_R(n)} - t_R(n)\right)$, for infinitely many values of n, the predictor $\tilde{p}_n$ is **not** $(\mathcal{A}_n, \varepsilon(n))$-OI with respect to p_n^* and the distributions H_n, where $\varepsilon(n) = 1/100m(n)$.*

Proof. Fix a value of n, and let $m = m(n)$. We construct $p^* = p_n^*$ and the distribution $H = H_n$ as follows:

- The domain is $Z = Z_n = \{(i,x) : i \in [1,\ldots,m(n)], x \in X_{i,n}\}$, where $X_{i,n}$ is the domain of the i-th function in the collection $\mathcal{F}_n$. We use Z_i to denote the subset of Z where the first coordinate is fixed to i (note that these subsets are disjoint).

- For an input in Z we have $p^*(i,x) = f_{i,n}(x)$, where $f_{i,n}$ is the i-th function in the collection $\mathcal{F}_n$. Note that the range of p^* is Boolean.

- The distribution H (over Z), has a uniform marginal distribution on $i \in [1,\ldots,m]$. Conditioning on a fixed value of i, the conditional distribution on x is $D_i = D_{i,n}$ (the i-th distribution in the collection $\mathcal{D}_n$).

The collection of distinguishers $\{A_1,\ldots,A_m\}$ is described in Figure 2. Each of these distinguishers gets oracle access to a predictor $p : Z \to [0,1]$. On input $(j \in [1,\ldots,m], x \in X_{j,n}, o \in \{0,1\})$, they try to distinguish whether or not o is drawn by p^*.

The bound on the running time of the A_i's follows by construction: A_1 runs in time $t_Q(n)$, simulating $Q(1^n, 1, x)$ and answering all oracle calls using the 0 oracle. For $i \geq 2$, the distinguisher A_i runs the downwards self-reduction, answering each of its oracle queries (which are meant to be answered by $f_{i-1,n}$) by running the random self-reduction $R(1^n, i-1, x)$ multiple times, and taking the majority of its answers (the repetition lowers the error probability, see below). The oracle queries in each invocation of R (which are to inputs in $X_{i-1,n}$) are answered using the given predictor p (restricted to inputs in Z_{i-1}, i.e. with the first coordinate fixed to $(i-1)$), rounding its answers to 0 or 1. By construction, the running time is $O(t_Q(n) + (t_R(n) \cdot q_Q(n) \cdot \log(q_Q(n))))$.

The intuition behind the hardness of achieving OI for the class of distinguishers $\mathcal{A}$ (see also Section 1.3) is that if $\tilde{p}$ is OI and is "correct on average" in computing f_{i-1} over random inputs, then

48

Algorithm 2 DISTINGUISHER $A_i^p(j, x, o)$

---

if $i \neq j$, output 0

otherwise, if $i = 1$, output 1 iff $o = Q^{\bar{0}}(1^n, 1, x)$ `//using the all-0 oracle to compute` f_1

otherwise, if $i \geq 2$:

   for $x' \in X_{i-1,n}$, let:

- $g(x') = \lceil p(i-1, x') \rfloor$ `//rounding` p`'s answers`

- $g'(x') = R^g(1^n, i-1, x')$ `//random self-reduction`

- $g''(x')$ computes the majority answer in $t = O(\log(q_Q(n)))$
  independent executions of g' on x' `//boosting the success probability`

   output 1 iff $o = Q^{g''}(1^n, i, x)$ `//downwards self-reduction to compute` $f_{i,n}(x)$

---

in the distinguisher A_i (which gets oracle access to $\tilde{p}$), the procedure g'', which uses $\mathcal{F}$'s random self reduction, computes f_{i-1} correctly w.h.p. on every input in X_{i-1}. By the downwards self-reducability of $\mathcal{F}$, this implies that for every input (i, x, o), w.h.p. over its coins the distinguisher A_i (with oracle access to $\tilde{p}$) accepts if and only if $o = f_i(n)$. Since it is always true that $o = f_i(n)$ when o is drawn by p^*, the fact that $\tilde{p}$ is OI implies that it should also be true w.h.p. when o is drawn by $\tilde{p}$. When we say a (real-valued) predictor $\tilde{p}$ "correctly computes" a Boolean function like f_i, we mean that when the value outputted by the predictor is *rounded*, the resulting boolean value agrees with f_i.

