Multicalibrated Partitions for Importance Weights, with Application to KL Divergence Estimation

Parikshit Gopalan*, Nina Narodytska†, Omer Reingold‡, Vatsal Sharan§, Udi Wieder*
* Apple † VMware Research ‡ Stanford University § University of Southern California

Abstract

The ratio between the probability that two distributions assign to points in the domain are called importance weights or density ratios and they play a fundamental role in machine learning and information theory. However, there are strong lower bounds known for point-wise accurate estimation of density ratios, and most theoretical guarantees require strong assumptions about the distributions. We motivate the problem of seeking accuracy guarantees for the distribution of importance weights conditioned on sub-populations belonging to a family $\mathcal{C}$ of subsets of the domain. We formulate *sandwiching bounds* for sets: upper and lower bounds on the expected importance weight conditioned on a set; as a notion of set-wise accuracy for importance weights. We argue that they capture intuitive expectations about importance weights, and are not subject to the strong lower bounds for point-wise guarantees. We introduce the notion of approximately multi-calibrated partitions for a class $\mathcal{C}$, inspired by recent work on multi-calibration in supervised learning [1] and show that the importance weights resulting from such partitions do satisfy sandwiching bounds. In contrast, we show that importance weights returned by popular algorithms in the literature may violate the sandwiching bounds. We present an efficient algorithm for constructing multi-calibrated partitions, given a weak agnostic learner for the class $\mathcal{C}$.

Index Terms

Importance weighting, agnostic learning, algorithmic fairness, maximum entropy

I. Introduction

Given two distributions P and R over a domain $\mathcal{X}$, the ratio $w^*(x) = R(x)/P(x)$ defines a function $w : \mathcal{X} \to \mathbb{R}$ known as the density-ratio or importance weights of R relative to P. We consider the problem of estimating this function given access to random samples from P and R.

This problem has been studied by several communities (often under different monikers) and has a wide variety of applications. In machine learning, it arises both in unsupervised problems such as anomaly detection and supervised settings such as domain adaptation. For anomaly detection in the inlier-based [2, 3] or semi-supervised model [4], P represents normal points or inliers, and R represents our observations. The goal is to find points x where $P(x)$ is small relative to $R(x)$; essentially the points where $w^*(x)$ is high [5, 6, 2, 7]. In domain adaptation, we get labelled training data from a distribution P but are interested in good prediction accuracy under a different test distribution R for which we only observe unlabelled data. The importance weights of R under P can be used to reweigh the loss function to correct for the distributional shift [8, 9, 10, 11, 12, 13]. In information theory, importance weight estimation (aka Radon-Nikodym derivative estimation), is applied to estimate various divergences such as the KL divergence and Renyi divergences between the distributions P and R [14, 15, 16, 17, 18]. These divergence measures themselves find several applications, such as two-sample tests for distinguishing between two distributions [19] and for independence testing [20]. In econometrics and statistics, importance weights (under the guise of propensity scores) play a major role in the theory of causal inference from observational data [21]. For further details and applications, we refer the reader to the book [3].

As a counter-weight to these applications, there are strong information-theoretic lower bounds known. The problem of estimating importance weights from samples is known to be hard for arbitrary P and R, as are applications like domain adaptation and divergence estimation [13, 22]. These lower bounds stem from sample complexity results on testing distributions, where given samples from both P and R we must decide whether $P = R$ or if they are far apart in statistical distance [22, 23]. These results imply that computing any reasonable point-wise approximation to w^* requires sample complexity exponential in the dimension of $\mathcal{X}$.

There are several algorithms known for density-ratio estimation in the literature based on moment matching, logistic regression, the Kullback-Liebler information estimation procedure (KLIEP) and more [3]. Many of these estimate importance weights by learning some distribution Q whose importance weight function w relative to P belongs to a certain parametric family. They choose the parameters to minimize some divergence between the distribution R and the model Q. For appropriate choices of divergence, this paradigm captures several popular approaches to density-ratio estimation including all those mentioned above [24]. If the true importance weight function w^* happens to lie in the parametric family, the algorithm will converge to the right answer. In the realistic non-realizable setting where w^* is not in the family, the algorithm finds the *best approximation* from the family, with quality measured by the chosen divergence. However, it is unclear how good this best approximation is, and the known guarantees are not always meaningful in some of the aforementioned applications.

Consider for example a researcher analyzing data about a disease outbreak in a county, where each point represents demographic and medical information about an individual. They model the data collected from patients as a distribution R, the general population in the county as another distribution P, and build a model w of the importance weights. Their goal is understanding the vulnerability of certain sub-populations that are represented by conjunctions of attributes.

1) Let C be a sub-population such that $R(C)/P(C) > 10$, so that the prevalence within C is 10 times higher than was expected based on the prior. The researcher would like a random sample of datapoints drawn from R conditioned on being in the set C to be assigned large weight by w, ideally at least 10. If not, w might not alert them to the increased prevalence within C.
2) Let C' be a sub-population such that a random sample of datapoints drawn from R conditioned on being in C' is assigned an average weight of 10 by w. Does this mean that the true importance weights are large in expectation for R conditioned on C'? Or might having large weights for C' be a false alarm?

In analogy to proof systems [25, Chapter 8], these conditions ask for **completeness** and **soundness** of the importance weights w respectively. Completeness requires that if a set C is important under R, then it receives large weights w on average under R. Soundness requires that if the average weight under R assigned to a set C' is large, this indicates that the set is important under R. Requiring such guarantees is natural from a group fairness perspective, especially in applications like anomaly detection. On one hand, (with the right formulation) set-wise guarantees do not imply strong point-wise guarantees, hence known lower bounds do not apply. On the other hand, we will show that (perhaps surprisingly) popular algorithms in the literature cannot give such guarantees. At a high level, these algorithms find the model which minimizes expected loss, and might not necessarily capture the behavior on sub-populations accurately.

a) Our Contributions: We summarize the main contributions of this paper briefly.

- We propose requiring set-wise accuracy guarantees for a class $\mathcal{C}$ of sets that include sub-populations we care about. We formulate Sandwiching bounds as a notion of set-wise accuracy for importance weights, and show that they capture the **completeness** and **soundness** for importance weights.
- We introduce the notion of **approximately multi-calibrated partitions** for a class $\mathcal{C}$, inspired by recent work on multi-calibration in supervised learning [1]. We show that the importance weights resulting from such partitions do satisfy sandwiching bounds.
- We present an efficient algorithm for constructing such multi-calibrated partitions, given a **weak agnostic learner** for the class $\mathcal{C}$, which adapts the Boosting by branching programs algorithm [26] to importance weight estimation.
- We show that importance weights returned by popular algorithms such as log-linear KLIEP [27, 3] (equivalently MaxEnt [28, 29, 30]) may violate the Sandwiching bounds, by constructing explicit examples.

b) Notation.: We use capitals $(P, Q, R, \cdots)$ to denote distributions and boldface $\mathbf{x}, \mathbf{y}, \cdots$ to denote random variables. We use $\mathbf{x} \sim P$ to denote sampling $\mathbf{x}$ according to distribution P. For $A \subseteq \mathcal{X}$, let $P(A) = \Pr_{\mathbf{x}\sim P}[\mathbf{x} \in A]$ and $P|_A$ denote P conditioned on A. To every distribution $Q(x)$, one can associate an importance weight function relative to P by $w(x) = Q(x)/P(x)$, we denote this by $Q = w \cdot P$. For any $A \subseteq \mathcal{X}$ observe that

$$\mathop{\mathbb{E}}_{\mathbf{x}\sim P|_A}[w(\mathbf{x})] = \sum_{x\in A} \frac{P(x)}{P(A)} \frac{Q(x)}{P(x)} = \frac{Q(A)}{P(A)}. \tag{1}$$

In our setting, we have a *target* distribution R and a *prior* distribution P, where $R = w^* \cdot P$. Our aim is to find importance weights w such that $Q = w \cdot P$ is a good model for R. We will consider a family of $\mathcal{C} = \{C \subseteq \mathcal{X}\}$ of subsets which include the sub-populations for which we desire guarantees. The collection can be infinite, and contain overlapping subsets, for instance taking $\mathcal{C}$ to be all decision trees or neural nets of a given size lets us capture sub-populations which are conjunctions of attributes. For our algorithmic results, we assume an efficient algorithm to learn the class of indicator functions of $C \in \mathcal{C}$, formally that $\mathcal{C}$ is *weakly agnostically learnable* (see Definition V.1).

II. Setwise guarantees and sandwiching bounds

We now rigorously formulate a notion called *sandwiching bounds* which formally capture the completeness and soundness requirements. We desire guarantees for a collection of sets $\mathcal{C} = \{C \subseteq \mathcal{X}\}$ that include the sub-populations of interest. While our definitions make sense for arbitrary $\mathcal{C}$, our algorithmic results require the indicators of the sets to be efficiently weakly agnostically learnable. For a distribution R and a set C, let $R|_C$ denote the distribution R conditioned on C. Ideally[1] we would like the following **strict Sandwiching bounds** to hold for every $C \in \mathcal{C}$:

$$\mathop{\mathbb{E}}_{\mathbf{x}\sim P|_C}[w^*(\mathbf{x})] \le \mathop{\mathbb{E}}_{\mathbf{x}\sim R|_C}[w(\mathbf{x})] \le \mathop{\mathbb{E}}_{\mathbf{x}\sim R|_C}[w^*(\mathbf{x})]. \tag{2}$$

[1] In practice we allow a bit of slack, see Theorem III.4.

The quantity in the middle is one that we can compute from random samples of R, given w. We want it to be sandwiched between the expectations of the ground-truth scores w^* under $P|_C$ and $R|_C$ (note that we do not have access to w^* explicitly). When $P|_C$ and $R|_C$ are identical, the upper and lower bounds in Equation (2) are equal. But in general, they could be far apart.

Let us see why sandwiching bounds indeed capture the aforementioned requirements. For the lower bound, observe that by Equation (1), $\mathbb{E}_{\mathbf{x}\sim P|_C}[w^*(\mathbf{x})] = R(C)/P(C)$ hence this inequality captures condition (1). The upper bound can be interpreted as saying that we want our weights to be conservative, they should not exaggerate the prevalence of anomalies within a set. This captures the soundness requirement in condition (2), if we were to replace the learned weights w by the ground truth w^*, the average weights would only increase.

Equation (2) implies the following outer inequality for w^* (independent of the model w):

$$\mathop{\mathbb{E}}_{\mathbf{x}\sim P|_C}[w^*(\mathbf{x})] \leq \mathop{\mathbb{E}}_{\mathbf{x}\sim R|_C}[w^*(\mathbf{x})]. \tag{3}$$

This is a consequence of convexity, as is apparent from the following restatement of Equation (2), which says that for sandwiching, we want the weights w to have the right correlation with the true weights w^* under $P|_C$.

Lemma II.1. *Equation* (2) *is equivalent to*

$$\left(\mathop{\mathbb{E}}_{\mathbf{x}\sim P|_C}[w^*(\mathbf{x})]\right)^2 \leq \mathop{\mathbb{E}}_{\mathbf{x}\sim P|_C}[w(\mathbf{x})w^*(\mathbf{x})] \leq \mathop{\mathbb{E}}_{\mathbf{x}\sim P|_C}\left[w^*(\mathbf{x})^2\right]. \tag{4}$$

Proof. We can rewrite the expectation under $R|_C$ in Equation (2) in terms of expectation under $P|_C$ as follows

$$\mathop{\mathbb{E}}_{\mathbf{x}\sim R|_C}[w(\mathbf{x})] = \frac{\sum_{x\in C} p(x)w(x)w^*(x)}{\sum_{x\in C} p(x)w^*(x)} = \frac{\mathbb{E}_{\mathbf{x}\sim P|_C}[w(\mathbf{x})w^*(\mathbf{x})]}{\mathbb{E}_{\mathbf{x}\sim P|_C}[w^*(\mathbf{x})]} \tag{5}$$

$$\mathop{\mathbb{E}}_{\mathbf{x}\sim R|_C}[w^*(\mathbf{x})] = \frac{\sum_{x\in C} p(x)w^*(x)^2}{\sum_{x\in C} p(x)w^*(x)} = \frac{\mathbb{E}_{\mathbf{x}\sim P|_C}[w^*(\mathbf{x})^2]}{\mathbb{E}_{\mathbf{x}\sim P|_C}[w^*(\mathbf{x})]}. \tag{6}$$

Plugging these into Equation (2), we can rewrite those inequalities as

$$\mathop{\mathbb{E}}_{\mathbf{x}\sim P|_C}[w^*(\mathbf{x})] \leq \frac{\mathbb{E}_{\mathbf{x}\sim P|_C}[w(\mathbf{x})w^*(\mathbf{x})]}{\mathbb{E}_{\mathbf{x}\sim P|_C}[w^*(\mathbf{x})]} \leq \frac{\mathbb{E}_{\mathbf{x}\sim P|_C}[w^*(\mathbf{x})^2]}{\mathbb{E}_{\mathbf{x}\sim P|_C}[w^*(\mathbf{x})]}$$

which is equivalent to Equation (4). □

III. Multi-calibrated partitions and sandwiching bounds

A collection of disjoint subsets $\mathcal{S} = \{S_i\}_{i=1}^m$ such that $\cup_i S_i = \mathcal{X}$ is called a partition of $\mathcal{X}$ of size m. For each $x \in \mathcal{X}$, there exists a unique $S \in \mathcal{S}$ containing it. The family of distributions Q we consider are obtained by fixing a partition $\mathcal{S}$ of $\mathcal{X}$ and then reweighing each $S \in \mathcal{S}$ so that its weight matches R. Within S, we retain the marginal distribution $P|_S$. In this section, we assume that P and R are both supported on the entire domain $\mathcal{X}$, so that the distributions $P|_S$ and $R|_S$ are well-defined for any non-empty subset S. We will relax this requirement in Section III-A

Definition III.1. *Given a prior distribution P and a target distribution R supported on $\mathcal{X}$, and a partition $\mathcal{S}$ of $\mathcal{X}$, the $(P, R, \mathcal{S})$-reweighted distribution Q over $\mathcal{X}$ is given by*

$$Q(x) = R(S)P(x|S) \text{ for } S \in \mathcal{S} \text{ s.t. } x \in S. \tag{7}$$

Equivalently, $Q = w \cdot P$ where $w(x) = R(S)/P(S)$ for all $x \in S \in \mathcal{S}$.

The equivalence follows by observing that $R(S)P(x|S) = R(S)P(x)/P(S) = w(x)P(x)$.

Intuitively, the goal of multi-calibration is to find a partition $\mathcal{S}$, hopefully of small size, whose reweighting will be sufficient to get accuracy for a large family of tests $\mathcal{C}$. We formalize this below.

Definition III.2. *(α-approximate multi-calibration) Let $\alpha > 0$. A partition $\mathcal{S}$ of $\mathcal{X}$ is α-multi-calibrated for $(P, R, \mathcal{C})$ if for every $C \in \mathcal{C}$ and $S \in \mathcal{S}$, the $(P, R, \mathcal{S})$-reweighted distribution Q satisfies*

$$\Big|Q(C \cap S) - R(C \cap S)\Big| \leq \alpha R(S). \tag{8}$$

The multi-calibration condition has the following equivalent formulations (proved in Appendix A):

Lemma III.3. *Equation* (8) *is equivalent to either of the following equations*

$$\Big|P(C|S) - R(C|S)\Big| \leq \alpha, \tag{9}$$

$$\left|w(S) - \mathop{\mathbb{E}}_{\mathbf{x}\sim P|_{C\cap S}}[w^*(x)]\right| \leq \frac{\alpha R(S)}{P(C \cap S)}. \tag{10}$$

Equation (9) reveals that the definition of multi-calibration is symmetric in P and R. Equation (10) connects our definition to multi-calibration in the supervised setting from [1]. To see this, consider the importance weights w^* of R as the ground truth, and w as our prediction which is fixed for each $S \in \mathcal{S}$. For each S and $C \in \mathcal{C}$, we compare our prediction to the conditional expectation of the ground truth, analogous to the classical notion of calibration for a predictor [31].

a) Multi-calibration implies sandwiching bounds.: For the weight function w, and $k \geq 1$ define

$$\|w\|_k = \left(\mathop{\mathbb{E}}_{\mathbf{x}\sim P}[w(\mathbf{x})^k]\right)^{1/k} = \left(\sum_{S\in\mathcal{S}} \frac{R(S)^k}{P(S)^{k-1}}\right)^{1/k}.$$

Observe $\|w\|_1 = 1$ and $\|w\|_k$ increases with k. The following result is proved in Appendix A:

Theorem III.4. *If the partition $\mathcal{S}$ is α-approximately multi-calibrated for $(P, R, \mathcal{C})$ and $w : \mathcal{X} \to \mathbb{R}$ is the corresponding importance weight function, then*

$$\mathop{\mathbb{E}}_{\mathbf{x}\sim P|_C}[w^*(\mathbf{x})] - 2\alpha\frac{\|w\|_2^2}{R(C)} \leq \mathop{\mathbb{E}}_{\mathbf{x}\sim R|_c}[w(\mathbf{x})] \leq \mathop{\mathbb{E}}_{\mathbf{x}\sim R|_C}[w^*(\mathbf{x})] + 3\alpha\frac{\|w\|_2^2}{R(C)} \tag{11}$$

Comparing Theorem III.4 to the strict sandwiching requirement stated in Equation (2), we see that the theorem allows for a slack of roughly $\alpha \|w\|_2^2 / R(C)$. Some such slack is unavoidable given our model where we only have access to random samples from both P and R. One can view α above as the accuracy parameter which decides the sample complexity. With more samples, α can be made smaller, hence we approach strict sandwiching, which can be thought of as the limit with infinitely many samples. In contrast, in Section IV, we give negative examples showing previous algorithms achieving α-multi-accuracy do not satisfy Sandwiching bounds even for $\alpha = 0$. In particular, no bound analogous to Equation (11) holds for those algorithms, even as the number of samples goes to infinity.

We justify why the terms $R(C)$ and $\|w\|_2$ appear in the slack. With access only to samples we cannot hope to give guarantees for sets C which are very small, for the same reason that guarantees for singleton point sets are not possible. In particular, with only a finite number of samples, we cannot tell whether $R(C)$ is 0, or just very small, similarly with $P(C)$. Hence, we can only expect meaningful guarantees for sets where $R(C)$ is reasonably large, say $1/\alpha$.

Similarly, some bound on the norm of the importance weights seems unavoidable. Imagine that we see a set S in our partition with 0.1-fraction of the samples from R, but no samples from P. It is hard to tell with finitely many samples whether $P(S) = 0$ so the importance weight here tends to infinity or whether $P(S)$ is non-zero but small, say, $1/10\alpha$, so the importance weight is just very large. The simple solution would be to assume a hard bound $\max w^*(x) \leq B$ for some constant B. We only require a bound on the l_2 norm which is weaker, because our weights always satisfy $\|w\|_2^2 \leq \|w^*\|_2^2 \leq \max_x w^*(x)$.