The distinguisher A_1 is constructed so that output indistinguishability w.r.t. this distinguisher implies that $\tilde{p}$ much be "correct on average" in computing f_1. This is the basis for an inductive argument, which shows that pt must also be "correct on average" in computing f_i for every possible value of i. Plugging (the rounding of) $\tilde{p}$ into the random self-reduction R, we obtain a machine that computes $\mathcal{F}$ correctly for every input (w.h.p. over its coins). Since computing $\mathcal{F}$ requires randomized time $\ell_f(n)$, we conclude that computing $\tilde{p}$ requires time $((\ell_f(n)/q_R(n)) - t_R(n))$. The formal induction argument follows from the following two claims:

Claim 6.5 (induction basis). *For every predictor $\tilde{p} : Z \to [0, 1]$ that satisfies $(\{A_1\}, \varepsilon(n))$-OI w.r.t p^*, it must be the case that:*

$$\Pr_{x \sim D_1} \left[\; \lceil \tilde{p}(1, x) \rfloor = f_{1,n}(x) \; \right] \geq 0.98.$$

Proof of Claim 6.5. The distinguisher A_1 accepts its input (j, x, o) iff $j = 1$ and $o = f_1(x)$. Since $p^*(j, x) = f_j(x)$ we have that:

$$\Pr_{(j,x) \sim H, o^* \sim p^*(j,x)} \left[\; A_1(j, x, o^*) = 1 \; \right] = 1/m(n).$$

If $\tilde{p}$ is $(\{A_1\}, \varepsilon(n))$-OI w.r.t p^*, where $\varepsilon(n) = 1/100m(n)$, we have that:

$$\Pr_{(j,x) \sim H, \tilde{o} \sim \tilde{p}(j,x)} \left[\; A_1(j, x, \tilde{o}) = 1 \; \right] \geq 0.99/m(n).$$

Now, since A_1 rejects when either $j \neq 1$ or $\tilde{o} \neq f_{1,n}(x)$, and since the distribution H is uniformly random over its first coordinate, we conclude that:

$$\Pr_{x \sim D_1, \tilde{o} \sim \tilde{p}(1,x)} [\ \tilde{o} \neq f_{1,n}(x)\ ] < 0.01.$$

In particular, we conclude that:

$$\Pr_{x \sim D_1} [\ |\tilde{p}(1,x) - f_{1,n}(x)| \geq 0.5\ ] < 0.02,$$

and the claim follows. $\qquad\qquad\square$

Claim 6.6 (induction step). *For every predictor $\tilde{p} : Z \rightarrow [0,1]$ and $i \in [2,\ldots,m]$. If $\tilde{p}$ is $(\{A_i\}, \varepsilon(n))$-OI w.r.t p^* and also:*

$$\Pr_{x \sim D_{i-1}} [\ \lceil \tilde{p}(i-1,x) \rceil = f_{i-1,n}(x)\ ] \geq 0.94, \tag{10}$$

then:

$$\Pr_{x \sim D_i} [\ \lceil \tilde{p}(i,x) \rceil = f_{i,n}(x)\ ] \geq 0.94,$$

Proof of Claim 6.6. The condition in Equation (10) implies that $\tilde{p}$ is "correct on-average" for computing f_{i-1}: after rounding, it agrees with f_{i-1} w.h.p. over random inputs from D_{i-1}. When we run the distinguisher A_i with oracle access to $\tilde{p}$, by the random self-reducability of $\mathcal{F}$, we conclude that for each input $x \in X_{i-1}$, the procedure g' computes f_{i-1} correctly with probability $2/3$. The procedure g'', which runs $t = O(\log(q_Q(n)))$ independent invocations of g', thus has an error probability bounded by $1/100 q_Q(n)$. The procedure $Q^{g''}(1^n, i, \cdot)$ runs $\mathcal{F}$'s downwards self-reduction using g'' as an oracle. Taking a union bound over Q's oracle calls, we conclude that the probability that they are all answered correctly is at least 0.99. When this is the case, $Q^{g''}(1^n, i, \cdot)$ correctly computes the function f_i. Thus, we conclude that for any $x \in X_i$, the distinguisher $A_i(i, x, o)$ accepts w.p. at least 0.99 if $o = f_i(x)$, and otherwise it rejects w.p. at least 0.99:

$$\Pr_{x \sim D_i, o^* \sim p^*(i,x)} [\ A_i(i, x, o^*) = 1\ ] \geq 0.99.$$

Since $\tilde{p}$ is $(\{A_i\}, \varepsilon(n))$-OI w.r.t p^*, where $\varepsilon(n) = 1/100m(n)$, and since $A_i(j, x, o)$ always rejects if $j \neq i$ and the distribution H is uniformly random over its first coordinate, it must be the case that:

$$\Pr_{x \sim D_i, \tilde{o} \sim \tilde{p}(i,x)} [\ A_i(i, x, \tilde{o}) = 1\ ] \geq 0.98.$$

Now, since $A_i(i, x, o)$ rejects w.p. at least 0.99 when $o \neq f_{i,n}(x)$, we conclude that:

$$\Pr_{x \sim D_i, \tilde{o} \sim \tilde{p}(i,x)} [\ \tilde{o} \neq f_{i,n}(x)\ ] < 0.03.$$

In particular, we conclude that:

$$\Pr_{x \sim D_i} [\ |\tilde{p}(i,x) - f_{i,n}(x)| \geq 0.5\ ] < 0.06,$$

and the claim follows.

$\square$

$\square$

6.2 Hardness of OI from Clique Counting

In this section we use the clique-counting problem to construct an ensemble of functions with scalable hardness, and derive a hardness result for OI. We describe this result, and conclude the section with a discussion of related work on the complexity of clique counting.

Theorem 6.7 (Scalable hardness from clique counting). *For $k \in \mathcal{N}$, let $t_{cc}(k, n)$ be the upper bound for counting the number of cliques of size k in an n-vertex graph. Suppose further that there is no $o(\ell_{cc}(k, n))$-time randomized algorithm for counting k-cliques.*

Then for every $m \in \mathcal{N}$, there exists an ensemble $\mathcal{F}$ of functions with scalable hardness (see Definition 6.2), where:

- *each collection in the ensemble is of size m, the domain is of size $\log |X_n| = \tilde{O}(n^2)$, and the range is $[n^m]$,*

- *the ensemble can be computed in time $t_f(n) = t_{cc,m}(n)$ and requires time $\ell_{f,m}(n) = \ell_{cc,m}(n)$,*

- *the runtime of $\mathcal{F}$'s downwards self-reduction is $t_Q(n) = \tilde{O}(n^3)$ and its query complexity is $q_Q(n) = n$,*

- *the runtime of $\mathcal{F}$'s random self-reduction is $t_R(n) = \tilde{O}(n^2)$ and its query complexity is $q_R(n) = \text{poly}(\log n)$. $\mathcal{F}$'s sampling time is $\tilde{O}(n^2)$ and the error rate is $1/4 - \varepsilon$ for an arbitrarily small constant $\varepsilon > 0$.*

Proof. In a nutshell, we want $f_{i,n}$, given an (unweighted) n-vertex graph, to output the number of cliques of size i in that graph. Note, however, that we need $f_{i,n}$ to be a Boolean function. We achieve this by taking a hardcore predicate of the clique counting function. In particular, we use the Goldreich-Levin hardcore predicate [GL89], taking an inner product of the count with a random input string. The construction follows.

Taking $\mathcal{G}_n$ to be the set of n-vertex graphs, we define $f_{i,n} : \mathcal{G}_n \to [1, \ldots, \binom{n}{i}]$ to be the number of cliques of size i in a given graph (note that this number is at most $\binom{n}{i}$). The upper and lower bounds on the running time of computing $f_{i,n}$ follow directly from the complexity of clique counting.

Downwards self-reduction. We use the downwards self-reduction for clique counting, which uses the observation that for every vertex v in a graph G, there is a bijection betweeen the i-cliques in G that include the vertex v, and the $(i-1)$-cliques in the graph $G^{(v)}$ of v's neighbors (the graph where we keep edges between v's neighbors and erase all other edges). When we sum the numbers of $(i - 1)$ cliques in the n graphs $G^{(v)}$, each i-clique in G is counted exactly i times. Dividing the sum by j gives the number of j-cliques in G:

$$i \cdot f_{i,n}(G) = \sum_{v \in [n]} f_{i-1,n}(G^{(v)}),$$

where $G^{(v)}$ is the graph induced by vertex v's neighbors (we identify the set of vertices with $[n]$). The downwards self-reduction runs in time $\tilde{O}(n^3)$ and makes n oracle queries.