A. A relaxed notion of approximate multi-calibration

We now relax the notion of approximate multi-calibration to handle the case where R and P might not have identical supports. We do this in a *robust* manner, that lets us handle the case where one of P and R assigns non-zero but tiny probability to some set T. For such T, enforcing the closeness of $P(C|T)$ and $R(C|T)$ based on random samples will be very expensive in terms of sample complexity. Such T corresponds to a region where the importance weight $w(T) = R(T)/P(T)$ is either very high or very low. We will relax our defintion to allow regions in the partition with $w(T) \leq \beta$ or $w(T') \geq 1/\beta$ for some parameter β without having to determine exactly what those weights are (which in itself would require many samples). This motivates the following definition.

Definition III.5. *Let $\alpha \geq 0, \beta \geq 0$, let $\mathcal{C} \subseteq 2^{\mathcal{X}}$ be a collection of sets. The partition $\mathcal{S} = \{S_1, \ldots, S_m, T_0, T_1\}$ is (α, β)-multi-calibrated for $\mathcal{C}$ under P, R if*

$$P(T_0) \leq \min(\beta, R(T_0)), \;\; R(T_1) \leq \min(\beta, P(T_1)), \tag{12}$$

$$\forall C \in \mathcal{C}, i \in [m], \;\; \Big|Q(C \cap S_i) - R(C \cap S_i)\Big| \leq \alpha R(S_i). \tag{13}$$

(α, β)-multi-calibration permits two *exceptional* subsets T_0 and T_1 that do not satisfy Equation (8), but these subsets must have small measure under P and R respectively. We will use $\mathcal{T} = \{T_0, T_1\}$ to denote the exceptional sets. The advantage of allowing for $\mathcal{T}$ is that for every $S \in \mathcal{S} \setminus \mathcal{T}$, we *can ensure* that $w(S) = R(S)/P(S)$ lies in the range $[\beta, 1/\beta]$, since sets violating this bound may be absorbed in $\mathcal{T}$. This lets us enforce Equation (13) with sample complexity that depends on $1/\beta$.

When $\beta = 0$ we recover the notion of α-multi-calibration. When $\beta > 0$, we show that the distributions R and P are β-close in statistical distance to distributions R^h and P^h respectively that are indeed α-multicalibrated.

Lemma III.6. *Define the distribution P^h which is identical to P on $\mathcal{S} \setminus \{T_0\}$. Let $P^h(T_0) = P(T_0)$, and $P^h|_{T_0} = R|_{T_0}$. Similarly, define R^h to be identical to R on $\mathcal{S} \setminus \{T_1\}$. Let $R^h(T_1) = R(T_1)$, and $R^h|_{T_1} = P|_{T_1}$. If the partition $\mathcal{S}$ is (α, β)-multi-calibrated for $(P, R, \mathcal{C})$, then $\mathrm{d_{TV}}(P^h, P) \leq \beta$, $\mathrm{d_{TV}}(R^h, R) \leq \beta$, and the partition $\mathcal{S}$ is α-multi-calibrated for $(P^h, R^h, \mathcal{C})$.*

We show a sandwiching bound for (α, β)-approximate multi-calibration. The error terms depend on β and $\|w\|_4^2$ in comparison to Theorem III.4. The proof appears in Appendix A.

Theorem III.7. *Assume the partition $\mathcal{S}$ is (α, β)-approximately multi-calibrated for $(P, R, \mathcal{C})$ and $w : \mathcal{X} \to \mathbb{R}$ is the corresponding importance weight function. Let*

$$\ell(\alpha, \beta, w) = \alpha \|w\|_2^2 + \sqrt{\beta} \|w\|_4^2 . \tag{14}$$

Then for every $C \in \mathcal{C}$,

$$\mathop{\mathbb{E}}_{\mathbf{x} \sim P|_C}[w^*(\mathbf{x})] - \frac{2\ell(\alpha, \beta, w)}{R(C)} - \frac{2(\alpha + 2\beta)}{P(C)} \leq \mathop{\mathbb{E}}_{\mathbf{x} \sim R|_c}[w(\mathbf{x})] \leq \mathop{\mathbb{E}}_{\mathbf{x} \sim R|_C}[w^*(\mathbf{x})] + \frac{3\ell(\alpha, \beta, w)}{R(C)}. \tag{15}$$

IV. Multi-accuracy alone does not guarantee sandwiching

We consider two well-studied methods for density ratio estimation; log-linear KLIEP [27, 3] and MaxEnt [28, 29, 32, 30], which are essentially duals of each other. Our motivation for considering these algorithms is two-fold. Firstly, they can be interpreted as giving a set-wise guarantee we call multi-accuracy, in analogy with the corresponding notion in supervised learning [1, 33], which is weaker than approximate multi-calibration. Secondly, these algorithms are known to out-perform other density-ratio estimation algorithms in the non-realizable setting [34]. To state these algorithms, we define two families of distributions.

Definition IV.1. *For $\alpha \in [0, 1]$, the distribution Q is α-multi-accurate in expectation (α-multiAE) for $(R, \mathcal{C})$ if for every $C \in \mathcal{C}$, it holds that $|Q(C) - R(C)| \leq \alpha$. Define $K^\alpha = K^\alpha(R, \mathcal{C})$ to be the set of all α-multi-accurate distributions for $(R, \mathcal{C})$.*

α-multi-calibration for the partition $\mathcal{S}$ implies multi-accuracy in expectation for the re-weighted distribution Q; this follows by summing Equation (8) over all $S \in \mathcal{S}$. For every $\mathcal{C}$ and $\alpha \geq 0$, the set K^α is a convex set, since it is given by linear constraints, and it is non-empty since $R \in K^\alpha$. Our second family of distributions are Gibbs distributions.

Definition IV.2. *A Gibbs distribution is a distribution of the form*

$$Q(x) = P(x) \exp\left(\sum_{c \in \mathcal{C}} \lambda_c c(x) - \lambda_0\right). \tag{16}$$

Let $\mathcal{G} = \mathcal{G}(P, \mathcal{C})$ denote the set of all Gibbs distributions.

Note that the free parameters are $\lambda_{\mathcal{C}} = \{\lambda_c\}_{c \in \mathcal{C}}$, from these, we set the normalization constant λ_0 to ensure that $\mathbb{E}_P[w(\mathbf{x})] = 1$, so that Q is a distribution. We now describe log-linear KLIEP and MaxEnt, both of which find a multi-accurate Gibbs distribution.

1) **Log-linear KLIEP** [27, 14, 3] : Find the Gibbs distribution $Q \in \mathcal{G}$ that minimizes $D(R\|Q)$. The goal in [27] is to find a good density-ratio estimate. Essentially this algorithm is proposed in the work of [14] for estimating KL divergence.
2) **MaxEnt** [28, 29, 32, 35, 30] : Find the distribution $Q^\alpha \in K^\alpha$ that minimizes $D(Q\|P)$.

We have not found the equivalence of these algorithms noted explicitly in the literature, but it follows from known results on convex duality [29, 32, 30].

Lemma IV.3. *[30, Theorem 2] For $Q \in \mathcal{G}(P, \mathcal{C})$ as in Equation* (63)*, let $\ell_1(Q) = \sum_{c \in \mathcal{C}} |\lambda_c|$. $Q^\alpha \in K^\alpha \cap \mathcal{G}$ is the optimal solution to each of the following programs:*

$$\min_{Q \in K^\alpha} D(Q\|P), \tag{17}$$

$$\min_{Q \in \mathcal{G}} D(R\|Q) + \alpha \ell_1(Q). \tag{18}$$

The first program is the one solved by MaxEnt. The second is an ℓ_1-regularized version of the program considered by KLIEP. We derive the exact KLIEP program by setting $\alpha = 0$, in the MaxEnt literature this was analyzed by [29]. Our main result in this section is that these algorithms do not guarantee sandwiching bounds. The proof is in Appendix D.

Theorem IV.4. *Fix any constant $B > 1$, and $\alpha \geq 0$. There exist distributions P, R on $\{0, 1\}^n$, a collections of sets $\mathcal{C}$ and $C \in \mathcal{C}$ such that if $Q^\alpha = w \cdot P$ is the solution to Program* (64) *then*

$$\mathop{\mathbb{E}}_{\mathbf{x} \sim P|_C}[w^*(\mathbf{x})] > B \mathop{\mathbb{E}}_{\mathbf{x} \sim R|_C}[w(\mathbf{x})].$$

There exist distributions P_1, R_1 on $\{0, 1\}^n$, a collections of sets $\mathcal{C}_1$ and $C_1 \in \mathcal{C}_1$ such that if $Q_1^\alpha = w_1 \cdot P_1$ is the solution to Program (64)*, then writing $R_1 = w_1^* \cdot P_1$,*

$$\mathop{\mathbb{E}}_{\mathbf{x} \sim R_1|_C}[w_1(\mathbf{x})] > B \mathop{\mathbb{E}}_{\mathbf{x} \sim R_1|_C}[w_1^*(\mathbf{x})].$$

Let us give some intuition for why multi-accuracy by itself cannot guarantee sandwiching whereas multi-calibration does. We remind the reader of Lemma II.1, which rephrases the sandwiching conditions as saying that the correlation between $w(x)$ and $w^*(x)$ under $P|_C$ behaves as expected. Multi-accuracy can be rephrased as saying that

$$\mathop{\mathbb{E}}_{P|_C}[w(x)] = \frac{Q(C)}{P(C)} \approx \frac{R(C)}{P(C)} = \mathop{\mathbb{E}}_{P|_C}[w^*(x)].$$

Thus while it guarantees that $w(x)$ and $w^*(x)$ have similar expectations under $P|_C$, it does not give guarantees on their correlation. In contrast, the calibration property of multi-calibration (captured by Equation (10)) guarantees that conditioned on each value $w(x) = w(S)$, the expected value of $w^*(x)$ under $P|_C$ is close to $w(S)$. This ensures that $w(x)$ and $w^*(x)$ are better correlated.

V. Algorithm for approximate multi-calibration

In this section, we give an efficient algorithm that computes a multicalibrated partition, given access to a weak agnostic learner for $\mathcal{C}$. The algorithm is inspired by Boosting via Branching Programs from [26]. We first define weak agnostic learning [36, 37]. Given a collection of sets $\mathcal{C} \subseteq 2^{\mathcal{X}}$, we associate $C \in \mathcal{C}$ with its indicator function $c : \mathcal{X} \to \{0,1\}$. With this view, we can also regard the set $\mathcal{C}$ as a hypothesis class with binary-valued functions $c : \mathcal{X} \to \{0,1\}$.

Definition V.1. *A (α, α', L)-weak agnostic learning algorithm for a class $\mathcal{C}$ is given L samples from a distribution $\mathcal{D} = (\mathbf{x}, \mathbf{y})$ where $\mathbf{x} \in \mathcal{X}$ and $\mathbf{y} \in \{0,1\}$. If there exists a set $C \in \mathcal{C}$ with corresponding indicator function $c : \mathcal{X} \to \{0,1\}$ such that $\Pr_{\mathcal{D}}[c(\mathbf{x}) = \mathbf{y}] \geq (1+\alpha)/2$, then the learner will return an indicator function $c' : \mathcal{X} \to \{0,1\}$ corresponding to some $C' \in \mathcal{C}$ such that $\Pr_{\mathcal{D}}[c'(\mathbf{x}) = \mathbf{y}] \geq (1+\alpha')/2$ for some $0 < \alpha' \leq \alpha$.*

We allow $\alpha - \alpha'$ to depend on L, typically it decreases with L. In the above definition, for simplicity we have defined the learner to be a proper learner (it returns a hypothesis within $\mathcal{C}$), and do not allow for probability of error. Given two distributions P and R, a weak learner for $\mathcal{C}$ can be used to find $C \in \mathcal{C}$ such that $|R(C) - P(C)|$ is large, a view that we will use hereafter.

Informally, the weak agnostic learning assumption says that if there is a hypothesis in $\mathcal{C}$ that labels the data reasonably well (say with 0-1 loss of 0.7), then we can efficiently find one that has a non-trivial advantage over random guessing (say with 0-1 loss of 0.51). Weak agnostic learning was introduced in [36]. It captures a common modeling assumption in practice, and is a well-studied notion in the computational learning literature [37, 38, 39]. The assumption has also been used in previous works in the multi-calibration literature [1, 40].

Given sample access to distributions P, R, we will construct a multicalibrated partition by starting from the trivial partition and iteratively modifying it till we achieve multi-calibration. We use $(\mathcal{S}^t, \mathcal{T}_0, \mathcal{T}_1)$ to denote the t^{th} partition, and Q^t to denote the corresponding reweighted distribution. The partition consists of three groups of sets:

- **Large weights:** $\mathcal{T}_0$ consisting of sets T such that $R(T)/P(T) \geq 2/\beta$.
- **Small weights:** $\mathcal{T}_1$ consisting of sets T such that $R(T)/P(T) \leq \beta/2$.
- **Medium weights:** $\mathcal{S}^t$ will consists of sets S such that $R(S)/P(S) \in [\beta/2, 2/\beta]$.

The collections $\mathcal{T}^0, \mathcal{T}^1$ start empty and grow monotonically. Once set T is added to either, that set is not modified. All sets in $\mathcal{T}_0$ will eventually be merged into a single set T_0 such that $P(T_0) \leq \beta$, while the sets in $\mathcal{T}_1$ will be merged into T_1. Doing the merging at the end simplifies the analysis, but it is fine to think of each as a single state that keeps growing. Our algorithm will mostly focus on the medium sets in $\mathcal{S}^t$, although occasionally sets will be added to $\mathcal{T}_0$ or $\mathcal{T}_1$, hence we use the superscript t to account for how it changes over iterations. The algorithm combines two operations.

- Split: This operation takes $S \in \mathcal{S}_t$ where $R(S), P(S)$ are sufficiently large, and $C \in \mathcal{C}$ such that $|P(C|S) - R(C|S)| > \alpha'$, and split S into two states, $C \cap S$ and $\bar{C} \cap S$. We find the pair S, C by running the weak agnostic learner to distinguish the distributions $P|_S$ and $R|_S$. The new sets are classified as small, medium or large.
- Merge: This operation is applied to $\mathcal{S}^t$ when the number states in it goes beyond a certain bound. It merges those states in $\mathcal{S}^t$ with similar importance weights into a single state, and halves the number of states.

Algorithm 1 $\mathrm{Split}(S, C)$

```
Input:
  • $S \in \mathcal{S}_t$ s.t. $R(S) \geq \beta/4m$, $P(S) \geq \beta/4m$.
  • $C \in \mathcal{C}$ s.t. $|R(C|S) - P(C|S)| > \alpha'$.
Replace $S$ with the two states $S_0 = S \cap C$ and $S_1 = S \cap \bar{C}$.
```

We will analyze the Split operation using $D(Q_t \| P)$ as the potential function.

Lemma V.2. *We have $D(Q_{t+1} \| P) - D(Q_t \| P) \geq 4R(S)\alpha'^2$.*

Proof. Since Q_{t+1} differs from Q_t by splitting S into $S \cap C$ and $S \cap \bar{C}$, we can write the LHS as

$$\begin{aligned}
&R(S \cap C) \log \left(\frac{R(S \cap C)}{P(S \cap C)} \right) + R(S \cap \bar{C}) \log \left(\frac{R(S \cap \bar{C})}{P(S \cap \bar{C})} \right) - R(S) \log \left(\frac{R(S)}{P(S)} \right) \\
&= R(S \cap C) \log \left(\frac{R(S \cap C)P(S)}{R(S)P(S \cap C)} \right) + R(S \cap \bar{C}) \log \left(\frac{R(S \cap \bar{C})P(S)}{R(S)P(S \cap \bar{C})} \right) \\
&= R(S) \left(R(C|S) \log \left(\frac{R(C|S)}{P(C|S)} \right) + R(\bar{C}|S) \log \left(\frac{R(\bar{C}|S)}{P(\bar{C}|S)} \right) \right). \qquad (19)
\end{aligned}$$

The expression in braces is the KL divergence between two Bernoulli random variables that are 1 with probability $R(C|S)$ and $P(C|S)$ respectively. Hence applying Pinsker's inequality [41] gives the desired bound:

$$R(C|S) \log \left(\frac{R(C|S)}{P(C|S)} \right) + R(\bar{C}|S) \log \left(\frac{R(\bar{C}|S)}{P(\bar{C}|S)} \right) \geq \left| R(C|S) - P(C|S) \right|^2 \geq 4\alpha'^2. \qquad (20)$$

□

Algorithm 2 Merge(δ)

```
Input: parameter $\delta$.
  1) Let $m = \lceil \frac{1}{\delta} \log \left( \frac{4}{\beta^2} \right) \rceil$.
  2) For each $i \in \{1, \dots, m\}$:
         Form a new state $S_i$ by merging all states $S' \in \mathcal{S}_j$ such that
             $\frac{R(S')}{P(S')} \in \left( \frac{e^{(i-1)\delta}\beta}{2}, \frac{e^{i\delta}\beta}{2} \right]$.
  3) Let $\mathcal{S}^{t+1} = \{S_i\}_{i=1}^{m}$, discarding any empty states.
```

Unlike Split, Merge can reduce the KL divergence, but we can bound the loss.

Lemma V.3. *We have* $D(Q_t \| P) - D(Q_{t+1} \| P) \leq \delta$.

Proof. Let $S'_1, \dots, S'_\ell \in \mathcal{S}_t$ denote the states that are merged to form $S_i \in \mathcal{S}_{t+1}$. For each $k \in [\ell]$,

$$\frac{R(S'_k)/P(S'_k)}{R(S_i)/P(S_i)} \leq e^{\delta}.$$

We use this to bound the decrease in potential from S_i as

$$\begin{aligned}
\sum_{k=1}^{\ell} R(S'_k) \log \left(\frac{R(S'_k)}{P(S'_k)} \right) - R(S_i) \log \left(\frac{R(S_i)}{P(S_i)} \right) &= \sum_{k=1}^{\ell} R(S'_k) \left(\log \left(\frac{R(S'_k)}{P(S'_k)} \right) - \log \left(\frac{R(S_i)}{P(S_i)} \right) \right) \\
&= \sum_{k=1}^{\ell} R(S'_k) \log \left(\frac{R(S'_k)/P(S'_k)}{R(S_i)/P(S_i)} \right) \leq \sum_{k=1}^{\ell} R(S'_k)\delta = R(S_i)\delta.
\end{aligned}$$

The claim follows by summing over all $S_i \in \mathcal{S}_{t+1}$. □

We use these in Algorithm 3 which computes a multi-calibrated partition.

To sketch the overall analysis briefly, we have shown that $D(Q_t \| P)$ increases during a Split, and decreases during a Merge. But there are m Split operations between any two Merge operations, so overall $D(Q_t \| P)$ increases, but can never exceed $D(R \| P)$. Details appear in appendix A.

Theorem V.4. *Given an* (α, α', L) *weak agnostic learning algorithm for* $\mathcal{C}$*, Algorithm 3 returns an* (α, β)*-approximately multi-calibrated partition of size* $m = O(\log(1/\beta)/(\alpha'^2\beta))$ *that can be represented by a* $\mathcal{C}$*-branching program where each node is labelled by* $c \in \mathcal{C}$*. The algorithm performs* $T = \widetilde{O}(D(R \| P)/(\beta^2\alpha'^4)$ *Split and Merge operations. It makes* $O(T)$ *calls to the weak agnostic learner, where each call requires* $\widetilde{O}(L/(\beta^2\alpha'^2))$ *samples from each of* R *and* P.