Random self-reduction. We use the random self-reduction for clique counting [GR18], the claimed bounds follow from their result:

Theorem 6.8 ([GR18] Theorem 1.1, worst-case to average-case reduction for counting cliques)**.** *For any constant t, there exists a distribution $D_{t,n}$ on n-vertex graphs and a $\tilde{O}(n^2)$-time worst-case to average-case reduction of counting t-cliques in n-vertex graphs to counting t-cliques in graphs generated according to this distribution, such that the reduction outputs the correct value with probability $2/3$ provided that the error rate* (of the average-case solver) *is a constant smaller than one fourth. Furthermore, the reduction makes* $\mathrm{poly}(\log n)$ *queries, and the distribution $D_{t,n}$ can be sampled in $\tilde{O}(n^2)$-time and is uniform on a set of $\exp(\tilde{\Omega}(n^2))$ graphs.*

$\square$

We remark that recent works of Boix-Adsera, Brennan and Bressler [BBB19], Goldreich [Gol20], and Dalirrooyfard, Lincoln and Vassilevska Williams [DLW20] show other worst-case to average-case reductions for clique counting and other fine-grained complexity problems. However, the reductions in these works require sub-constant error rates, and thus it is not clear that our construction can be instantiated using these results.

Hardness of OI. Putting together Theorem 6.4 (which gives scalably hard functions with polylog output length) and Proposition 6.3 (which shows how to go from non-Boolean to Boolean scalable hardness) together, we derive a Boolean scalably hard function collection (the degradataion in the parameters is poly-logarithmic). Plugging this collection into Theoream 6.4, we obtain p^* for which OI is hard, based on the complexity of counting cliques in a graph. The resulting corollary is stated below.

Corollary 6.9. *For $k \in \mathcal{N}$, let $t_{cc}(k, n)$ be an upper bound for the time complexity of counting the number of cliques of size k in an n-vertex graph. Suppose further that there is no $o(\ell_{cc}(k, n))$-time randomized algorithm for counting k-cliques.*

Then for every $m \in \mathcal{N}$, there exist:

- *an ensemble of predictors $p^* = \{p_n^* : Z_n \to \{0, 1\}\}$, where the domain consists of strings of length $\tilde{O}(n^2)$, and p^* is computable in time $\tilde{O}(t_{cc}(m, n))$,*

- *an ensemble of distributions $H = \{H_n\}$, which can be sampled in quasi-linear time (i.e. time $\tilde{O}(n^2)$), where each H_n is over Z_n,*

- *and an ensemble $\{\mathcal{A}_n\}$ of collections of distinguishers, where each collection is of size m, that is computable in time $\tilde{O}(n^3)$,*

*such that for every ensemble $\{\tilde{p}_n : Z_n \to [0, 1]\}$ of predictors that are computable in time $\left(\ell_{cc}(m, n) \cdot \log^{-\omega(1)}(n)\right)$, for infinitely many values of n, the predictor $\tilde{p}_n$ is **not** $(\mathcal{A}_n, \varepsilon(n))$-OI with respect to p_n^* and the distributions H_n, where $\varepsilon(n) = 1/100m$.*

The complexity of clique counting. The complexity of clique counting, and of clique detection (which is no harder), has been widely studied and related to other computational problems. Chen *et al.* [CCF+05] and Chen *et al.* [CHKX06] show that detecting k-cliques in $n^{o(k)}$ time would refute the Exponential Time Hypothesis (ETH) [IP01], and that counting k-cliques in $n^{o(k)}$-time would refute #ETH. The clique detection problem is complete for the parameterized complexity class $W[1]$ (see Downey and Flum [DF99]), and the clique counting problem is complete for $\#W[1]$ (Flum and Grohe [FG04]). It has been conjectured that the fastest known algorithms for clique counting and detection are tight: in particular, that there is no $n^{\omega k - o(1)}$-time algorithm for detecting $3k$-cliques (where ω is the matrix multiplication exponent). Abboud, Backurs and Vassilevska Williams [ABW18] showed several implications of this conjecture. For constant k, Abboud, Lewi and Williams [ALW14] show that the k-SUM conjecture implies a $n^{\lceil k/2 \rceil - o(1)}$ lower bound for k-clique detection. Abboud [Abb19] shows that detecting k-cliques in $O(n^{\varepsilon k / \log k})$-time would refute the Set-Cover Conjecture.