VI. Other Related Work

In Section VIII, we discussed some of the diverse applications of importance weights or density-ratio estimation which span many communities. We refer the reader to the book [42] for a more comprehensive overview of the related work, especially in the context of the machine learning literature. In addition to density ratio estimation, relevant results appear in the literature under the subject of divergence estimation [14, 15, 17, 18], learning max-Entropy distributions [28, 30] and domain adaptation

```
Algorithm 3 Multi-Calibrate($P, R, \mathcal{C}, \alpha, \beta$)
Inputs: $\alpha, \beta > 0$, distributions $P, R$, class $\mathcal{C}$ that is $(\alpha, \alpha', L)$-weakly agnostically learnable.
Output: A partition that is $(\alpha, \beta)$-multicalibrated for $\mathcal{C}$ under $P, R$.

Let $\mathcal{S}^1 = \{\mathcal{X}\}, \mathcal{T}_0^1 = \mathcal{T}_1^1 = \{\}$. Let $\delta = \beta\alpha'^2/2$, and $m = \lceil \frac{1}{\delta} \log\left(\frac{4}{\beta^2}\right) \rceil$.
For $t \geq 1$
  1) If $|\mathcal{S}_t| \geq 2m$, then run $\mathrm{Merge}(\delta)$.
  2) If the weak agnostic leaner finds $S \in \mathcal{S}^t$, $C \in \mathcal{C}$ such that
         $R(S) \geq \beta/4m, P(S) \geq \beta/4m, \left|R(C|S) - P(C|S)\right| \geq \alpha'.$
     2.1. Run $\mathrm{Split}(S, C)$ and obtain $S_0, S_1$
     2.2. If $P(S_0) < \beta/4m$ and $P(S_0) < R(S_0)$, place $S_0$ in $\mathcal{T}_0$. Else, if $R(S_0) < \beta/4m$ place $S_0$ in $\mathcal{T}_1$.
     2.3. Repeat previous step for $S_1$
     2.4. Repeat the loop
     If the weak learner fails, exit the loop.
Post-Processing:
  1) Move all $S \in \mathcal{S}^t$ such that $P(S) < \beta/4m$ and $P(S) \leq R(S)$ from $\mathcal{S}^t$ to $T_0$. Move all remaining $S \in \mathcal{S}^t$ such that $R(S) < \beta/4m$ from $\mathcal{S}^t$ to $T_1$.
  2) Merge all $T \in \mathcal{T}_0$ into a single state $T_0$. Merge all $T \in \mathcal{T}_1$ into a single state $T_1$.
  3) Return the partition $\mathcal{S} = \mathcal{S}^t \cup \{T_0\} \cup \{T_1\}$.
```

[12, 13, 43]. Kernel based approaches for estimating importance weights have also been proposed, starting with Kernel Mean Matching (KMM) introduced in [44]. If the underlying kernel is universal, then under the limit of infinite data KMM provably recovers the true importance weights [45, 44]. In the limit of finite data however, kernel based approaches can be interpreted as generalizations of moment-matching methods such as [46] which seek to match some moments of the data [24]. In the context of our work, such based approaches guarantee multi-accuracy in expectation with respect to the moments or the feature space more generally.

Calibration has been well-studied in the statistics literature, in the context of forecasting [31]. It was introduced in the algorithmic fairness literature by [47]. The notion of multi-calibration as a multi-group fairness notion in supervised learning was introduced in [1] (see also [48]) and has subsequently generated significant interest [33, 40, 49, 50]. But all previous work to our knowledge has been in the supervised setting. Our work appears to be the first to introduce notions of group-fairness in unsupervised learning. Our algorithm for computing a multi-calibrated partition is inspired by the Boosting by Branching programs work of [26], which in turn builds on [51]. The work of [37] showed that the Mansour-McAllester algorithm can be viewed as an agnostic boosting algorithm, improving on the results of [36] who introduced the notion of agnostic boosting.

VII. Conclusion

In this work, we have put forth a theoretical framework for reasoning about the accuracy of importance weights for sub-populations of the dataset. We presented an algorithm that provably gives much stronger guarantees in this regard than previously known algorithms in the literature. The next step is to implement this algorithm and test its guarantees for real-world datasets. The implications of these stronger guarantees for the numerous applications of importance weights should also be explored: among these we consider divergence estimation and anomaly detection to be particularly promising.

VIII. Introduction

The problem of comparing and contrasting distributions is central to machine learning. As an illustrative example, consider a medical researcher who has patient data from a prior outbreak (P) of a disease and a more recent one (R). The data contains medical and demographic information as features. The researcher wishes to identify patterns of shifts between the epidemiological behavior of the two outbreaks. The researcher models the two outbreaks as distributions P and R on a domain $\mathcal{X}$ of features and then measures the divergence between them. There are several divergences studied in the literature, of which the Kullback-Leibler (KL) divergence is arguably the most important [41]. It is defined as

$$D\left(R \| P\right) = \mathop{\mathbb{E}}_{\mathbf{x} \sim R}\left[\log\left(\frac{R(\mathbf{x})}{P(\mathbf{x})}\right)\right]. \tag{21}$$

Say the researcher develops some model which finds that the divergence $D\left(R \| P\right)$ is large, and concludes that the two outbreaks differ significantly. Going further, the researcher would want to determine how various *sub-groups* of the population contribute to the divergence. Concretely, let C be some sub-population of interest, say, people of a certain age bracket. Let

$R|_C$ (respectively $P|_C$) be the distribution conditioned on $x \in C$. Can the researcher's model be trusted to give insight into the *conditional* divergence between $R|_C$ and $P|_C$? Or might the model be misleading by significantly over/under estimating $D\left(R|_C \| P|_C\right)$?

This is important not just for the utility of the researcher, it is also motivated by the desire for the set C to be treated fairly by the model. Realistically, there could be many (possible even infinite) sub-groups of interest, and they might overlap. We model them as coming from a family $\mathcal{C}$ of subsets of the domain. The fundamental question that we ask is, given $\mathcal{C}$ and two distributions R and P, can we learn a single global model which reliably estimates the divergence for sets in $\mathcal{C}$ simultaneously? This question is an instance of a larger challenge that machine learning research is currently grappling with: when can we trust models which are trained to minimize some objective over a large population to be accurate for certain sub-groups of the original population? This is particularly challenging when the collection of sub-groups we care about is large and possibly overlapping.

As first step, we need to define our desiderata for reliable estimates for subgroups. Any rigorous definition that does not place restrictions on R and P immediately runs into information-theoretic barriers [22, 23]. The following proposition shows that distinguishing whether the KL divergence between two distributions P and R is 0 or maximally large requires a sample complexity polynomial in the size of the domain, which could be exponential in the dimensionality of the data.

Proposition VIII.1. *Given any $t > 0$ and a domain $\mathcal{X}$ of size $|\mathcal{X}|$, consider two distributions P, R over $\mathcal{X}$ such that $R(x)/P(x) \leq t$ (so that $D\left(R\|P\right) \leq \log(t)$). Any algorithm that can distinguish whether $D\left(R\|P\right) = 0$ or $D\left(R\|P\right) \geq \log(t)/10$ with success probability at least $2/3$ from i.i.d. samples from P and R, requires at least $\Omega(\sqrt{|\mathcal{X}|})$ samples.*

The main conceptual contribution of this work is to propose *multi-group attribution* as a meaningful notion of reliable estimation for sub-groups that evades this barrier and is efficiently achievable.

A. Multi-group attribution

To discuss in more detail the barrier presented by Lemma VIII.1 and how we circumvent it, we need some notation. Given sample access to two distributions R and P over a domain $\mathcal{X}$, our goal is to estimate $D\left(R\|P\right)$. Let $\mathcal{C} = \{C : \mathcal{X} \to \{0,1\}\}$ denotes a family of subsets that contains the sub-groups we are interested in.[2] We want to be able to meaningfully attribute portions of our estimate of KL divergence to various subsets $C \in \mathcal{C}$. Eq. 21 shows that measuring the divergence does not require models for both P and R, rather it suffices to have a model w for the ratio $w^*(x) = R(x)/P(x)$, as the divergence is just the expectation of $\log w^*$ under R. The function w is referred to as the importance weights of our model or the density ratio [3]. Indeed, while in general w^* cannot be learned exactly, there are numerous algorithms in the literature that learn models w for w^* satsfying certain desiderata [27, 15, 30]. In this paper we focus on procedures which use importance weights to estimate the KL divergence.

Given $w : \mathcal{X} \to \mathbb{R}^+$ where $\mathbb{E}_P[w(x)] = 1$, let Q be the distribution defined by $Q(x) = w(x)P(x)$, we denote this by $Q = w \cdot P$. For every $C \subset \mathcal{X}$, let $Q(C) = \Pr_Q[\mathbf{x} \in C]$. When $Q(C) > 0$, let $Q|_C = Q(x)/Q(C)$ for $x \in C$ denote the conditional distribution of Q over C. We let $w^*(x) = R(x)/P(x)$ denote the ground truth importance weights. Denote the KL divergence between Bernoulli variables with parameters p and q by

$$d(p,q) = p \log\left(\frac{p}{q}\right) + (1-p)\log\left(\frac{1-p}{1-q}\right). \tag{22}$$

a) Defining attribution.: To provide guarantees for sub-groups of the population we ask: how much of the total KL divergence should one *attribute* to a sub-group $C \in \mathcal{C}$? Our starting point is the following decomposition of the divergence, which follows from the chain rule [41]:

Lemma VIII.2. *Let $C \subseteq \mathcal{X}$ and let $\bar{C} = \mathcal{X} \setminus C$. Then*

$$\begin{aligned} D\left(R\|P\right) = d(R(C), P(C)) + R(C)D\left(R|_C\|P|_C\right) + \\ R(\bar{C})D\left(R|_{\bar{C}}\|P|_{\bar{C}}\right). \end{aligned}$$

1) The *marginal term* $d(R(C), P(C))$ accounts for differences in the measure of C under R and P.
2) We call $R(C)D\left(R|_C\|P|_C\right)$ the *conditional contribution from* C as it is attributable to differences between R and P conditioned on C. Similarly $R(\bar{C})D\left(R|_{\bar{C}}\|P|_{\bar{C}}\right)$ is the *conditional contribution from* C.

Clearly the first two terms in this decomposition are attributable to C. This motivates our definition of *ideal* attribution, which lets us estimate these terms.

Definition VIII.3. *The distribution Q satisfies ideal multi-group attribution for $(P, R, \mathcal{C})$ if $\forall C \in \mathcal{C}$,*

$$Q(C) = R(C), \tag{23}$$

$$D\left(Q|_C\|P|_C\right) = D\left(R|_C\|P|_C\right). \tag{24}$$

[2] Assume for simplicity that $\mathcal{C}$ is closed under complement.

The first condition is known as multiaccuracy [**?**], and there are many known algorithms that achieve it. It suffices to correctly estimate the marginal term. Having Equation (24) as well would let us estimate the conditional contribution from C. Unfortunately, as discussed earlier, finding Q satisfying this equation (or even a reasonable approximation) is known to be information-theoretically prohibitive (Proposition VIII.1). Instead we propose the following relaxation.

Definition VIII.4. *The distribution Q satisfies (exact) multi-group attribution for $(P, R, \mathcal{C})$ if $\forall C \in \mathcal{C}$,*

$$Q(C) = R(C), \tag{25}$$

$$D\left(R|_C \| Q|_C\right) + D\left(Q|_C \| P|_C\right) = D\left(R|_C \| P|_C\right). \tag{26}$$

We call the second condition the Pythagorean property, in keeping with the literature [41]. Comparing it to (24), there is an additional $D\left(R|_C \| Q|_C\right)$ term on the left. The Pythagorean property relaxes the condition that $R|_C$ and $Q|_C$ must be close (which is impossible to achieve efficiently), to asking that $Q|_C$ lies *in between* $R|_C$ and $P|_C$. Why is this relaxation meaningful for attribution?

b) Soundness and Improvement.: The Pythagorean property implies the following inequalities for the model Q that we will refer to as *soundness* and *improvement* respectively:

$$D\left(R|_C \| P|_C\right) \geq D\left(Q|_C \| P|_C\right), \tag{27}$$

$$D\left(R|_C \| P|_C\right) \geq D\left(R|_C \| Q|_C\right). \tag{28}$$

Let us explain why these are desirable conditions. Soundness and multiaccuracy together imply that

$$R(C) D\left(R|_C \| P|_C\right) \geq Q(C) D\left(Q|_C \| P|_C\right)$$

hence our model Q is conservative in estimating the conditional contribution of C. This endows the model Q with the following *soundness* guarantee: if it attributes large divergence to any set $C \in \mathcal{C}$, the ground truth conditional contribution of C is only larger.

To see why improvement is desirable for attribution, assume it is violated for some $C \in \mathcal{C}$, where $D\left(R|_C \| P|_C\right) < D\left(R|_C \| Q|_C\right)$. Intuitively, Q is meant to be a reweighting of P which is closer to R. But conditioned on C, it is farther from R than P was. Given this, it is unclear that the divergence it attributes to C is meaningful. Improvement guarantees that Q *simultaneously improves* on P as a model for R conditioned on every $C \in \mathcal{C}$; which is desirable both for accuracy and fairness.

c) Another view of the Pythagorean property.: We show how the Pythagorean property arises from requiring that two natural estimators for the conditional KL divergence be equal. Say we have a model $Q = w \cdot P$ for R. We wish to use it to estimate $D\left(R|_C \| P|_C\right)$ for some $C \in \mathcal{C}$. One natural estimator uses $Q|_C$ in place of $R|_C$:

$$\begin{aligned} D\left(Q|_C \| P|_C\right) &= \mathop{\mathbb{E}}_{Q|_C}\left[\log\left(\frac{Q(x)P(C)}{Q(C)P(x)}\right)\right] \\ &= \mathop{\mathbb{E}}_{Q|_C}[\log(w(x)] + \log\left(\frac{P(C)}{Q(C)}\right). \end{aligned} \tag{29}$$

This estimator is always positive by the positivity of KL divergence, but it may violate soundness (27).

A second estimator is obtained by using $w(x)$ (suitably renormalized) in place of the true weights $w^*(x)$ and appears in [3, 27, 15]:

$$\mathop{\mathbb{E}}_{R|_C}\left[\log\left(\frac{w(x)P(C)}{Q(C)}\right)\right] = \mathop{\mathbb{E}}_{R|_C}[\log(w(x))] + \log\left(\frac{P(C)}{Q(C)}\right). \tag{30}$$

The only difference from the previous estimator is that we compute the expectation over $R|_C$. But this estimator has rather different guarantees: we can rewrite it as

$$\begin{aligned} \mathop{\mathbb{E}}_{R|_C}\left[\log\left(\frac{w(x)P(C)}{Q(C)}\right)\right] &= \mathop{\mathbb{E}}_{R|_C}\left[\log\left(\frac{Q|_C(x)}{P|_C(x)}\right)\right] \\ &= \mathop{\mathbb{E}}_{R|_C}\left[\log\left(\frac{R|_C(x)}{P|_C(x)}\right) - \log\left(\frac{R|_C(x)}{Q|_C(x)}\right)\right] \\ &= D\left(R|_C \| P|_C\right) - D\left(R|_C \| Q|_C\right). \end{aligned} \tag{31}$$

Since $D\left(R|_C \| Q|_C\right) \geq 0$, this estimator always guarantees soundness. However, unlike the previous estimator, this estimator might be negative. Improvement (28) captures the condition that it is non-negative. If the two estimators are equal, we get an estimate which is non-negative and a lower bound, and where the model Q satisfies both soundness and improvement. This equivalence is characterized by the Pythagorean property.

Lemma VIII.5. *The following are equivalent:*

1) *The two estimators are equal.*
2) *The Pythagorean property Equation* (26) *holds conditioned on* C.
3) $\mathbb{E}_{R|_C}[\log(w(x))] = \mathbb{E}_{Q|_C}[\log(w(x))]$.

Proof of Lemma VIII.5. The first estimator equals the LHS of (2) by definition (Equation (29)), whereas the second equals the RHS of (2) by Equation (31). The equivalence of (1) and (3) follows by equating the RHS of Equations (29) and (30). □

d) Overview of our results:: We now outline the main contributions of the paper.

1) **Definition of efficient multi-group attribution.** We motivate the problem of multi-group attribution in KL divergence estimation. We present a definition that gives meaningful guarantees but circumvents known information theoretic barriers.
2) **Algorithm for multi-group attribution.** We give an efficient algorithm for multi-group attribution. Our main technical contribution (Section X) is to show that multi-group attribution can be derived from the multi-calibrated partitions introduced by [**?**]. Informally, this is a partition of the domain $\mathcal{X}$ into regions $S_1, \ldots, S_m$ where the condition $R|_{S_i}(C) = P|_{S_i}(C)$ holds for every $i \in [m]$ and $C \in \mathcal{C}$. This means that conditioned on the partition, no $C \in \mathcal{C}$ can distinguish between R and P. Formally, our partitions satisfy a relaxed notion of this condition, which allows them to be computed efficiently. Given such a partition, our main result, Theorem X.1 shows that the distribution $Q = w \cdot P$ where $w(S_i) = R(S_i)/P(S_i)$ for all $x \in S_i$ satisfies multi-group attribution for $\mathcal{C}$.
3) **Impossibility result for prior algorithms.** In Section X-B, we complement the above result with a strong negative result showing that a number of well-known algorithms in the literature viz. Log-linear KLIEP [27, 3], the Gibbs distribution based estimator from [14, 15] and the MaxEnt algorithm [28, 29, 32, 35, 30] produce distributions Q that do not satisfy multi-group attribution. While they guarantee multiaccuracy (23), they do not guarantee either soundness or improvement.
4) **Experimental validation.** In Section XI, we test the performance of various algorithms for estimating KL divergence between distributions and the divergence when conditioned on various sub-populations, for mixtures of Gaussians, MNIST-based data and a weather dataset. Our results are in line with what the theory predicts: algorithms that guarantee multi-group attribution provide better KL estimates, both overall and for sub-populations.

We discuss additional related work in Section XII. Additional proofs and results are in the Appendix.

IX. Definitions

All distributions are over a domain $\mathcal{X}$. We work with discrete domains for simplicity, but all results can be generalized to the continuous setting. We say the importance weights w are explicit if the function $w : \mathcal{X} \to \mathbb{R}$ is computable efficiently; this does not necessarily require the pdf of Q to be explicit. For importance weights w, let $\|w\|_\infty = \max_{x \in \mathcal{X}}(w(x), 1/w(x))$. For $Q = w \cdot P$, we have $D(Q\|P) \in [0, \log(\|w\|_\infty)]$. Let $Q(C) = \Pr_Q[\mathbf{x} \in C]$, and let $Q|_C$ denote Q conditioned on C. We let $R = w^* \cdot P$.

a) Multi-group attribution.: We present a relaxation of Definition VIII.4 that allows slack in the equalities, which is inherent as we only get sample access to the distributions; since even checking that $R(C)$ and $P(C)$ are exactly equal is hard given samples.

Definition IX.1. *Let* $\alpha, \beta \geq 0$. Q *satisfies* (α, β) *multi-group attribution for* $(P, R, \mathcal{C})$ *if* $\forall C \in \mathcal{C}$,

$$\Big|Q(C) - R(C)\Big| \leq \alpha, \tag{32}$$

$$\begin{aligned}\Big|D(R|_C\|Q|_C) + D(Q|_C\|P|_C) - D(R|_C\|P|_C)\Big|& \\ \leq \beta/R(C).&\end{aligned} \tag{33}$$

We refer to these conditions as approximate multiaccuracy and the approximate Pythagorean property respectively. In the RHS of Equation (33), we normalize by $R(C)$. This means that Pythagorean property is meaningful only when $R(C)$ is not too small. This is inevitable in the sample access setting, where we cannot get meaningful bounds for very small sets in the Pythagorean property (in contrast to multiaccuracy). To see why, suppose we take $O(1/\alpha^2)$ samples from R and P, and don't see any samples lying in C. We can be confident that $R(C), P(C) \leq \alpha$, hence approximate mutliaccuracy holds. However, we cannot say anything about $D(R|_C\|P|_C)$, which could be anywhere in $[0, \log(\|w^*\|_\infty))$.

The reason we use two approximation parameters is that they are of different scale. Being the difference of probabilities, $\alpha \in [0, 1]$. It can be shown $\beta \in [0, 2\log(\|w^*\|_\infty)]$ for models Q where $\|w\|_\infty \leq \|w^*\|_\infty$. One can think of α, β as parameters that control the sample complexity. To achieve smaller values of α, β we need more samples, but this lets us get stronger guarantees, and reason about smaller sets C. Our bounds are meaningful for C when $R(C) \geq \alpha$.

The approximate Pythagorean property (33) implies approximate soundness and improvement:

$$D(R|_C\|P|_C) \geq D(Q|_C\|P|_C) - \beta/R(C), \tag{34}$$

$$D(R|_C\|P|_C) \geq D(R|_C\|Q|_C) - \beta/R(C). \tag{35}$$

b) Multiaccuracy.: If the distribution Q satisfies Equation (32), we say it is α-multiaccurate for $(R, \mathcal{C})$. The set of all α-multiaccurate distributions forms a convex polytope that we denote by $K^\alpha(R, \mathcal{C})$.

c) Partitions and multicalibration.: The following use of partitions to define importance weights is from [**?**].

Definition IX.2. *A partition $\mathcal{S} = \{S_1, \ldots, S_m\}$ of $\mathcal{X}$ is a collection of disjoint sets whose union is $\mathcal{X}$. Given distributions R, P, the $(R, \mathcal{S})$-reweighting of P is the distribution $Q = w \cdot P$ whose importance weights are $w(x) = w(S_i) = R(S_i)/P(S_i)$ for $x \in S_i$.*

The above importance weights satisfy $\|w\|_\infty \leq \|w^*\|_\infty$, indeed $\|w\|_\infty$ might be bounded even if $\|w^*\|_\infty$ is not. Every distribution Q on $\mathcal{X}$ induces a distribution on $\mathcal{S}$, let $\mathbf{S} \sim Q$ denote sampling from $\mathcal{S}$ according to it.

Lemma IX.3. *Let Q be the $(R, \mathcal{S})$ reweighting of P. Then Q and R induce identical distributions on $\mathcal{S}$. For every $i \in [m]$, $Q|_{S_i} = P|_{S_i}$.*

This lemma gives a natural coupling of Q and R: sample $\mathbf{S} \sim R$, and then sample $\mathbf{x} \sim P|_\mathbf{S}$ to generate a sample from Q and $\mathbf{x}' \sim R|_\mathbf{S}$ to generate a sample from R. We next define the notion of multi-calibrated partitions.

Definition IX.4. *Let $\mathcal{C}$ be a collection of subsets of $\mathcal{X}$. We say that the partition $\mathcal{S}$ is α-approximately multi-calibrated for $(P, R, \mathcal{C})$ if for all $C \in \mathcal{C}$,*

$$\mathop{\mathbb{E}}_{\mathbf{S} \sim R}\left[\left|R|_\mathbf{S}(C) - P|_\mathbf{S}(C)\right|\right] \leq \alpha. \tag{36}$$

The original notion of α-multi-calibration from [**?**] requires that for every $i \in [m]$ and $C \in \mathcal{C}$, $\left|R|_{S_i}(C) - P|_{S_i}(C)\right| \leq \alpha$. Our notion of α-approximate multi-calibration is weaker, since it only requires closeness of the conditional distributions on average, hence it is implied by α-multi-calibration. For $\mathcal{C}$ which is weakly agnostically learnable, the algorithm for (α, β)-multi-calibration in [**?**] can be used to compute an α-approximately multi-calibrated partition by setting $\beta = 1/\|w^*\|_\infty$. The number of states is $\mathrm{poly}(\log(\|w^*\|_\infty), 1/\alpha)$, and the running time is in time $\mathrm{poly}(\|w^*\|_\infty, 1/\alpha)$. We refer the reader to [52, 1**?**] for the definition of weak agnostic learning and a discussion of when it is reasonable.

X. Attribution from multi-calibration

The main theorem in this section is the following:

Theorem X.1. *If $\mathcal{S}$ is α-approximately multi-calibrated for $(P, R, \mathcal{C})$, then the $(R, \mathcal{S})$ reweighting of P, $Q = w \cdot P$ satisfies (α, β) multi-group attribution for $\mathcal{C}$ where $\beta = 2\alpha \log(\|w\|_\infty)$.*

Let Q be the $(R, \mathcal{S})$ reweighting of P. It is easy to show that approximate multi-calibration implies multi-accuracy (Lemma A.14 in Appendix A). The crux is to show the Pythagorean property. We describe the main technical ideas used in the proof. Multicalibration guarantees the closeness of the probability of belonging to C under Q and R conditioned on a (random) set S_i. The following lemma instead conditions on $C \in \mathcal{C}$ and considers the distributions induced by $R|_C$ and $Q|_C$ on the sets S_i in the partition, and shows that they are close assuming multicalibration.

Lemma X.2. *For every $C \in \mathcal{C}$, we have*

$$\left|\sum_{i \in [m]} R|_C(S_i) - Q|_C(S_i)\right| \leq \frac{2\alpha}{R(C)}. \tag{37}$$

Let us sketch how this helps prove the Pythagorean property. By Lemma VIII.5, it suffices to show that the random variable $\log(w)$ has similar expectations under $R|_C$ and $Q|_C$. By the definition of w, $\log(w)$ is constant on each $S_i \in \mathcal{S}$. Since Q and R induce statistically close distributions on $\mathcal{S}$ and $\log(w)$ is a bounded by $\log(\|w\|_\infty)$ we can bound the difference in expectation. Formally, we prove the following bound:

Lemma X.3. *For every $C \in \mathcal{C}$,*

$$\begin{aligned}&\left|D\left(R|_C \| P|_C\right) - D\left(R|_C \| Q|_C\right) - D\left(Q|_C \| P|_C\right)\right| \\ &\quad \leq (2\alpha/R(C)) \log(\|w\|_\infty).\end{aligned}$$

As discussed before, the degradation for small $R(C)$ is expected. Similarly, some dependence on $\|w\|_\infty$ is to be expected, since if $\|w\|_\infty$ is unbounded, then so are $D\left(R\|P\right)$ and $D\left(Q\|P\right)$. The proof of Theorem X.1 follows from Lemmas A.14 and X.3. For completeness, we include our overall algorithm for KL estimation in the next section.

A. Algorithm for KL estimation

Algorithm 1 is our full algorithm for estimating KL divergence. We first find a multi-calibrated partition $\mathcal{S}$, for which we use the algorithm of [**?**]. The estimation procedure is straightforward given this partition. Recall that we use the distribution Q as the model for R, where Q is defined in Lemma IX.3 as the distribution such that $Q(S) = R(S)$ and $Q|_S = P|_S$ for every set $S \in \mathcal{S}$. Therefore for any $x \in S$, $Q(x) = (R(S)/P(S))P(x)$ and the KL Divergence $D\left(Q\|P\right)$ which is our estimate for $D\left(R\|P\right)$ can be written as,

$$\begin{aligned} D\left(Q\|P\right) &= \sum_{S\in\mathcal{S}}\sum_{x\in S} Q(x)\log\left(\frac{Q(x)}{P(x)}\right) \\ &= \sum_{S\in\mathcal{S}}\sum_{x\in S}\frac{R(S)}{P(S)}P(x)\log\left(\frac{R(S)}{P(S)}\right) \\ &= \sum_{S\in\mathcal{S}} R(S)\log\left(\frac{R(S)}{P(S)}\right). \end{aligned}$$

Similarly, for any set C, the conditional KL divergence $D\left(Q|_C\|P|_C\right)$ which is our estimate for $D\left(R|_C\|P|_C\right)$ is given by

$$\begin{aligned} D\left(Q|_C\|P|_C\right) &= \sum_{S\in\mathcal{S}}\sum_{x\in S}\frac{R(S)P(x)}{P(S)Q(C)}\log\left(\frac{R(S)P(C)}{P(S)Q(C)}\right) \\ &= \sum_{S\in\mathcal{S}}\frac{R(S)P(S\cap C)}{P(S)Q(C)}\log\left(\frac{R(S)/R(C)}{P(S)/P(C)}\right). \end{aligned}$$

Note that $Q(C) = R(C)$ because of multiaccuracy (23), and $\frac{R(S)}{P(S)} = \frac{R(S\cap C)}{P(S\cap C)}$ because the partition $\mathcal{S}$ is multicalibrated with respect to $\mathcal{C}$ (Definition IX.4). Therefore,

$$\begin{aligned} D\left(Q|_C\|P|_C\right) &= \sum_{S\in\mathcal{S}}\frac{R(S\cap C)}{R(C)}\log\left(\frac{R(S)/R(C)}{P(S)/P(C)}\right) \\ &= \sum_{S\in\mathcal{S}}\frac{R(S\cap C)}{R(C)}\log\left(\frac{R(S\cap C)/R(C)}{P(S\cap C)/P(C)}\right). \end{aligned}$$

Algorithm 1: Algorithm to estimate conditional KL Divergence $D\left(R|_C\|P|_C\right)$ for distributions R and P and some set C belonging to a family $\mathcal{C}$. Choosing $C = \mathcal{X}$ (the entire domain) we find the unconditional KL divergence $D\left(R\|P\right)$.

```
// Finding a multi-calibrated partition
fit(samples from R, samples from P, family of sets $\mathcal{C}$):
    Use Algorithm 3 from [? ] to find a multicalibrated partition $\mathcal{S}$
    return: $\mathcal{S}$
// Estimating KL Divergence
estimate(samples from R, samples from P, set C, multi-calibrated partition $\mathcal{S}$):
    Let $\widehat{R}$ and $\widehat{P}$ be empirical distributions corresponding to samples from $R$ and $P$ respectively
    $kl_{est} \leftarrow 0$
    for every set $S \in \mathcal{S}$ do
        if $\widehat{P}'(S) \neq 0$ and $\widehat{R}'(S) \neq 0$ then
            $kl_{est} = kl_{est} + (\widehat{R}(S\cap C)/\widehat{R}(C))\log\left(\frac{\widehat{R}(S\cap C)/\widehat{R}(C)}{\widehat{P}(S\cap C)/\widehat{P}(C)}\right)$
    return: $kl_{est}$
```

B. (No) Attribution from multi-accuracy

We give an explicit example that shows that a number of popular algorithms in the literature do not satisfy multi-group attribution. We consider the following algorithms: MaxEnt [28, 29, 32, 35, 30], Log-linear KLIEP [27, 3] and divergence estimation using Gibbs distributions [14, 15]. In Appendix D we describe these algorithms in more detail and show they are essentially equivalent (an observation that we treat as folklore though we have not seen it stated explicitly). All these algorithms guarantee multiaccuracy. For e.g. the MaxEnt algorithm finds the α-multiaccurate distribution $Q^\alpha \in K^\alpha(R,\mathcal{C})$ which minimizes $D\left(Q^\alpha\|P\right)$. While the Pythagorean property was known for the distributions Q^α when $\alpha = 0$ without conditioning on $\mathcal{C}$ (see [29]), perhap surprisingly we show that no such bound holds when conditioned on $\mathcal{C}$; in fact even the soundness and improvement conditions implied by the Pythagorean property do not hold for Q^α. We state our result for $\alpha = 0$, it can be extended for any $\alpha \in [0, 1/4)$. We have not attempted to optimize the constant in the lower bound. Recall the definition of $d(p, q)$ in Equation (22).

Theorem X.4. *There exist distributions P, R on $\{0,1\}^2$, a collections of sets $\mathcal{C}$ and $C \in \mathcal{C}$ where $R(C) = 1/2$ and $D\left(R|_C \| P|_C\right) = 0$ but $D\left(R|_C \| Q^0|_C\right) = D\left(Q^0|_C \| P|_C\right) = d(3/4, 1/2)$. So Q^0 does not satisfy (α, β)-multigroup attribution for $\mathcal{C}$ for any $\beta < d(3/4, 1/2)$.*

Thus 0-multiaccuracy does not imply β close to 0 for (α, β) multi-group attribution. In contrast, 0-approximate multicalibration implies $\alpha = \beta = 0$ by Theorem X.1.

XI. Experiments

We compare the performance of the multi-calibrated algorithm to various common algorithms for estimating KL divergence between distributions and the divergence when conditioned on various sub-populations. We consider three data-sets, in the first both distributions are mixtures of Gaussians with a slight variation of parameters across both distribution. This is a simple synthetic data-set where the true KL-divergence could be analytically derived. The second data-set is based on MNIST-based data, the two distributions are different mixtures of odd and even digits. Here an approximation of the true divergence could be analytically derived. The third data set is real-world historical weather data for the city of Las Vegas as pulled from NOAA, where the ground-truth is unknown. Our results are in line with what the theory predicts: multi-calibration provides a more accurate estimate of the divergence, both overall and per sub-population. This is true across all data-sets, even the relatively simple mixture of Gaussians.

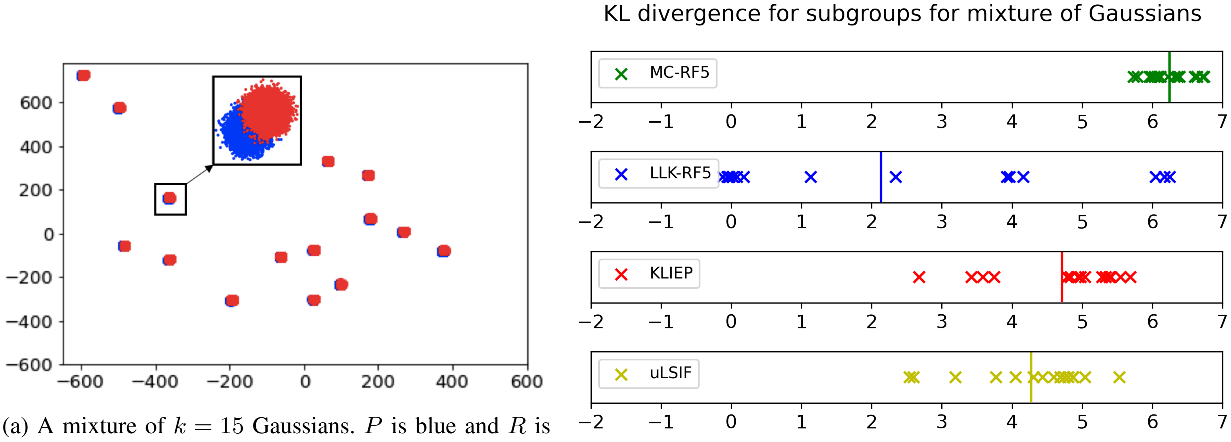
a) The Algorithms.: We test four algorithms: (i) KLIEP with code taken from [27]. KLIEP is a regression over the basic features of the data with a KL loss function; (ii) uLSIF from [53], which is a regression with least squares loss; (iii) an implementation of LL-KLIEP (denoted by LLK) aka log-linear KLIEP which satisfies multi-accuracy (Section X-B); (iv) an implementation of approximate multi-calibration as described in [**?**] (denoted by MC). The algorithm is an adaptation of the Boosting by branching programs algorithm [26, 51] to importance weight estimation. We implemented LL-KLIEP and MC in Python. While KLIEP and uLSIF regress over the basic features of the data, both LL-KLIEP and MC work with any family of binary base classifiers, effectively using the outputs of all classifiers in the family as the feature set. We test our algorithms using 3 different families of base classifiers $\mathcal{C}$, all using the standard implementation in sklearn: (1) Random Forest with depth 5 and 10 estimators (RF5), (2) a threshold function over a single feature (DT1, this is a decision tree with depth 1), (3) logistic regression (LR, we only test this for the MNIST data). We denote MC and LLK with logistic regression as MC-LR, LLK-LR and so on. Since our goal is to test the performance of the algorithms, and not the expressiveness of $\mathcal{C}$, it makes sense to use relatively simple classifiers for $\mathcal{C}$ and see how the algorithms perform in attributing divergence. Simple classes like conjunctions and shallow decision trees are of interest from a fairness perspective.

b) Mixtures of Gaussians.: Our goal is to create a simple pair of distributions P, R, where the notion of sub-populations is natural. To this end we set P to be a mixture of k equally-weighted d-dimensional Gaussians for $k = \{5, 10, 15\}$ and $d = 2$. Each component in the mixture has identity covariance, and their means are sampled from a Normal distribution with variance $k \cdot 10000 \cdot I$, therefore all the components are far apart. R is similar to P but with the means of all the Gaussians translated by 2.5 along both coordinates. See Fig. 1a for an example. We refer to each Gaussian in P and its shifted counterpart in Q as a pair. The divergence between every pair of Gaussians (such as those in the box in Fig. 1a) can be calculated by the closed form expression for the KL-divergence between shifted Gaussians, and is ≈ 9 for our parameters. Since the Gaussians are well separated (the KL divergence between any two components of P is > 500) this is also a good estimate for $D\left(R \| P\right)$. We generate $N = 30000$ samples from P and R to train the algorithms. We report results for larger values of N and d in Appendix D (we find them to be consistent with the results in this section). We first test how well the algorithms estimate $D\left(R \| P\right)$, without conditioning on sub-populations. The results in Table I show that MC performs well consistently for k and with different base classifiers.