Under the conjecture (see above) that $3k - cliques$ cannot be counted in randomized time $n^{\omega k - o(1)}$, we obtain (for the same k) a distribution $\mathcal{D}^*$ of nature that can be sampled in time $O(n^{k\omega})$, and a collection $\mathcal{A}$ of $3k$ distinguishers, which all run in $O(n^3)$ time, so that there is no $(\mathcal{A}, 1/300k)$-oracle-access-OI predictor whose running time is smaller than $n^{\omega k}$ by more than a polylogarithmic factor.

6.3   Scalable Hardness from $\mathrm{BPP} \neq \mathrm{PSPACE}$

In this section we construct an ensemble of functions with scalable hardness that are PSPACE-hard to compute. This construction is based on the PSPACE complete problem of Trevisan and Vadhan [TV07] (which is both random self-reducible and downwards self-reducible). Under the assumption that $\mathrm{BPP} \neq \mathrm{PSPACE}$, we derive a hardness result for OI. We note that in this result (unlike our hardness results that are based on fine-grained complexity), Natures predictor p^* is not poly-time compuable (alternatively, Nature's distribution $\mathcal{D}^*$ cannot be sampled in polynomial time). We also note the hard-to-obtain indistinguishability advantage is polynomially small (rather than a constant in the aforementioned results).

Theorem 6.10 (Scalable hardness from $\mathrm{BPP} \neq \mathrm{PSPACE}$ [TV07], Lemma 4.1). *Suppose that $\mathrm{BPP} \neq \mathrm{PSPACE}$. Then there exists an ensemble $\mathcal{F}$ of functions with scalable hardness (see Definition 6.2), where:*

- *each collection in the ensemble is of size $m(n) = \mathrm{poly}(n)$, the domain and the range are of size $\exp(\mathrm{poly}(n))$,*

- *the ensemble can be computed in $\exp(\mathrm{poly}(n))$ time and requires super-polynomial time in n,*

- *the runtime and query complexity of $\mathcal{F}$'s downwards self-reduction are $\mathrm{poly}(n)$,*

- *the runtime and query complexity of $\mathcal{F}$'s random self-reduction are $\mathrm{poly}(n)$. $\mathcal{F}$'s sampling time is $\mathrm{poly}(n)$ and the error rate is is $1/4 - \varepsilon$ for an arbitrarily small constant $\varepsilon > 0$.*

We note that Trevisan and Vadhan [TV07] only claimed a random self-reduction with error rate $1/\mathrm{poly}(n)$. The error rate claimed above (and, indeed, error rates that are very close to 1) is implied by the worst-case to average-case (or rare-case) reduction of Sudan, Trevisan and Vadhan [STV01].

Putting together Theorem 6.10 and Proposition 6.3, we derive a Boolean scalably hard function collection (the degradataion in the parameters is polynomial). Plugging this collection into Theoream 6.4, we obtain p^* for which OI is hard, assuming BPP $\neq$ PSPACE. The resulting corollary is stated below.

Corollary 6.11. *Suppose that* BPP $\neq$ PSPACE. *Then there exist:*

- *an ensemble of predictors $p^* = \{p_n^* : Z_n \to \{0, 1\}\}$, where the domain consists of strings of length* $\mathrm{poly}(n)$, *and p^* is computable in time* $\exp(\mathrm{poly}(n))$,

- *an ensemble of distributions $H = \{H_n\}$, which can be sampled in* $\mathrm{poly}(n)$ *time, where each H_n is over Z_n,*

- *and an ensemble $\{\mathcal{A}_n\}$ of collections of distinguishers, where each collection is of size $m(n) = \mathrm{poly}(n)$, that is computable in time* $\mathrm{poly}(n)$,

such that for every ensemble $\{\tilde{p}_n : Z_n \to [0, 1]\}$ of predictors that are computable in time $\mathrm{poly}(n)$, *for infinitely many values of n, the predictor $\tilde{p}_n$ is* **not** *$(\mathcal{A}_n, \varepsilon(n))$-OI with respect to p_n^* and the distributions H_n, where $\varepsilon(n) = 1/m(n)$.*

References

[Abb19]   Amir Abboud. Fine-grained reductions and quantum speedups for dynamic programming. In Christel Baier, Ioannis Chatzigiannakis, Paola Flocchini, and Stefano Leonardi, editors, *46th International Colloquium on Automata, Languages, and Programming, ICALP 2019, July 9-12, 2019, Patras, Greece*, volume 132 of *LIPIcs*, pages 8:1–8:13. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019.