TABLE I: KL estimates for mixture of k Gaussians, $N = 30000$, averaged over 5 trials

k	KL	LLK-RF5	MC-RF5	LLK-DT1	MC-DT1	KLEIP	uLSIF
5	9.02	$\mathbf{6.43 \pm 0.26}$	$\mathbf{6.59 \pm 0.22}$	1.97 ± 0.31	$\mathbf{6.38 \pm 0.51}$	5.41 ± 0.21	$\mathbf{6.68 \pm 0.01}$
10	9.02	3.64 ± 1.00	$\mathbf{6.03 \pm 0.36}$	0.68 ± 0.08	$\mathbf{6.05 \pm 0.33}$	5.02 ± 0.21	5.22 ± 0.10
15	9.02	2.23 ± 0.27	$\mathbf{5.92 \pm 0.04}$	0.36 ± 0.10	5.04 ± 0.58	4.96 ± 0.42	4.39 ± 0.14

Once the model had been fitted, we use it *without retraining* to estimate the divergence between pairs of Gaussians in the mixture. Formally this corresponds to conditioning on a set C which is a bounding box around the pair of Gaussians, as demonstrated in Fig. 1a. Since the Gaussians are far apart and have negligible overlap in their densities, $D\left(R_C \| P_C\right)$ is very close to the closed form divergence calculated earlier between the two shifted Gaussians (≈ 9). Results are summarized in Fig. 1b, where the x-axis indicates the divergence, and each mark indicates a pair of sub-populations, with the horizontal line indicating the average. We see that MC estimates the divergence across all sub-populations well, while all others have a fair



(a) A mixture of $k = 15$ Gaussians. P is blue and R is red. The zoomed-in box shows a sub-population—a pair of Gaussians from P and R.

(b) KL estimates for the 15 sub-populations.

Fig. 1: KL estimation for mixtures of Gaussians, $k = 15, d = 2, N = 30000$.

bit of variance in their estimates for each pair and seem to miss a lot of the divergence from some pairs. Also, as the theory predicts, all estimates are *lower bounds* on the true divergence.

We report additional results in Appendix D including the standard deviation of the divergences across the sub-populations (with error bars), and also contour plots which provide a clear visual explanation of how MC does better on sub-populations.

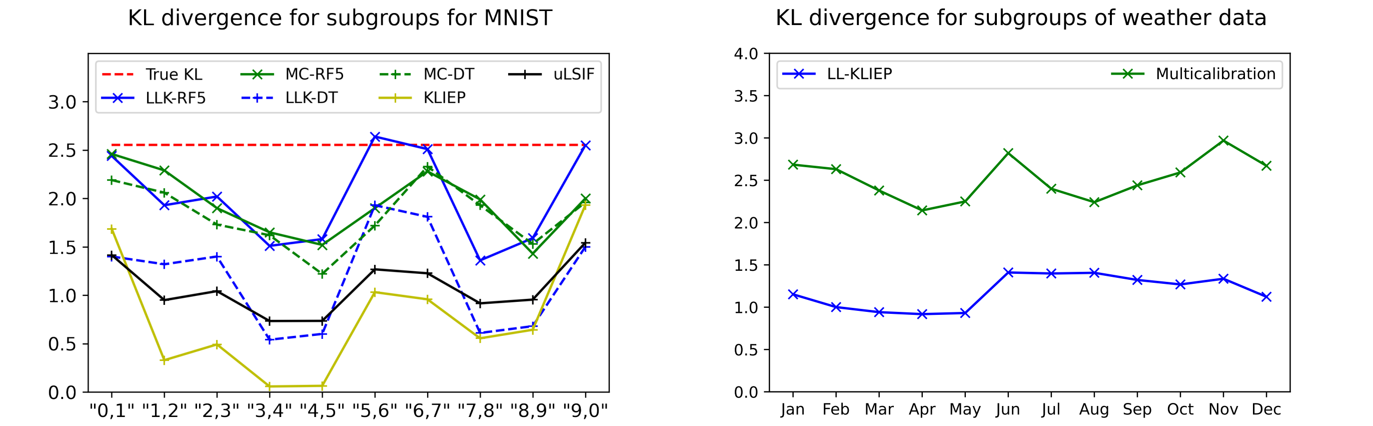
c) MNIST based data.: Our second set of experiments are based on MNIST images [54]. For a bias variable $\delta \in (0.5, 1)$ we define the distribution P_δ to sample an even digit with probability δ and an odd digit with probability $1 - \delta$. R_δ samples an odd digit with probability δ and an even digit with probability $1 - \delta$. Therefore, an upper bound on the true KL divergence $D\left(R_\delta \| P_\delta\right)$ is $D\left(R_\delta \| P_\delta\right) \leq d(\delta, 1 - \delta)$ (see Equation (22)). Note that both MC and LL-KLIEP are guaranteed to provide a lower bound on the true KL divergence (for any choice of the base family of classifiers), and hence the true KL-divergence must be sandwiched between these estimates. As before, we first test the algorithms on estimating the divergence across the entire population. The results are summarized in Table II. We see that MC and LL-KLIEP are quite similar when the base family of classifiers is a random forest. The quality of LL-KLIEP drops significantly when the classifier is weaker: depth 1 decision tree or logistic regression. We see that KLIEP and uLSIF are significantly worse.

TABLE II: KL estimation for MNIST

bias	True KL bound	LLK-RF5	MC-RF5	LLK-DT1	MC-DT1	LLK-LR	MC-LR	KLIEP	uLSIF
0.6	0.12	**0.07**	**0.07**	0.04	0.05	0.04	0.04	0.04	0.05
0.7	0.49	**0.36**	0.31	0.22	0.27	0.24	0.26	0.15	0.19
0.8	1.2	**0.92**	0.89	0.55	0.79	0.64	0.72	0.43	0.42
0.9	2.56	1.93	**1.96**	1.18	1.72	1.39	1.53	0.78	1.08
0.95	3.85	**2.96**	2.85	1.55	2.58	1.9	2.22	1.32	1.16

In this data set we consider single digits as sub-populations. Note that the digits are not perfectly classified by our base classifiers, so strictly speaking they are not part of subsets for which the algorithm is multi-calibrated. Our goal here is precisely to test how well our predictions hold in settings which do not strictly conform to our assumptions. In our experiment we set the bias $\delta = 0.9$. We then set a sub-population C to be images of two consecutive digits. Since one digit is odd and one digit is even we have $D\left(R_C \| P_C\right) = D\left(R_{0.9} \| P_{0.9}\right) \leq d(0.9, 0.1) = 2.56$ (as before MC and LL-KLIEP still provide lower bounds on $D\left(R_C \| P_C\right)$). We tested all 10 sub populations of this form. Results are depicted in Fig. 2a. We see MC and LL-KLIEP perform reasonably well, and much better than the rest of the algorithms. Interestingly, all algorithms struggle with the digits 4 and 8.

d) Weather data.: We analyzed historic weather data for the city of Las Vegas [55]. The data consists of nine weather features (min temp, max temp, humidity, wind, etc) measured once an hour. The first distribution consists of measurements from the decade $1980 - 1989$, the second distribution consists of measurements from the decade $2010 - 2019$, each data-set consists of roughly $85,000$ rows. Unlike the previous experiments, we do not have ground truth KL divergence estimates for this data. However, as remarked earlier both MC and LL-KLIEP provide estimates which are lower bounds on the true KL divergence. So we compare them by seeing which algorithm finds more divergence, more is better. The underlying base family of classifiers for both algorithms was Random Forests of depth 5 and 10 trees. The difference is pretty stark: while LL-KLIEP finds a divergence of 1.18, MC finds a divergence of 2.46. Fig. 2b has the KL estimates for the sub-populations corresponding to each month. We can see that MC also finds a higher divergence for each sub population. MC identifies June and November



(a) KL estimates for sub-groups, each sub-group is a pair of an odd and an even digit.

(b) KL estimations for sub-groups corresponding to the 12 months for weather data

Fig. 2: KL estimates for sub-groups for MNIST and weather data.

as being particularly affected months, while LL-KLIEP identifies the summer months at large with a higher divergence and finds practically no divergence in the winter months. Presumably most of the divergence across the distributions is due to climate change, and the MC algorithms indicate that climate change is significant. As a sanity check we also measure the divergence between the eighties and the nineties and found it to be insignificant at 0.18 and 0.2.

XII. Other Related Work

In Section VIII, we discussed some of the diverse applications of importance weights or density-ratio estimation which span many communities. We refer the reader to the book [42] for a more comprehensive overview of the related work, especially in the context of the machine learning literature. In addition to density ratio estimation, relevant results appear in the literature under the subject of divergence estimation [14, 15, 17, 18], learning max-Entropy distributions [28, 30] and domain adaptation [12, 13, 43]. Kernel based approaches for estimating importance weights have also been proposed, starting with Kernel Mean Matching (KMM) introduced in [44]. If the underlying kernel is universal, then under the limit of infinite data KMM provably recovers the true importance weights [45, 44]. In the limit of finite data however, kernel based approaches can be interpreted as generalizations of moment-matching methods such as [46] which seek to match some moments of the data [24]. In the context of our work, such based approaches guarantee multi-accuracy in expectation with respect to the moments or the feature space more generally.

Calibration has been well-studied in the statistics literature, in the context of forecasting [31]. It was introduced in the algorithmic fairness literature by [47]. The notion of multi-calibration as a multi-group fairness notion in supervised learning was introduced in [1] (see also [48]) and has subsequently generated significant interest [33, 40, 49, 50]. But all previous work to our knowledge has been in the supervised setting. Our work appears to be the first to introduce notions of group-fairness in unsupervised learning. Our algorithm for computing a multi-calibrated partition is inspired by the Boosting by Branching programs work of [26], which in turn builds on [51]. The work of [37] showed that the Mansour-McAllester algorithm can be viewed as an agnostic boosting algorithm, improving on the results of [36] who introduced the notion of agnostic boosting.

References

[1] Ú. Hébert-Johnson, M. P. Kim, O. Reingold, and G. N. Rothblum, "Multicalibration: Calibration for the (computationally-identifiable) masses," in *Proceedings of the 35th International Conference on Machine Learning, ICML*, 2018.

[2] S. Hido, Y. Tsuboi, H. Kashima, M. Sugiyama, and T. Kanamori, "Inlier-based outlier detection via direct density ratio estimation," in *2008 Eighth IEEE International Conference on Data Mining*, pp. 223–232, IEEE, 2008.

[3] M. Sugiyama, T. Suzuki, and T. Kanamori, *Density Ratio Estimation in Machine Learning*. Cambridge University Press, 2012.

[4] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM computing surveys (CSUR)*, vol. 41, no. 3, pp. 1–58, 2009.

[5] B. Schölkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, and R. C. Williamson, "Estimating the support of a high-dimensional distribution," *Neural computation*, vol. 13, no. 7, pp. 1443–1471, 2001.

[6] V. Hodge and J. Austin, "A survey of outlier detection methodologies," *Artificial intelligence review*, vol. 22, no. 2, pp. 85–126, 2004.

[7] A. Smola, L. Song, and C. H. Teo, “Relative novelty detection,” in *Artificial Intelligence and Statistics*, pp. 536–543, 2009.

[8] H. Shimodaira, “Improving predictive inference under covariate shift by weighting the log-likelihood function,” *Journal of statistical planning and inference*, vol. 90, no. 2, pp. 227–244, 2000.

[9] B. Zadrozny, “Learning and evaluating classifiers under sample selection bias,” in *Proceedings of the Twenty-First International Conference on Machine Learning*, p. 114, 2004.

[10] M. Sugiyama and K.-R. Müller, “Input-dependent estimation of generalization error under covariate shift,” *Statistics and Decisions-International Journal Stochastic Methods and Models*, vol. 23, no. 4, pp. 249–280, 2005.

[11] S. Bickel, M. Brückner, and T. Scheffer, “Discriminative learning for differing training and test distributions,” in *Proceedings of the 24th International Conference on Machine Learning*, pp. 81–88, 2007.

[12] C. Cortes, Y. Mansour, and M. Mohri, “Learning bounds for importance weighting,” in *Advances in neural information processing systems*, pp. 442–450, 2010.

[13] S. Ben-David and R. Urner, “On the hardness of domain adaptation and the utility of unlabeled target samples,” in *Algorithmic Learning Theory - 23rd International Conference, ALT*, 2012.

[14] X. Nguyen, M. J. Wainwright, and M. I. Jordan, “Nonparametric estimation of the likelihood ratio and divergence functionals,” in *2007 IEEE International Symposium on Information Theory*, pp. 2016–2020, IEEE, 2007.

[15] X. Nguyen, M. J. Wainwright, and M. I. Jordan, “Estimating divergence functionals and the likelihood ratio by convex risk minimization,” *IEEE Transactions on Information Theory*, vol. 56, no. 11, pp. 5847–5861, 2010.

[16] M. Yamada, T. Suzuki, T. Kanamori, H. Hachiya, and M. Sugiyama, “Relative density-ratio estimation for robust distribution comparison,” *Neural computation*, vol. 25, no. 5, pp. 1324–1370, 2013.

[17] Q. Wang, S. R. Kulkarni, and S. Verdú, “Divergence estimation of continuous distributions based on data-dependent partitions,” *IEEE Transactions on Information Theory*, vol. 51, no. 9, pp. 3064–3074, 2005.

[18] Q. Wang, S. R. Kulkarni, and S. Verdú, “Divergence estimation for multidimensional densities via k-nearest-neighbor distances,” *IEEE Transactions on Information Theory*, vol. 55, no. 5, pp. 2392–2405, 2009.

[19] M. Wornowizki and R. Fried, “Two-sample homogeneity tests based on divergence measures,” *Computational Statistics*, vol. 31, no. 1, pp. 291–313, 2016.

[20] T. Suzuki, M. Sugiyama, J. Sese, and T. Kanamori, “Approximating mutual information by maximum likelihood density ratio estimation,” in *New challenges for feature selection in data mining and knowledge discovery*, pp. 5–20, 2008.

[21] P. Rosenbaum and D. Rubin, “The central role of the propensity score in observational studies for causal effects,” *Biometrika*, vol. 70, pp. 41–55, 04 1983.

[22] T. Batu, L. Fortnow, R. Rubinfeld, W. D. Smith, and P. White, “Testing closeness of discrete distributions,” *J. ACM*, vol. 60, no. 1, pp. 4:1–4:25, 2013.

[23] P. Valiant, “Testing symmetric properties of distributions,” *SIAM J. Comput.*, vol. 40, no. 6, pp. 1927–1968, 2011.

[24] M. Sugiyama, T. Suzuki, and T. Kanamori, “Density-ratio matching under the bregman divergence: a unified framework of density-ratio estimation,” *Annals of the Institute of Statistical Mathematics*, vol. 64, no. 5, pp. 1009–1044, 2012.

[25] S. Arora and B. Barak, *Computational Complexity: A Modern Approach.* Cambridge University Press, 2006.

[26] Y. Mansour and D. McAllester, “Boosting using branching programs,” *Journal of Computer and System Sciences*, vol. 64, no. 1, pp. 103–112, 2002.

[27] M. Sugiyama, S. Nakajima, H. Kashima, P. V. Bunau, and M. Kawanabe, “Direct importance estimation for covariate shift adaptation,” *Annals of the Institute of Statistical Mathematics*, 2008.

[28] E. T. Jaynes, “Information theory and statistical mechanics,” *Physical review*, vol. 106, no. 4, p. 620, 1957.

[29] S. Della Pietra, V. Della Pietra, and J. Lafferty, “Inducing features of random fields,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 19, no. 4, pp. 380–393, 1997.

[30] M. Dudik, S. J. Phillips, and R. E. Schapire, “Maximum entropy density estimation with generalized regularization and an application to species distribution modeling,” *Journal of Machine Learning Research*, vol. 8, no. Jun, pp. 1217–1260, 2007.

[31] A. P. Dawid, “Objective probability forecasts,” *University College London, Dept. of Statistical Science. Research Report 14*, 1982.

[32] J. Kazama and J. Tsujii, “Evaluation and extension of maximum entropy models with inequality constraints,” in *EMNLP ’03: Proceedings of the 2003 conference on Empirical Methods in Natural Language Processing*, pp. 137–144, 01 2003.

[33] M. P. Kim, A. Ghorbani, and J. Zou, “Multiaccuracy: Black-box post-processing for fairness in classification,” in *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, pp. 247–254, 2019.

[34] T. Kanamori, T. Suzuki, and M. Sugiyama, “Theoretical analysis of density ratio estimation,” *IEICE transactions on fundamentals of electronics, communications and computer sciences*, vol. 93, no. 4, pp. 787–798, 2010.

[35] M. Dudik, S. J. Phillips, and R. E. Schapire, “Performance guarantees for regularized maximum entropy density estimation,” in *International Conference on Computational Learning Theory (COLT)*, pp. 472–486, Springer, 2004.

[36] S. Ben-David, P. M. Long, and Y. Mansour, “Agnostic boosting,” in *14th Annual Conference on Computational Learning Theory, COLT*, 2001.

[37] A. T. Kalai, Y. Mansour, and E. Verbin, "On agnostic boosting and parity learning," in *Proceedings of the 40th Annual ACM Symposium on Theory of Computing, Victoria, British Columbia, Canada, May 17-20, 2008*, pp. 629–638, ACM, 2008.

[38] V. Kanade and A. Kalai, "Potential-based agnostic boosting," *Advances in Neural Information Processing Systems*, vol. 22, pp. 880–888, 2009.

[39] V. Feldman, "Distribution-specific agnostic boosting," *arXiv preprint arXiv:0909.2927*, 2009.

[40] C. Jung, C. Lee, M. Pai, A. Roth, and R. Vohra, "Moment multicalibration for uncertainty estimation," in *Conference on Learning Theory*, pp. 2634–2678, PMLR, 2021.

[41] T. M. Cover and J. A. Thomas, *Elements of information theory (2. ed.)*. Wiley, 2006.

[42] M. Sugiyama, T. Suzuki, and T. Kanamori, *Density ratio estimation in machine learning*. Cambridge University Press, 2012.

[43] I. Redko, E. Morvant, M. Habrard, A.and Sebban, and Y. Bennani, *Advances in Domain Adaptation Theory*. Elsevier, 2019.

[44] J. Huang, A. Gretton, K. Borgwardt, B. Schölkopf, and A. J. Smola, "Correcting sample selection bias by unlabeled data," in *Advances in Neural Information Processing Systems*, pp. 601–608, 2007.

[45] C. Cortes, M. Mohri, M. Riley, and A. Rostamizadeh, "Sample selection bias correction theory," in *International Conference on Algorithmic Learning Theory (ALT)*, pp. 38–53, Springer, 2008.

[46] J. Qin, "Inferences for case-control and semiparametric two-sample density ratio models," *Biometrika*, vol. 85, no. 3, pp. 619–630, 1998.

[47] J. M. Kleinberg, S. Mullainathan, and M. Raghavan, "Inherent trade-offs in the fair determination of risk scores," in *8th Innovations in Theoretical Computer Science Conference, ITCS*, 2017.

[48] M. Kearns, S. Neel, A. Roth, and Z. S. Wu, "Preventing fairness gerrymandering: Auditing and learning for subgroup fairness," in *International Conference on Machine Learning*, pp. 2564–2572, 2018.

[49] N. Barda, D. Riesel, A. Akriv, J. Levy, U. Finkel, G. Yona, D. Greenfeld, S. Sheiba, J. Somer, E. Bachmat, *et al.*, "Developing a covid-19 mortality risk prediction model when individual-level data are not available," *Nature communications*, vol. 11, no. 1, pp. 1–9, 2020.

[50] C. Dwork, M. P. Kim, O. Reingold, G. N. Rothblum, and G. Yona, "Outcome indistinguishability," in *ACM Symposium on Theory of Computing (STOC'21)*, 2021.

[51] M. J. Kearns and Y. Mansour, "On the boosting ability of top-down decision tree learning algorithms," *J. Comput. Syst. Sci.*, vol. 58, no. 1, pp. 109–128, 1999.

[52] S. Ben-David, P. M. Long, and Y. Mansour, "Agnostic boosting," in *Computational Learning Theory, 14th Annual Conference on Computational Learning Theory, COLT 2001 and 5th European Conference on Computational Learning Theory, EuroCOLT 2001, Amsterdam, The Netherlands, July 16-19, 2001, Proceedings* (D. P. Helmbold and R. C. Williamson, eds.), vol. 2111 of *Lecture Notes in Computer Science*, pp. 507–516, Springer, 2001.

[53] T. Kanamori, S. Hido, and M. Sugiyama, "A least-squares approach to direct importance estimation," *The Journal of Machine Learning Research*, vol. 10, pp. 1391–1445, 2009.

[54] Y. LeCun and C. Cortes, "MNIST handwritten digit database," 2010.

[55] "Historic weather data." https://openweathermap.org.

Appendix

Proof of Lemma III.3. The first equivalence holds since we have $Q(C \cap S) = R(S)P(C|S)$, while $R(C \cap S) = R(S)R(C|S)$. We substitute these in Equation (8) and divide by $R(S)$ to derive Equation (9).

Applying Equation (1) to R with $A = C \cap S$, we get

$$R(C \cap S) = P(C \cap S) \mathop{\mathbb{E}}_{\mathbf{x} \sim P|_{C \cap S}}[w^*(x)]$$

whereas by the definition of Q we have

$$Q(C \cap S) = R(S)P(C|S) = \frac{R(S)P(C \cap S)}{P(S)} = w(S)P(C \cap S).$$

Hence the LHS of Equation (8) can be written as

$$\Big| R(C \cap S) - Q(C \cap S) \Big| = P(C \cap S) \left| w(S) - \mathop{\mathbb{E}}_{\mathbf{x} \sim P|_{C \cap S}}[w^*(x)] \right|$$

We derive Equation (10) by diving both sides of Equation (8) by $P(C \cap S)$. □

Next, we prove Theorem III.4. Using the formulation in Equation (4), we will analyze $\mathbb{E}_{\mathbf{x} \sim P|_C}[w(\mathbf{x})w^*(\mathbf{x})]$. The key steps are the next two technical lemmas Lemma A.1 and A.2.

Lemma A.1. *We have*

$$\left|\underset{\mathbf{x}\sim P|_C}{\mathbb{E}}[w(\mathbf{x})w^*(\mathbf{x})] - \underset{\mathbf{S}\sim P|_C}{\mathbb{E}}[w(\mathbf{S})^2]\right| \leq \frac{\alpha\left\|w\right\|_2^2}{P(C)}. \tag{38}$$

Proof of Lemma A.1. We sample from the distribution $P|_C$ in two steps:

1) We first sample $\mathbf{S} \in \mathcal{S}$ according to the marginal distribution induced by $P|_C$ where $\Pr[\mathbf{S} = S_i] = P(C \cap S_i)/P(C)$.
2) We then sample $\mathbf{x} \in \mathbf{S}$ according to $P|_{C\cap\mathbf{S}}$ so that $\Pr[\mathbf{x} = x] = P(x)/P(C \cap \mathbf{S})$.

This allows us to use the fact that $w(x) = w(\mathbf{S})$ remains constant within each set of the partition, and that approximate multi-calibration implies that $\mathbb{E}_{P|_{\mathbf{S}\cap C}}[w^*(x)]$ is close to $w(\mathbf{S})$ by Equation (10).

$$\underset{\mathbf{x}\sim P|_C}{\mathbb{E}}[w(\mathbf{x})w^*(\mathbf{x})] = \underset{\mathbf{S}\sim P|_C}{\mathbb{E}}\ \underset{\mathbf{x}\sim P|_{\mathbf{S}\cap S}}{\mathbb{E}}[w(\mathbf{x})w^*(\mathbf{x})] = \underset{\mathbf{S}\sim P|_C}{\mathbb{E}} w(\mathbf{S}) \underset{\mathbf{x}\sim P|_{\mathbf{S}\cap C}}{\mathbb{E}}[w^*(\mathbf{x})].$$

Hence using Equation (10) we have

$$\begin{aligned}
\left|\underset{\mathbf{x}\sim P|_C}{\mathbb{E}}[w(\mathbf{x})w^*(\mathbf{x})] - \underset{\mathbf{S}\sim P|_C}{\mathbb{E}}[w(\mathbf{S})^2]\right| &= \left|\underset{\mathbf{S}\sim P|_C}{\mathbb{E}}\left[w(\mathbf{S}) \underset{\mathbf{x}\sim P|_{\mathbf{S}\cap C}}{\mathbb{E}}[w^*(\mathbf{x})] - w(\mathbf{S})^2\right]\right| \\
&\leq \underset{\mathbf{S}\sim P|_C}{\mathbb{E}}\left[w(\mathbf{S})\left|\underset{\mathbf{x}\sim P|_{\mathbf{S}\cap C}}{\mathbb{E}}[w^*(\mathbf{x})] - w(\mathbf{S})\right|\right] \\
&\leq \underset{\mathbf{S}\sim P|_C}{\mathbb{E}}\left[w(\mathbf{S})\frac{\alpha R(\mathbf{S})}{P(\mathbf{S}\cap C)}\right] \\
&= \sum_{S\in\mathcal{S}} \frac{P(S\cap C)}{P(C)}\frac{R(S)}{P(S)}\frac{\alpha R(S)}{P(S\cap C)} \\
&= \sum_{S\in\mathcal{S}} \frac{\alpha R(S)^2}{P(S)P(C)} = \frac{\alpha\|w\|^2}{P(C)}.
\end{aligned}$$

□

Lemma A.2. *We have*

$$\left(\underset{x\sim P|_C}{\mathbb{E}}[w^*(\mathbf{x})]\right)^2 - 2\alpha\frac{R(C)}{P(C)} \leq \underset{\mathbf{S}\sim P|_C}{\mathbb{E}}[w(\mathbf{S})^2] \leq \underset{\mathbf{x}\sim P|_C}{\mathbb{E}}[w^*(\mathbf{x})^2] + 2\alpha\frac{\|w\|_2^2}{P(C)} \tag{39}$$

Proof of Lemma A.2. We start with the lower bound. By Equation (10) we have

$$\underset{\mathbf{S}\sim P|_C}{\mathbb{E}}[w(\mathbf{S})] \geq \underset{\mathbf{S}\sim P|_C}{\mathbb{E}}\left[\underset{\mathbf{x}\sim P|_{\mathbf{S}\cap C}}{\mathbb{E}}[w^*(\mathbf{x})] - \frac{\alpha R(S)}{P(C\cap S)}\right] = \underset{\mathbf{x}\sim P|_C}{\mathbb{E}}[w^*(\mathbf{x})] - \frac{\alpha}{P(C)}.$$

Using this bound and the convexity of x^2,

$$\underset{\mathbf{S}\sim P|_C}{\mathbb{E}}[w(\mathbf{S})^2] \geq \left(\underset{\mathbf{S}\sim P|_C}{\mathbb{E}}[w(\mathbf{S})]\right)^2 \geq \left(\underset{\mathbf{x}\sim P|_C}{\mathbb{E}}[w^*(\mathbf{x})] - \frac{\alpha}{P(C)}\right)^2 \geq \left(\underset{x\sim P|_C}{\mathbb{E}}[w^*(\mathbf{x})]\right)^2 - 2\alpha\frac{R(C)}{P(C)}.$$

We now show the upper bound.

$$\begin{aligned}
\underset{\mathbf{x}\sim P|_C}{\mathbb{E}}[w^*(\mathbf{x})^2] = \underset{\mathbf{S}\sim P|_C}{\mathbb{E}}\ \underset{\mathbf{x}\sim P|_{\mathbf{S}\cap C}}{\mathbb{E}}[w^*(\mathbf{x})^2] &\geq \underset{\mathbf{S}\sim P|_C}{\mathbb{E}}\left(\underset{\mathbf{x}\sim P|_{\mathbf{S}\cap C}}{\mathbb{E}}[w^*(\mathbf{x})]\right)^2 \\
&\geq \underset{\mathbf{S}\sim P|_C}{\mathbb{E}}\left[\left(w(S) - \frac{\alpha R(S)}{P(S\cap C)}\right)^2\right] \\
&\geq \underset{\mathbf{S}\sim P|_C}{\mathbb{E}}\left[w(S)^2 - 2w(S)\frac{\alpha R(S)}{P(S\cap C)}\right].
\end{aligned} \tag{40}$$

We have

$$\underset{\mathbf{S}\sim P|_C}{\mathbb{E}}\left[w(S)\frac{\alpha R(S)}{P(S\cap C)}\right] = \sum_{S\in\mathcal{S}}\frac{P(S\cap C)}{P(C)}\frac{R(S)}{P(S)}\frac{\alpha R(S)}{P(S\cap C)} = \sum_{S\in\mathcal{S}}\alpha\frac{R(S)^2}{P(S)P(C)} = \frac{\alpha\|w\|^2}{P(C)}.$$

Plugging this into Equation (40) gives

$$\underset{\mathbf{x}\sim P|_C}{\mathbb{E}}[w^*(\mathbf{x})^2] \geq \underset{\mathbf{S}\sim P|_C}{\mathbb{E}}[w(S)^2] - 2\frac{\alpha\|w\|^2}{P(C)}$$

which gives the desired upper bound. □

We now put these together to prove Theorem III.4.

Proof of Theorem III.4. We claim the following inequalities hold

$$\left(\mathop{\mathbb{E}}_{\mathbf{x}\sim P|_C}[w^*(x)]\right)^2 - \alpha\frac{\|w\|_2^2 + R(C)}{P(C)} \leq \mathop{\mathbb{E}}_{\mathbf{x}\sim P|_C}[w(\mathbf{x})w^*(\mathbf{x})] \leq \mathop{\mathbb{E}}_{\mathbf{x}\sim P|_C}[w^*(x)^2] + 3\alpha\frac{\|w\|_2^2}{P(C)}. \tag{41}$$

These are an immediate consequence of Lemma A.1 showing that $\mathbb{E}_{\mathbf{x}\sim P|_C}[w(\mathbf{x})w^*(\mathbf{x})]$ and $\mathbb{E}_{\mathbf{S}\sim P|_C}[w(\mathbf{S})^2]$ are close, and Lemma A.2 which gives a sandwiching bound for $\mathbb{E}_{\mathbf{S}\sim P|_C}[w(\mathbf{S})^2]$.

Equation (41) equivalent to Equation (11). To see this, we use the following equalities from Equation (5) and (6):

$$\mathop{\mathbb{E}}_{\mathbf{x}\sim P|_C}[w(\mathbf{x})w^*(\mathbf{x})] = \mathop{\mathbb{E}}_{\mathbf{x}\sim R|_C}[w(\mathbf{x})] \mathop{\mathbb{E}}_{\mathbf{x}\sim P|_C}[w^*(\mathbf{x})],$$
$$\mathop{\mathbb{E}}_{\mathbf{x}\sim P|_C}[w^*(\mathbf{x})^2] = \mathop{\mathbb{E}}_{\mathbf{x}\sim R|_C}[w^*(\mathbf{x})] \mathop{\mathbb{E}}_{\mathbf{x}\sim P|_C}[w^*(\mathbf{x})].$$

We plug these into Equation (41) and divide throughout by $\mathbb{E}_{\mathbf{x}\sim P|_C}[w^*(\mathbf{x})] = R(C)/P(C)$ to derive Equation (11) and complete the proof. □

In this section, we prove Theorem III.7 which asserts that for

$$\ell(\alpha, \beta, w) = \alpha\|w\|_2^2 + \sqrt{\beta}\|w\|_4^2$$

the following bounds hold for every $C \in \mathcal{C}$,

$$\mathop{\mathbb{E}}_{\mathbf{x}\sim P|_C}[w^*(\mathbf{x})] - \frac{2\ell(\alpha,\beta,w)}{R(C)} - \frac{2(\alpha+2\beta)}{P(C)} \leq \mathop{\mathbb{E}}_{\mathbf{x}\sim R|_c}[w(\mathbf{x})] \leq \mathop{\mathbb{E}}_{\mathbf{x}\sim R|_C}[w^*(\mathbf{x})] + \frac{3\ell(\alpha,\beta,w)}{R(C)}.$$

Throughout this section, we assume that $\mathcal{S} = \{S_1, \ldots, S_m, T_0, T_1\}$ is (α, β)-approximate multi-calibration for $(P, R, \mathcal{C})$. We will use $S \in \mathcal{S}$ to denote a generic set in the partition, which could one of the S_is or T_js. We will now prove a sequence of technical lemmas that will be used to prove our bounds.

We first formalize our claim that we can assume weights for $S \in \mathcal{S} \setminus \mathcal{T}$ may be assumed to be bounded, by showing that sets whose weights are outside this range have small probability under one of the distributions.

Lemma A.3. *Let $T \subseteq \mathcal{X}$ and $c \geq 1$ be such that $w(T) = R(T)/P(T) \geq c/\beta$. Then $P(T) \leq \beta/c$.*

Proof. Since $R(T)/P(T) \geq c/\beta$, we have $P(T) \leq \beta R(T)/c \leq \beta/c$. □

We provide the proof of Lemma III.6

Proof of Lemma III.6. The statistical distance bounds hold since P^h and P only differ on T_0 and $P^h(T_0) = P(T_0) \leq \beta$. We verify that the partition is multi-calibrated by showing that $|P(C|S) - R(C|S)| \leq \alpha$ for every state $S \in \mathcal{S}$. For any $i \in [m]$, we have

$$\left|R^h(C|S_i) - P^h(C|S_i)\right| = \left|R(C|S_i) - P(C|S_i)\right| \leq \alpha.$$

where the equality holds since since P^h and P (and R^h and R) are identical on the states S_i for $i \in [m]$ and the inequality is from Equation (13). The conditional distributions $P^h|_{T_0}$ and $R^h|_{T_0}$ are identical since they both equal $R|_{T_0}$ by construction. Hence $R^h(C|T_0) = P^h(C|T_0)$ for all $C \in \mathcal{C}$, so the condition holds. A similar argument holds for T_1. □

A corollary is that (α, β)-approximate multi-calibration implies $(\alpha + 2\beta)$-multi-accuracy.

Lemma A.4. *If $\mathcal{S}$ is (α, β)-approximately multi-calibrated for $(P, R, \mathcal{C})$, then the $(P, R, \mathcal{S})$-reweighted distribution Q is $\gamma = (\alpha + 2\beta)$-multi-accurate for $(P, R, \mathcal{C})$.*

Lemma A.5. *For all $C \in \mathcal{C}$, we have*

$$\sum_{i\in[m]} \frac{P(S_i \cap C)}{P(C)}\left(\frac{R(S_i \cap C)^2}{P(S_i \cap C)^2} - \frac{R(S_i)R(S_i \cap C)}{P(S_i)P(S_i \cap C)}\right) \geq -\frac{3\alpha\|w\|_2^2}{P(C)}, \tag{42}$$

$$\sum_{i\in[m]} \frac{P(S_i \cap C)}{P(C)}\left(\frac{R(S_i)R(S_i \cap C)}{P(S_i)P(S_i \cap C)} - \frac{R(S_i)^2}{P(S_i)^2}\right) \geq -\frac{\alpha\|w\|_2^2}{P(C)}. \tag{43}$$

Proof. Using the approximate multi-calibration condition (Equation (9)), we have

$$\frac{R(S_i \cap C)^2}{P(S_i \cap C)^2} \geq \left(\frac{R(S_i)}{P(S_i)} - \alpha\frac{R(S_i)}{P(S_i \cap C)}\right)^2 \geq \frac{R(S_i)^2}{P(S_i)^2} - 2\alpha\frac{R(S_i)^2}{P(S_i)P(S_i \cap C)}$$
$$\frac{R(S_i)R(S_i \cap C)}{P(S_i)P(S_i \cap C)} \leq \frac{R(S_i)}{P(S_i)}\left(\frac{R(S_i)}{P(S_i)} + \alpha\frac{R(S_i)}{P(S_i \cap C)}\right) = \frac{R(S_i)^2}{P(S_i)^2} + \alpha\frac{R(S_i)^2}{P(S_i)P(S_i \cap C)}.$$

Subtracting the two bounds and averaging over S_is we get

$$\begin{aligned}\sum_{i\in[m]}\frac{P(S_i\cap C)}{P(C)}\left(\frac{R(S_i\cap C)^2}{P(S_i\cap C)^2}-\frac{R(S_i)R(S_i\cap C)}{P(S_i)P(S_i\cap C)}\right) &\geq -3\alpha\sum_{i\in[m]}\frac{P(S_i\cap C)}{P(C)}\frac{R(S_i)^2}{P(S_i)P(S_i\cap C)}\\ &= -\frac{3\alpha}{P(C)}\sum_{i\in[m]}\frac{R(S_i)^2}{P(S_i)}\\ &\geq -\frac{3\alpha}{P(C)}\left\|w\right\|_2^2\end{aligned}$$

which proves Equation (42).

We now prove (43). By the approximate multi-calibration condition,

$$\begin{aligned}\sum_{i\in[m]}\frac{P(S_i\cap C)}{P(C)}\frac{R(S_i)R(S_i\cap C)}{P(S_i)P(S_i\cap C)} &\geq \sum_{i\in[m]}\frac{P(S_i\cap C)}{P(C)}\frac{R(S_i)}{P(S_i)}\left(\frac{R(S_i)}{P(S_i)}-\alpha\frac{R(S_i)}{P(S_i\cap C)}\right)\\ &= \sum_{i\in[m]}\frac{P(S_i\cap C)}{P(C)}\frac{R(S_i)^2}{P(S_i)^2}-\sum_{i\in[m]}\frac{R(S_i)^2}{P(C)P(S_i)}.\end{aligned}$$

Hence

$$\sum_{i\in[m]}\frac{P(S_i\cap C)}{P(C)}\left(\frac{R(S_i)R(S_i\cap C)}{P(S_i)P(S_i\cap C)}-\frac{R(S_i)^2}{P(S_i)^2}\right)\geq -\frac{\alpha\left\|w\right\|_2^2}{P(C)}.$$

□

Next we consider the T_0 term and show the following bounds

Lemma A.6. *For all $C\in\mathcal{C}$, we have*

$$\frac{P(T_0\cap C)}{P(C)}\left(\frac{R(T_0\cap C)^2}{P(T_0\cap C)^2}-\frac{R(T_0)R(T_0\cap C)}{P(T_0)P(T_0\cap C)}\right)\geq -\frac{\sqrt{\beta}\left\|w\right\|_4^2}{P(C)}, \tag{44}$$

$$\frac{P(T_0\cap C)}{P(C)}\left(\frac{R(T_0)R(T_0\cap C)}{P(T_0)P(T_0\cap C)}-\frac{R(T_0)^2}{P(T_0)^2}\right)\geq -\frac{\sqrt{\beta}\left\|w\right\|_4^2}{P(C)}. \tag{45}$$

Proof. If we have

$$\frac{R(T_0\cap C)}{P(T_0\cap C)}\geq\frac{R(T_0)}{P(T_0)}$$

then clearly both the LHSes are non-negative, hence both bounds hold. Assume this is not the case, then we have

$$\begin{aligned}\frac{P(T_0\cap C)}{P(C)}\left(\frac{R(T_0\cap C)^2}{P(T_0\cap C)^2}-\frac{R(T_0)R(T_0\cap C)}{P(T_0)P(T_0\cap C)}\right) &\geq -\frac{P(T_0\cap C)}{P(C)}\frac{R(T_0)R(T_0\cap C)}{P(T_0)P(T_0\cap C)}\\ &\geq -\frac{P(T_0\cap C)}{P(C)}\frac{R(T_0)^2}{P(T_0)^2}\\ &\geq -\frac{1}{P(C)}\frac{R(T_0)^2}{P(T_0)}\end{aligned}$$

and similarly

$$\frac{P(T_0\cap C)}{P(C)}\left(\frac{R(T_0)R(T_0\cap C)}{P(T_0)P(T_0\cap C)}-\frac{R(T_0)^2}{P(T_0)^2}\right)\geq -\frac{1}{P(C)}\frac{R(T_0)^2}{P(T_0)}. \tag{46}$$

We can bound this as

$$\frac{R(T_0)^2}{P(T_0)}=P(T_0)\frac{R(T_0)^2}{P(T_0)^2}=\left(P(T_0)\cdot P(T_0)w(T_0)^4\right)^{1/2}\leq\sqrt{\beta}\left\|w\right\|_4^2$$

where we use $P(T_0)\leq\beta$ and $P(T_0)w(T_0)^4\leq\left\|w\right\|_4^4$. Plugging this into Equation (46) completes the proof. □

Finally for the set T_1 we show the following.