[ABW18]   Amir Abboud, Arturs Backurs, and Virginia Vassilevska Williams. If the current clique algorithms are optimal, so is valiant's parser. *SIAM J. Comput.*, 47(6):2527–2555, 2018.

[AGSV19]   Brian Axelrod, Shivam Garg, Vatsal Sharan, and Gregory Valiant. Sample amplification: Increasing dataset size even when learning is impossible. *arXiv preprint arXiv:1904.12053*, 2019.

[ALW14]   Amir Abboud, Kevin Lewi, and Ryan Williams. Losing weight by gaining edges. In Andreas S. Schulz and Dorothea Wagner, editors, *Algorithms - ESA 2014 - 22th Annual European Symposium, Wroclaw, Poland, September 8-10, 2014. Proceedings*, volume 8737 of *Lecture Notes in Computer Science*, pages 1–12. Springer, 2014.

[BBB19]   Enric Boix-Adserà, Matthew Brennan, and Guy Bresler. The average-case complexity of counting cliques in erdos-rényi hypergraphs. In David Zuckerman, editor, *60th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2019, Baltimore, Maryland, USA, November 9-12, 2019*, pages 1256–1280. IEEE Computer Society, 2019.

[BL19]   Avrim Blum and Thodoris Lykouris. Advancing subgroup fairness via sleeping experts. *arXiv preprint arXiv:1909.08375*, 2019.

[BM07]   Avrim Blum and Yishay Mansour. From external to internal regret. *Journal of Machine Learning Research*, 8(Jun):1307–1324, 2007.

[BNS+16]   Raef Bassily, Kobbi Nissim, Adam Smith, Thomas Steinke, Uri Stemmer, and Jonathan Ullman. Algorithmic stability for adaptive data analysis. In *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing*, pages 1046–1059. ACM, 2016.

[BRSV17]   Marshall Ball, Alon Rosen, Manuel Sabin, and Prashant Nalini Vasudevan. Average-case fine-grained hardness. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, pages 483–496, 2017.

[BRSV18]   Marshall Ball, Alon Rosen, Manuel Sabin, and Prashant Nalini Vasudevan. Proofs of work from worst-case assumptions. In Hovav Shacham and Alexandra Boldyreva, editors, *Advances in Cryptology - CRYPTO 2018 - 38th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2018, Proceedings, Part I*, volume 10991 of *Lecture Notes in Computer Science*, pages 789–819. Springer, 2018.

[CCF+05]   Jianer Chen, Benny Chor, Mike Fellows, Xiuzhen Huang, David W. Juedes, Iyad A. Kanj, and Ge Xia. Tight lower bounds for certain parameterized np-hard problems. *Inf. Comput.*, 201(2):216–231, 2005.

[CHKX06]   Jianer Chen, Xiuzhen Huang, Iyad A. Kanj, and Ge Xia. Strong computational lower bounds via parameterized complexity. *J. Comput. Syst. Sci.*, 72(8):1346–1367, 2006.

[Daw82]   AP Dawid. Objective probability forecasts'. Technical report, Research Report 14, Department of Statistical Science, University College London, 1982.

[Daw15]   Philip Dawid. On individual risk. *Synthese*, 194(9):3445–3474, Nov 2015.

[DF99]   Rodney G. Downey and Michael R. Fellows. *Parameterized Complexity*. Monographs in Computer Science. Springer, 1999.

[DFH+15a]   Cynthia Dwork, Vitaly Feldman, Moritz Hardt, Toni Pitassi, Omer Reingold, and Aaron Roth. Generalization in adaptive data analysis and holdout reuse. In *Advances in Neural Information Processing Systems*, pages 2350–2358, 2015.

[DFH+15b]   Cynthia Dwork, Vitaly Feldman, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Aaron Roth. The reusable holdout: Preserving validity in adaptive data analysis. *Science*, 349(6248):636–638, 2015.