Lemma A.7. *For all $C\in\mathcal{C}$, we have*

$$\frac{P(T_1\cap C)}{P(C)}\left(\frac{R(T_1\cap C)^2}{P(T_1\cap C)^2}-\frac{R(T_1)R(T_1\cap C)}{P(T_1)P(T_1\cap C)}\right)\geq -\frac{\beta}{P(C)}, \tag{47}$$

$$\frac{P(T_1\cap C)}{P(C)}\left(\frac{R(T_1\cap C)R(T_1)}{P(T_1\cap C)P(T_1)}-\frac{R(T_1)^2}{P(T_1)^2}\right)\geq -\frac{\beta}{P(C)}. \tag{48}$$

Proof. If

$$\frac{R(T_1 \cap C)}{P(T_1 \cap C)} \geq \frac{R(T_1)}{P(T_1)}$$

then both LHSs are non-negative, so the bound holds. Else,

$$\frac{R(T_1 \cap C)}{P(T_1 \cap C)} \leq \frac{R(T_1)}{P(T_1)} \leq 1$$

where the inequality is by second by the definition of T_1. So we have the lower bound

$$\frac{P(T_1 \cap C)}{P(C)} \left(\frac{R(T_1 \cap C)^2}{P(T_1 \cap C)^2} - \frac{R(T_1)R(T_1 \cap C)}{P(T_1)P(T_1 \cap C)} \right) \geq -\frac{P(T_1 \cap C)}{P(C)} \frac{R(T_1)R(T_1 \cap C)}{P(T_1)P(T_1 \cap C)} \geq -\frac{\beta}{P(C)}$$

since $P(T_1 \cap C) \leq \beta$, and the other two ratios are at most 1. This proves Equation (47). Equation (48) is shown similarly. □

Lemma A.8. *For all $C \in \mathcal{C}$, we have*

$$\mathop{\mathbb{E}}_{\mathbf{x} \sim R|_C}[w^*(\mathbf{x})] + \frac{3}{R(C)}(\alpha \|w\|_2^2 + \sqrt{\beta}\,\|w\|_4^2) \geq \mathop{\mathbb{E}}_{\mathbf{x} \sim R|_C}[w(\mathbf{x})|].$$

Proof. We first show the bound

$$\mathop{\mathbb{E}}_{\mathbf{x} \sim P|_C}[w^*(\mathbf{x})^2 - w(\mathbf{x})w^*(\mathbf{x})] \geq -\frac{3}{P(C)}(\alpha \|w\|_2^2 + \sqrt{\beta}\,\|w\|_4^2). \tag{49}$$

We have

$$\begin{aligned}
\mathop{\mathbb{E}}_{\mathbf{x} \sim P|_C}[w^*(\mathbf{x})^2] &= \mathop{\mathbb{E}}_{\mathbf{S} \sim P|_C} \mathop{\mathbb{E}}_{\mathbf{x} \sim P|_{\mathbf{S} \cap C}}[w^*(\mathbf{x})^2] \geq \mathop{\mathbb{E}}_{\mathbf{S} \sim P|_C} \left(\mathop{\mathbb{E}}_{\mathbf{x} \sim P|_{\mathbf{S} \cap C}}[w^*(\mathbf{x})] \right)^2 \\
&= \sum_{S \in \mathcal{S}} \frac{P(S \cap C)}{P(C)} \frac{R(S \cap C)^2}{P(S \cap C)^2} \\
&= \sum_{i \in [m]} \frac{P(S_i \cap C)}{P(C)} \frac{R(S_i \cap C)^2}{P(S_i \cap C)^2} + \sum_{j \in \{0,1\}} \frac{P(T_j \cap C)}{P(C)} \frac{R(T_j \cap C)^2}{P(T_j \cap C)^2}.
\end{aligned} \tag{50}$$

On the other hand,

$$\begin{aligned}
\mathop{\mathbb{E}}_{\mathbf{x} \sim P|_C}[w(x)w^*(\mathbf{x})] &= \mathop{\mathbb{E}}_{\mathbf{S} \sim P|_C} \left[w(\mathbf{S}) \mathop{\mathbb{E}}_{\mathbf{x} \sim P|_{\mathbf{S} \cap C}}[w^*(\mathbf{x})] \right] = \sum_{S \in \mathcal{S}} \frac{P(S \cap C)}{P(C)} \frac{R(S)}{P(S)} \frac{R(S \cap C)}{P(S \cap C)} \\
&= \sum_{i \in [m]} \frac{P(S_i \cap C)}{P(C)} \frac{R(S_i)R(S_i \cap C)}{P(S_i)P(S_i \cap C)} + \sum_{j \in \{0,1\}} \frac{P(T_j \cap C)}{P(C)} \frac{R(T_j)R(T_j \cap C)}{P(T_j)P(T_j \cap C)}.
\end{aligned} \tag{51}$$

We subtract the Equation (51) from (50). We then apply the lower bounds from Equation (42) to bound the contribution from the S_is, Equation (45) for T_0 and Equation (48) for T_1 to get

$$\begin{aligned}
\mathop{\mathbb{E}}_{\mathbf{x} \sim P|_C}[w^*(\mathbf{x})^2 - w(\mathbf{x})w^*(\mathbf{x})] &\geq -\frac{1}{P(C)}(3\alpha \,\|w\|_2^2 + \sqrt{\beta}\,\|w\|_4^2 + \beta) \\
&\geq -\frac{3}{P(C)}(\alpha \,\|w\|_2^2 + \sqrt{\beta}\,\|w\|_4^2)
\end{aligned}$$

which proves the bound claimed in Equation (49).

To derive the claim from this, we use the following equalities from Equation (5) and (6):

$$\mathop{\mathbb{E}}_{\mathbf{x} \sim P|_C}[w(\mathbf{x})w^*(\mathbf{x})] = \mathop{\mathbb{E}}_{\mathbf{x} \sim R|_C}[w(\mathbf{x})] \frac{R(C)}{P(C)},$$

$$\mathop{\mathbb{E}}_{\mathbf{x} \sim P|_C}[w^*(\mathbf{x})^2] = \mathop{\mathbb{E}}_{\mathbf{x} \sim R|_C}[w^*(\mathbf{x})] \frac{R(C)}{P(C)}.$$

Plugging these into Equation (49) gives

$$\frac{R(C)}{P(C)} \mathop{\mathbb{E}}_{\mathbf{x} \sim R|_C}[w^*(x) - w(x)] \geq -\frac{3}{P(C)}(\alpha \,\|w\|_2^2 + \sqrt{\beta}\,\|w\|_4^2)$$

which gives the claimed bound upon rearranging. □

Lemma A.9. *For all $C \in \mathcal{C}$, we have*

$$\mathop{\mathbb{E}}_{\mathbf{x} \sim R|_C}[w(\mathbf{x})] + \frac{2}{R(C)}(\alpha \|w\|_2^2 + \sqrt{\beta}\,\|w\|_4^2) + \frac{2(\alpha + 2\beta)}{P(C)} \geq \mathop{\mathbb{E}}_{\mathbf{x} \sim P|_C}[w^*(\mathbf{x})|].$$

Proof. Recall that by Equation (51)

$$\mathop{\mathbb{E}}_{\mathbf{x}\sim P|_C}[w(x)w^*(\mathbf{x})] = \sum_{i\in[m]} \frac{P(S_i\cap C)}{P(C)}\frac{R(S_i)R(S_i\cap C)}{P(S_i)P(S_i\cap C)} + \sum_{j\in\{0,1\}} \frac{P(T_j\cap C)}{P(C)}\frac{R(T_j)R(T_j\cap C)}{P(T_j)P(T_j\cap C)}.$$

Recall the bounds from Equations (43), (45) and (48) which state

$$\begin{aligned}\sum_{i\in[m]} \frac{P(S_i\cap C)}{P(C)}\left(\frac{R(S_i)R(S_i\cap C)}{P(S_i)P(S_i\cap C)} - \frac{R(S_i)^2}{P(S_i)^2}\right) &\geq -\frac{\alpha\left\|w\right\|_2^2}{P(C)}\\ \frac{P(T_0\cap C)}{P(C)}\left(\frac{R(T_0)R(T_0\cap C)}{P(T_0)P(T_0\cap C)} - \frac{R(T_0)^2}{P(T_0)^2}\right) &\geq -\frac{\sqrt{\beta}\left\|w\right\|_4^2}{P(C)}\\ \frac{P(T_1\cap C)}{P(C)}\left(\frac{R(T_1\cap C)R(T_1)}{P(T_1\cap C)P(T_1)} - \frac{R(T_1)^2}{P(T_1)^2}\right) &\geq -\frac{\beta}{P(C)}.\end{aligned}$$

Adding these bounds, we get

$$\sum_{S\in\mathcal{S}} \frac{P(S\cap C)}{P(C)}\frac{R(S)R(S\cap C)}{P(S)P(S\cap C)} \geq \sum_{S\in\mathcal{S}} \frac{P(S\cap C)}{P(C)}\frac{R(S)^2}{P(S)^2} - \frac{2}{P(C)}(\alpha\left\|w\right\|_2^2 + \sqrt{\beta}\left\|w\right\|_4^2). \tag{52}$$

We also have

$$\sum_{S\in\mathcal{S}} \frac{P(S\cap C)}{P(C)}\frac{R(S)^2}{P(S)^2} \geq \left(\sum_{S\in\mathcal{S}} \frac{P(S\cap C)}{P(C)}\frac{R(S)}{P(S)}\right)^2. \tag{53}$$

But note that

$$\sum_{S\in\mathcal{S}} P(S\cap C)\frac{R(S)}{P(S)} = \sum_{S\in\mathcal{S}} R(S)P(C|S) = Q(C) \geq R(C) - \alpha - 2\beta$$

by Lemma A.4 showing that (α,β)-approximate multi-calibration implies $(\alpha+2\beta)$-multi-accuracy. Plugging this into Equation (53) gives

$$\sum_{S\in\mathcal{S}} \frac{P(S\cap C)}{P(C)}\frac{R(S)^2}{P(S)^2} \geq \left(\frac{R(C)-\alpha-2\beta}{P(C)}\right)^2 \geq \left(\frac{R(C)}{P(C)}\right)^2 - 2(\alpha+2\beta)\frac{R(C)}{P(C)^2}. \tag{54}$$

Putting Equations (52) and (53) together with Equation (51) gives

$$\mathop{\mathbb{E}}_{\mathbf{x}\sim P|_C}[w(\mathbf{x})w^*(\mathbf{x})] \geq \left(\mathop{\mathbb{E}}_{\mathbf{S}\sim P|_C}[w^*(x)]\right)^2 - \frac{2}{P(C)}(\alpha\left\|w\right\|_2^2 + \sqrt{\beta}\left\|w\right\|_4^2) - 2(\alpha+2\beta)\frac{R(C)}{P(C)^2}.$$

Using Equations (5) and (6) and diving both sides by $R(C)/P(C)$ gives

$$\mathop{\mathbb{E}}_{\mathbf{x}\sim R|_C}[w(\mathbf{x})] \geq \mathop{\mathbb{E}}_{\mathbf{x}\sim P|_C}[w^*(\mathbf{x})] - \frac{2}{R(C)}(\alpha\left\|w\right\|_2^2 + \sqrt{\beta}\left\|w\right\|_4^2) - \frac{2(\alpha+2\beta)}{P(C)}.$$

□

In this section we will show that the importance weights w^α found by the solution to the program in (64) (the problem solved by MaxEnt, which is equivalent to the problem solved by KLIEP) need not satisfy the sandwiching bounds. Indeed, for either direction of the sandwiching bound, we will show instances where the inequality is off by an arbitrarily large constant factor. Thus while one would like $\mathbb{E}_{P|_C}[w^*(x)] \leq \mathbb{E}_{R|_C}[w(x)]$, we will exhibit P, R and $\mathcal{C}$ such that the importance weights w^α found by MaxEnt are such that the ratio $\mathbb{E}_{P|_C}[w^*(x)]/\mathbb{E}_{R|_C}[w(x)]$ is arbitrarily large, and similarly for the upper bound. Both our counterexamples work by starting with a small example on $\{0,1\}^2$ that shows some small constant gap and then tensoring to amplify the gap.

Lemma A.10. *There exist distribution P, R on $\{0,1\}^2$, a collections of sets $\mathcal{C}$ and $C\in\mathcal{C}$ such that the MaxEnt algorithm run on $(P,R,\mathcal{C})$ with any $\alpha\geq 0$ returns a distribution Q^α with importance weights w^α such that*

$$\mathop{\mathbb{E}}_{\mathbf{x}\sim P|_C}[w^*(\mathbf{x})] > \mathop{\mathbb{E}}_{\mathbf{x}\sim R|_C}[w^\alpha(\mathbf{x})].$$

Proof. We first consider the case when $\alpha = 0$. Let P be the uniform distribution on $\{0,1\}^2$. Let R be the distribution where

$$R(00) = 0, R(01) = R(10) = 3/8, R(11) = 1/4.$$

We denote the two coordinates x_0, x_1, and let $\mathcal{C}$ consist of all subcubes of dimension 1. Hence $\mathcal{C} = \{x : x_i = a\}_{i\in\{0,1\},a\in\{0,1\}}$.

The distribution Q^α for $\alpha = 0$ is the product distribution which matches the marginal distributions on each coordinate: $Q^\alpha(x_0 = 1) = Q^\alpha(x_1 = 1) = 5/8$ and the coordinates are independent. The multi-accuracy constraints $Q^\alpha(x_0 = 1) = R(x_0 = 1)$ and $Q^\alpha(x_1 = 1) = R(x_1 = 1)$ are clearly satisfied, and Q^α is the maximum entropy distribution satisfying these constraints.

We can compute the following importance weights

1) $w^\alpha(11) = (5/8)^2/(1/2)^2 = 25/16$ whereas $w^*(11) = 1$.
2) $w^\alpha(10) = (5/8 \cdot 3/8)/(1/2)^2 = 15/16$, whereas $w^*(10) = (3/8)/(1/4) = 3/2$; ditto for 01.

For intuition as to why this is a gap example, note that this shows that while w^* assigns high weights to 01 and 10, w^α assigns these points weights less than 1, and instead assigns a high weight to 11. Thus an algorithm that was labelling points with w^α exceeding 1 as anomalies would report 11 as the sole anomaly, and miss both 01 and 10.

We consider the set $C = \{10, 11\} = \{x : x_0 = 1\}$. Note that $P|_C$ is uniform on $x_1 \in \{0, 1\}$, whereas $R|_C(x_1 = 1) = 2/5$. Then it follows that

$$\mathop{\mathbb{E}}_{\mathbf{x}\sim R|_C}[w^\alpha(\mathbf{x})] = 3/5 \cdot 15/16 + 2/5 \cdot 25/16 = 19/16$$

$$\mathop{\mathbb{E}}_{\mathbf{x}\sim P|_C}[w^*(\mathbf{x})] = 1/2 \cdot 1 + 1/2 \cdot 3/2 = 5/4$$

hence $\mathbb{E}_{\mathbf{x}\sim P|_C} w^*(\mathbf{x}) > \mathbb{E}_{\mathbf{x}\sim R|_C} w^\alpha(\mathbf{x})$.

For the case $\alpha > 0$, first note that the maximum entropy distribution is still a product distribution since conditioning reduces entropy. Secondly, as we increase α the bias of the individual coordinates in Q moves towards $1/2$, but this only makes the gap larger (since $\mathbb{E}_{\mathbf{x}\sim R|_C}[w^\alpha(\mathbf{x})] = 1$ if $Q^\alpha = P$). □

Intuitively, in the example above, while Q^α assigns the right weight of $5/8$ to the set C, within C the distribution of weight is misaligned with R, leading to low expected weight under $R|_C$. We now tensor this example to amplify the gap.

Theorem A.11. *For any constant $B > 1$, there exist distributions P, R on $\{0, 1\}^n$, a collections of sets $\mathcal{C}$ and $C \in \mathcal{C}$ such that the MaxEnt algorithm run on $(P, R, \mathcal{C})$ with any $\alpha \geq 0$ returns a distribution Q^α with importance weights w^α such that*

$$\mathop{\mathbb{E}}_{\mathbf{x}\sim P|_C}[w^*(\mathbf{x})] > B \mathop{\mathbb{E}}_{\mathbf{x}\sim R|_C}[w^\alpha(\mathbf{x})].$$

Proof. We now consider the k-wise tensor of the instances constructed in Lemma A.10. The domain is $\{0, 1\}^{2k}$ where the coordinates are denoted $x_0, \ldots, x_{2k-1}$. We consider the pair of distributions $P_k = (P)^k$ which is uniform on $2k$ bits, $R_k = (R)^k$ which the product of k independent copies of R on the pairs $\{x_{2i}x_{2i+1}\}_{i=1}^{k-1}$. Let $\mathcal{C}_k$ consist of all subcubes of dimension k where we restrict one co-ordinate out of x_{2i}, x_{2i+1} for $i \in \{0, \ldots, k-1\}$. One can verify that MaxEnt returns $Q_k^\alpha = (Q^\alpha)^k$ which is just the product distribution on $\{0, 1\}^{2k}$ with $\Pr[x_i = 1] = 5/8$ for every coordinate.