[DFH⁺15c] Cynthia Dwork, Vitaly Feldman, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Aaron Leon Roth. Preserving statistical validity in adaptive data analysis. In *Proceedings of the forty-seventh annual ACM symposium on Theory of computing*, pages 117–126. ACM, 2015.

[DHP⁺12] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. Fairness through awareness. In *Proceedings of the 3rd innovations in theoretical computer science conference*, pages 214–226, 2012.

[DKR⁺19] Cynthia Dwork, Michael P. Kim, Omer Reingold, Guy N. Rothblum, and Gal Yona. Learning from outcomes: Evidence-based rankings. In *2019 IEEE 60th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 106–125. IEEE, 2019.

[DLW20] Mina Dalirrooyfard, Andrea Lincoln, and Virginia Vassilevska Williams. New techniques for proving fine-grained average-case hardness. *CoRR*, abs/2008.06591, 2020.

[DMNS06] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of cryptography conference*, pages 265–284. Springer, 2006.

[DN92] Cynthia Dwork and Moni Naor. Pricing via processing or combatting junk mail. In Ernest F. Brickell, editor, *Advances in Cryptology - CRYPTO '92, 12th Annual International Cryptology Conference, Santa Barbara, California, USA, August 16-20, 1992, Proceedings*, volume 740 of *Lecture Notes in Computer Science*, pages 139–147. Springer, 1992.

[Fel10] Vitaly Feldman. Distribution-specific agnostic boosting. In *Proceedings of the First Symposium on Innovations in Computer Science'10*, 2010.

[FG04] Jörg Flum and Martin Grohe. The parameterized complexity of counting problems. *SIAM J. Comput.*, 33(4):892–922, 2004.

[FL99] Drew Fudenberg and David K Levine. An easier way to calibrate. *Games and economic behavior*, 29(1-2):131–137, 1999.

[FS97] Yoav Freund and Robert E Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences*, 55(1):119–139, 1997.

[FV98] Dean P Foster and Rakesh V Vohra. Asymptotic calibration. *Biometrika*, 85(2):379–390, 1998.

[GL89] Oded Goldreich and Leonid A Levin. A hard-core predicate for all one-way functions. In *Proceedings of the twenty-first annual ACM symposium on Theory of computing*, pages 25–32, 1989.

[Gol06] Oded Goldreich. *Foundations of Cryptography: Volume 1, Basic Tools*. Cambridge University Press, USA, 2006.

[Gol08] Oded Goldreich. *Computational Complexity: A Conceptual Perspective*. Cambridge University Press, USA, 1 edition, 2008.

[Gol09]   Oded Goldreich. *Foundations of Cryptography: Volume 2, Basic Applications.* Cambridge University Press, 2009.

[Gol20]   Oded Goldreich. On counting t-cliques mod 2. *Electron. Colloquium Comput. Complex.*, 27:104, 2020.

[GPAM+14]   Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[GR18]   Oded Goldreich and Guy N. Rothblum. Counting t-cliques: Worst-case to average-case reductions and direct interactive proof systems. In Mikkel Thorup, editor, *59th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2018, Paris, France, October 7-9, 2018*, pages 77–88. IEEE Computer Society, 2018.

[Hau92]   David Haussler. Decision theoretic generalizations of the pac model for neural net and other learning applications. *Information and computation*, 100(1):78–150, 1992.

[HKRR18]   Úrsula Hébert-Johnson, Michael P. Kim, Omer Reingold, and Guy Rothblum. Multi-calibration: Calibration for the (computationally-identifiable) masses. In *International Conference on Machine Learning*, pages 1939–1948, 2018.

[HPS16]   Moritz Hardt, Eric Price, and Nati Srebro. Equality of opportunity in supervised learning. In *Advances in neural information processing systems*, pages 3315–3323, 2016.

[IP01]   Russell Impagliazzo and Ramamohan Paturi. On the complexity of k-sat. *J. Comput. Syst. Sci.*, 62(2):367–375, 2001.

[JLP+20]   Christopher Jung, Changhwa Lee, Mallesh M Pai, Aaron Roth, and Rakesh Vohra. Moment multicalibration for uncertainty estimation. *arXiv preprint arXiv:2008.08037*, 2020.

[KGZ19]   Michael P. Kim, Amirata Ghorbani, and James Zou. Multiaccuracy: Black-box post-processing for fairness in classification. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, pages 247–254, 2019.

[Kim20]   Michael Pum-Shin Kim. *A Complexity-Theoretic Perspective on Fairness.* PhD thesis, Stanford University, 2020.

[KMR16]   Jon Kleinberg, Sendhil Mullainathan, and Manish Raghavan. Inherent trade-offs in the fair determination of risk scores. *arXiv preprint arXiv:1609.05807*, 2016.

[KMV08]   Adam Tauman Kalai, Yishay Mansour, and Elad Verbin. On agnostic boosting and parity learning. In *Proceedings of the fortieth annual ACM symposium on Theory of computing*, pages 629–638. ACM, 2008.

[KNRW18]   Michael Kearns, Seth Neel, Aaron Roth, and Zhiwei Steven Wu. Preventing fairness gerrymandering: Auditing and learning for subgroup fairness. In *International Conference on Machine Learning*, pages 2564–2572, 2018.

[KNRW19]   Michael Kearns, Seth Neel, Aaron Roth, and Zhiwei Steven Wu. An empirical study of rich subgroup fairness for machine learning. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 100–109, 2019.

[KRR18]   Michael P. Kim, Omer Reingold, and Guy N. Rothblum. Fairness through computationally-bounded awareness. *Advances in Neural Information Processing Systems*, 2018.

[KS94]   Michael J Kearns and Robert E Schapire. Efficient distribution-free learning of probabilistic concepts. *Journal of Computer and System Sciences*, 48(3):464–497, 1994.

[KSS94]   Michael J Kearns, Robert E Schapire, and Linda M Sellie. Toward efficient agnostic learning. *Machine Learning*, 17(2-3):115–141, 1994.

[KW14]   Diederik P Kingma and Max Welling. Auto-encoding variational bayes, 2014.

[Lip89]   Richard J. Lipton. New directions in testing. In Joan Feigenbaum and Michael Merritt, editors, *Distributed Computing And Cryptography, Proceedings of a DIMACS Workshop, Princeton, New Jersey, USA, October 4-6, 1989*, volume 2 of *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, pages 191–202. DIMACS/AMS, 1989.

[LW94]   Nick Littlestone and Manfred K Warmuth. The weighted majority algorithm. *Information and computation*, 108(2):212–261, 1994.

[LW12]   Richard J Lipton and Ryan Williams. Amplifying circuit lower bounds against polynomial time with applications. In *2012 IEEE 27th Conference on Computational Complexity*, pages 1–9. IEEE, 2012.

[Reg10]   Oded Regev. The learning with errors problem. *Invited Survey in CCC*, 2010.

[San03]   Alvaro Sandroni. The reproducible properties of correct forecasts. *International Journal of Game Theory*, 32(1):151–159, 2003.

[SCM20]   Eliran Shabat, Lee Cohen, and Yishay Mansour. Sample complexity of uniform convergence for multicalibration. *arXiv preprint arXiv:2005.01757*, 2020.

[SSBD14]   Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

[SSV03]   Alvaro Sandroni, Rann Smorodinsky, and Rakesh V Vohra. Calibration with many checking rules. *Mathematics of operations Research*, 28(1):141–153, 2003.

[STV01]   Madhu Sudan, Luca Trevisan, and Salil P. Vadhan. Pseudorandom generators without the XOR lemma. *J. Comput. Syst. Sci.*, 62(2):236–266, 2001.

[TTV09]   Luca Trevisan, Madhur Tulsiani, and Salil Vadhan. Regularity, boosting, and efficiently simulating every high-entropy distribution. In *2009 24th Annual IEEE Conference on Computational Complexity*, pages 126–136. IEEE, 2009.

[TV07]   Luca Trevisan and Salil P. Vadhan. Pseudorandomness and average-case complexity via uniform reductions. *Comput. Complex.*, 16(4):331–364, 2007.

[Vad12]   Salil P. Vadhan. *Pseudorandomness.* Now Publishers Inc., Hanover, MA, USA, 2012.

[Val84]   Leslie G Valiant. A theory of the learnable. *Communications of the ACM*, 27(11):1134–1142, 1984.

[ZME20]   Shengjia Zhao, Tengyu Ma, and Stefano Ermon. Individual calibration with randomized forecasting. *arXiv preprint arXiv:2006.10288*, 2020.