Let w_k^α and $w_k^*(x)$ denote the importance weights of Q^α and R_k with respect to P_k. A key observations is that importance weights tensor: for any $x \in \{0, 1\}^{2k}$,

$$w_k^*(x) = \frac{R_k(x)}{P_k(x)} = \prod_{i=0}^{k-1} \frac{R(x_{2i}x_{2i+1})}{P(x_{2i}x_{2i+1})} = \prod_{i=0}^{k-1} w^*(x_{2i}x_{2i+1})$$

$$w_k^\alpha(x) = \frac{Q_k^\alpha(x)}{P_k(x)} = \prod_{i=0}^{k-1} \frac{Q^\alpha(x_{2i}x_{2i+1})}{P(x_{2i}x_{2i+1})} = \prod_{i=0}^{k-1} w^\alpha(x_{2i}x_{2i+1}).$$

We consider the set $C = \{x : x_{2i} = 1, i \in \{0, \ldots, k-1\}\}$. The key property of this set is that the conditional distributions $P_k|_C = (P|_C)^k$ and $R_k|_C = (R|_C)^k$ are also product distributions of the conditional distributions. Hence we have

$$\mathop{\mathbb{E}}_{\mathbf{x}\sim R_k|_C}[w_k^\alpha(\mathbf{x})] = \mathop{\mathbb{E}}_{\mathbf{x}\sim R_k|_C}\left[\prod_{i=0}^{k-1} w^\alpha(\mathbf{x}_{2i}\mathbf{x}_{2i+1})\right] = \prod_{i=1}^{k-1} \mathop{\mathbb{E}}_{\mathbf{x}_{2i}\mathbf{x}_{2i+1}\sim R|_C}[w^\alpha(\mathbf{x}_{2i}\mathbf{x}_{2i+1})] = (19/16)^k$$

$$\mathop{\mathbb{E}}_{\mathbf{x}\sim P_k|_C}[w_k^*(\mathbf{x})] = \mathop{\mathbb{E}}_{\mathbf{x}\sim P_k|_C}\left[\prod_{i=0}^{k-1} w^*(\mathbf{x}_{2i}\mathbf{x}_{2i+1})\right] = \prod_{i=1}^{k-1} \mathop{\mathbb{E}}_{\mathbf{x}_{2i}\mathbf{x}_{2i+1}\sim P|_C}[w^*(\mathbf{x}_{2i}\mathbf{x}_{2i+1})] = (5/4)^k.$$

Now take k sufficiently large so that $(5/4)^k > B(19/16)^k$. □

We now construct a gap example for the other direction of the sandwiching bounds, where $\mathbb{E}_{R|_C}[w(x)] > \mathbb{E}_{R_C}[w^*(x)]$. Again we start with a small constant gap and amplify it by tensoring. We will only describe the construction for achieving the small constant gap, the tensoring step is identical to Theorem X.4.

Theorem A.12. *For any constant $B > 1$, there exist distributions P, R on $\{0, 1\}^n$, a collections of sets $\mathcal{C}$ and $C \in \mathcal{C}$ such that the MaxEnt algorithm run on $(P, R, \mathcal{C})$ with any $\alpha \geq 0$ returns a distribution Q^α with importance weights w^α such that*

$$\mathop{\mathbb{E}}_{\mathbf{x}\sim R|_C}[w^\alpha(\mathbf{x})] > B \mathop{\mathbb{E}}_{\mathbf{x}\sim R|_C}[w^*(\mathbf{x})].$$

Proof. We consider the case $\alpha = 0$, the general case follows as in the proof of Lemma A.10 by a suitable choice of parameters. As before let P be uniform on $\{0,1\}^2$. Consider the distribution R given by

$$R(00) = 2/16, R(10) = 6/16, R(01) = 3/16, R(11) = 5/16.$$

As before we let $\mathcal{C}$ consist of all subcubes of dimension 1. The distribution Q^α is the product distribution on x_0 and x_1 where $\Pr[x_0 = 1] = 11/16$ and $\Pr[x_1 = 1] = 1/2$. We will use the set $C = \{01, 11\}$, so that $R|_C(01) = 3/8, R|_C(11) = 5/8$.

We compute the importance weights within C as follows:

$$w^*(01) = 3/4, w^*(11) = 5/4$$
$$w^\alpha(01) = 5/8, w^\alpha(11) = 11/8.$$

Hence we have the conditional expectations

$$\mathop{\mathbb{E}}_{\mathbf{x} \sim R|_C}[w^*(\mathbf{x})] = 3/8 \cdot 3/4 + 5/8 \cdot 5/4 = 34/32.$$
$$\mathop{\mathbb{E}}_{\mathbf{x} \sim R|_C}[w^\alpha(\mathbf{x})] = 3/8 \cdot 5/8 + 5/8 \cdot 11/8 = 35/32$$

hence $\mathbb{E}_{\mathbf{x} \sim R|_C}[w^\alpha(\mathbf{x})] > \mathbb{E}_{\mathbf{x} \sim R|_C}[w^*(\mathbf{x})]$. We can amplify this gap by tensoring. □

Theorem A.13. *Algorithm 3 returns a partition $\mathcal{S}$ that is (α, β)-approximately multi-calibrated for $\mathcal{C}$ under P, R.*

Proof. We first prove that $P(T_0) \le \beta$ and $P(T_0) \le R(T_0)$. We can write $T_0 = \cup_i T'_i \cup_j S'_j$ where the sets T'_i were added to T_0 during the loop, when they were created during a Split operation, and the sets S'_j were moved from $\mathcal{S}^t$ in the post-processing step. Then $R(T'_j)/P(T'_j) \ge 2/\beta$ for all j, hence $R(\cup_j T'_j)/P(\cup_j T'_j) \ge 2/\beta$. But by Lemma A.3, this implies that $P(\cup_j T'_j) \le \beta/2$. The sets S'_j are added to T_0 because $P(S'_j) \le \beta/4m$. Since there are at most $2m$ such sets (else we would have run Merge), we have $P(\cup_j S'_j) \le 2m\beta/4m \le \beta/2$. Overall

$$P(T_0) \le P(\cup_i T'_i) + P(\cup_j S'_j) \le \beta/2 + \beta/2 = \beta.$$

Further, for every set T merged into T_0 it holds that $P(T) \le R(T)$ and therefore $P(T_0) \le R(T_0)$. A similar argument shows that $R(T_1) \le \beta$ and $R(T_1) < P(T_1)$.

We need to show that every set $S \in \mathcal{S}^t$ satisfies $\|R(C|S) - P(C|S)\| \le \alpha$ for all $C \in \mathcal{C}$. Note that S satisfies $R(S) \ge \beta/4m$ and $P(S) \ge \beta/4m$, else it would have been removed from $\mathcal{S}^t$ in the post-processing step. Hence, if it violates this condition, the weak agnostic learner would find a $C' \in \mathcal{C}$ such that $\|R(C'|S) - P(C'|S)\| \ge \alpha'$, so we would not exit the loop at the t^{th} iteration.

This shows that the partition $\mathcal{S}$ is (α, β)-approximately multi-calibrated. □

Next we analyze the running time and sample complexity, proving Theorem V.4

Proof of Theorem V.4. Each iteration but the last involves one call to either Split or Merge. We bound the number of calls to Merge, denoted ℓ. Assume the merge operations happen in interations $t^1 < t^2 \cdots < t^\ell)$. Every Split operation increases the number of states by 1, whereas Merge reduces it from $2m$ to a number is the range $\{1, \ldots, m\}$. Hence $2m \le t^{k+1} - t^k \ge m$. Each Split operation acts on a set S where $R(S) \ge \beta/4m$, and by Lemma V.2, it increase the KL divergence by $4R(S)\alpha'^2$. The Merge operation decreases it by $\delta = \alpha'^2\beta/2$. Hence we have

$$D\left(Q_{t^{k+1}} \| P\right) - D\left(Q_{t^k} \| P\right) \ge m\frac{\beta}{4m}4\alpha'^2 - \delta = \delta.$$

Thus the KL divergence between successive Merge operations increases by δ. We start with the trivial partition, so $Q^1 = P$. Since $\mathcal{S}^T$ is partition, if Q^T denotes the corresponding reweighted distribution, then $D\left(Q^T \| P\right) \le D\left(R \| P\right)$. Hence

$$\ell\delta \le D\left(Q^T \| P\right) - D\left(Q^1 \| P\right) \le D\left(R \| P\right)$$

hence $\ell \le D\left(R \| P\right)/\delta$. The total number of iterations is bounded by

$$T \le (2m+1)\ell = O(\log(1/\beta) D\left(R \| P\right)/\delta^2) = \widetilde{O}(D\left(R \| P\right)/(\beta^2\alpha'^4)).$$

For one Split iteration, we might make $O(m)$ calls to the weak learner, one per state to find the pair S, C on which to run Split. However, once we fail to find a good C for S, we do not need to try S again until the state is modified, which cannot happen before the next Merge iteration. This shows that there are at most $4m$ calls to the agnostic learner between two merge operations, $2m$ successful ones and $2m$ unsuccessful ones. Hence the number of calls to the learner is bounded by $4m\ell = O(T)$.

Finally, we address the sample complexity. We need to run the learner on the distributions $P|_S$ and $R|_S$ where $R(S), P(S) \ge \beta/4m$. If the sample complexity of the agnostic learner is L then $O(Lm/\beta) = \widetilde{O}(L/(\alpha'\beta)^2)$ samples from each of P and R will suffice to ensure that we have sufficiently many samples from $P|_S$ and $R|_S$ respectively. □

Finally we note that the partition we compute can be represented by a $\mathcal{C}$-branching program where each node is labelled by $c \in \mathcal{C}$.

A. Proofs from Section VIII

Proof of Proposition VIII.1. The proof follows by a simple application of the birthday paradox, we note that it is possible to prove stronger lower bounds using better constructions [22, 23].

We first consider the case when P and R are both uniform distributions supported on half the domain (and the supports of both P and R are unknown to the algorithm). Note that by the birthday paradox, we not not expect to see any repetitions in $\sqrt{\mathcal{X}}/10$ samples drawn from a uniform distribution over a support of size $|\mathcal{X}|/2$, with probability $9/10$. Therefore, with $\sqrt{\mathcal{X}}/20$ samples drawn from P and R, with probability $9/10$ we do not expect to any repetitions in the samples in either the case when $R = P$, or R is a uniform distribution over a different support. Therefore no algorithm can distinguish between the case when $R = P$ or when the support of R is drawn randomly and independently of P with success probability more than $1/10$ given $O(\sqrt{\mathcal{X}})$ samples.

We can now leverage this lower bound to show a lower bound for any t and some pair of distributions R' and P' such that $R'(x)/P'(x) \leq t \;\forall\; x \in \mathcal{X}$. Let U be the uniform distribution over $\mathcal{X}$. For P and R as defined in the previous paragraph, let $P' = (2/t)U + (1 - 2/t)P$ and $R' = (2/t)U + (1 - 2/t)R$. Notice that in this case $R'(x)/P'(x) \in \{1, t-1\} \;\forall\; x \in \mathcal{X} \implies R'(x)/P'(x) \leq t \;\forall\; x \in \mathcal{X}$. We note that if the support of R is chosen randomly and independently of the support of P, then by a Chernoff bound with probability $9/10$ the overlap in their supports is at most $|\mathcal{X}|/3$, which implies that $D(R'\|P') \geq \log(t)/10$ with probability $9/10$. We now observe that if there exists an algorithm to distinguish whether $D(R'\|P') = 0$ or $D(R'\|P') \geq \log(t)/10$ with success probability at least p with $O(\sqrt{\mathcal{X}})$ samples, then it can be used to distinguish whether $R = P$ or R is a uniform distribution over a different support as in the previous setup with success probability at least $(p - 1/10)$ with $O(\sqrt{\mathcal{X}})$ samples. To verify this, observe that it is easy to generate m samples from P' and R' given m samples from P and R. Therefore, by the lower bound in the previous paragraph, no algorithm can distinguish whether $D(R'\|P') = 0$ or $D(R'\|P') \geq \log(t)/10$ with success probability at least $2/3$ with $O(\sqrt{\mathcal{X}})$ samples. □

Proof of Lemma VIII.2. We have

$$\begin{aligned}
D(R\|P) &= \mathop{\mathbb{E}}_{R} \log\left(\frac{R(x)}{P(x)}\right) \\
&= R(C) \mathop{\mathbb{E}}_{R|_C} \log\left(\frac{R(x)}{P(x)}\right) + R(\bar{C}) \mathop{\mathbb{E}}_{R|_{\bar{C}}} \log\left(\frac{R(x)}{P(x)}\right) \\
&= R(C) \mathop{\mathbb{E}}_{R|_C} \log\left(\frac{R|_C(x)R(C)}{P|_C(x)P(C)}\right) + R(\bar{C}) \mathop{\mathbb{E}}_{R|_{\bar{C}}} \log\left(\frac{R|_{\bar{C}}(x)R(\bar{C})}{P|_{\bar{C}}(x)P(\bar{C})}\right) \\
&= R(C) \mathop{\mathbb{E}}_{R|_C} \log\left(\frac{R|_C(x)}{P|_C(x)}\right) + R(\bar{C}) \mathop{\mathbb{E}}_{R|_{\bar{C}}} \log\left(\frac{R|_{\bar{C}}(x)}{P|_{\bar{C}}(x)}\right) \\
&+ R(C) \mathop{\mathbb{E}}_{R|_C} \log\left(\frac{R(C)}{P(C)}\right) + R(\bar{C}) \log\left(\frac{R(\bar{C})}{P(\bar{C})}\right) \\
&= R(C)D(R|_C\|P|_C) + R(\bar{C})D(R|_{\bar{C}}\|P|_{\bar{C}}) + \mathrm{d}(R(C), P(C)).
\end{aligned}$$

□

B. Proofs from Section IX

Proof of Lemma IX.3. Note that for every $i \in [m]$,

$$Q(S_i) = \sum_{x \in S_i} P(x)w(S_i) = P(S_i)\frac{R(S_i)}{P(S_i)} = R(S_i).$$

Under Q every $x \in S_i$, has the same importance weight $w(S_i)$ relative to P, hence the conditional distributions $Q|_{S_i}$ and $P|_{S_i}$ are identical. □

C. Claims and proofs from Section X

Lemma A.14. *Q is α-multi-accurate for $(R, \mathcal{C})$.*

Proof. Using items (1) and (2) of Lemma IX.3, we can write

$$Q(C) = \sum_{i \in [m]} R(S_i)P|_{S_i}(C) = \mathop{\mathbb{E}}_{\mathbf{S} \sim R}[P|_{\mathbf{S}}(C)], \; R(C) = \mathop{\mathbb{E}}_{\mathbf{S} \sim R}[R|_{\mathbf{S}}(C)]$$

Hence

$$\left|Q(C) - R(C)\right| = \left|\mathop{\mathbb{E}}_{\mathbf{S}\sim R}[P|_{\mathbf{S}}(C) - R|_{\mathbf{S}}(C)]\right| \leq \mathop{\mathbb{E}}_{\mathbf{S}\sim R}\left[\left|P|_{\mathbf{S}}(C) - R|_{\mathbf{S}}(C)\right|\right] \leq \alpha.$$

□

Proof of Lemma X.2. By Lemma IX.3 we have

$$Q(S_i \cap C) = R(S_i)P|_{S_i}(C), \quad R(S_i \cap C) = R(S_i)R|_{S_i}(C).$$

Hence

$$\begin{aligned}\left|\frac{Q(S_i \cap C)}{Q(C)} - \frac{R(S_i \cap C)}{R(C)}\right| &= R(S_i)\left|\frac{P|_{S_i}(C)}{Q(C)} - \frac{R|_{S_i}(C)}{R(C)}\right| \\ &\leq \frac{R(S_i)}{R(C)}\left|P|_{S_i}(C) - R|_{S_i}(C)\right| + R(S_i)P|_{S_i}(C)\left|\frac{1}{Q(C)} - \frac{1}{R(C)}\right| \\ &\leq \frac{R(S_i)}{R(C)}\left|P|_{S_i}(C) - R|_{S_i}(C)\right| + Q(S_i \cap C)\frac{\alpha}{Q(C)R(C)}\end{aligned}$$

where we use $|Q(C) - R(C)| \leq \alpha$ by Lemma A.14. We use this to bound the LHS of Equation (37) as

$$\begin{aligned}\left|\sum_{i\in[m]} Q|_C(S_i) - R|_C(S_i)\right| &\leq \sum_{i\in[m]}\left|\frac{Q(S_i \cap C)}{Q(C)} - \frac{R(S_i \cap C)}{R(C)}\right| \\ &\leq \sum_{i\in[m]}\frac{R(S_i)}{R(C)}\left|P|_{S_i}(C) - R|_{S_i}(C)\right| + \sum_{i\in[m]} Q(S_i \cap C)\frac{\alpha}{Q(C)R(C)} \\ &\leq \frac{\alpha}{R(C)} + Q(C)\frac{\alpha}{Q(C)R(C)} = \frac{2\alpha}{R(C)}.\end{aligned}$$

where the last line uses the definition of approximate multi-calibration. □

Proof of Lemma X.3. By Equations (30) and (31)

$$D\left(R|_C \| P|_C\right) - D\left(R|_C \| Q|_C\right) = \mathop{\mathbb{E}}_{\mathbf{S}\sim R|_C}[\log(w(\mathbf{S}))] + \log\left(\frac{P(C)}{Q(C)}\right)$$

By Equation (29)

$$D\left(Q|_C \| P|_C\right) = \mathop{\mathbb{E}}_{\mathbf{x}\sim Q|_C}\left[\log\left(\frac{Q|_C(\mathbf{x})}{P|_C(\mathbf{x})}\right)\right] = \mathop{\mathbb{E}}_{\mathbf{S}\sim Q|_C}[\log(w(\mathbf{S}))] + \log\left(\frac{P(C)}{Q(C)}\right)$$

Subtracting we get

$$\begin{aligned}\left|D\left(R|_C \| P|_C\right) - D\left(R|_C \| Q|_C\right) - D\left(Q|_C \| P|_C\right)\right| &= \left|\mathop{\mathbb{E}}_{\mathbf{S}\sim R|_C}[\log(w(\mathbf{S})] - \mathop{\mathbb{E}}_{\mathbf{S}\sim Q|_C}[\log(w(\mathbf{S})]\right| \\ \leq \left|\sum_{i\in[m]} R|_C(S_i) - Q|_C(S_i)\right| &\max_{i\in[m]}|\log(w(S_i))| \leq 2\alpha\frac{\log(\|w\|_\infty)}{R(C)}\end{aligned}$$

where we use the $(1,\infty)$ version of Holder's inequality, and then Lemma X.2. □

D. Proofs from Section X-B

In this section we prove Theorem X.4.

Proof of Theorem X.4. Let P be the uniform distribution on $\{0,1\}^2$. Let R be the distribution where

$$R(00) = 1/4, R(01) = 1/4, R(10) = 0, R(11) = 1/2.$$

We denote the two coordinates x_0, x_1, and let $\mathcal{C}$ consist of all subcubes of dimension 1. Hence $\mathcal{C} = \{x : x_i = a\}_{i\in\{0,1\},a\in\{0,1\}}$.

The distribution Q^0 for $\alpha = 0$ is the product distribution which matches the marginal distributions on each coordinate: $Q^0(x_0 = 1) = 1/2, Q^0(x_1 = 1) = 3/4$, and the coordinates are independent. The multi-accuracy constraints hold since

$$Q(x_0 = 1) = R(x_0 = 1) = 1/2$$

$$Q(x_1 = 1) = R(x_1 = 1) = 3/4$$

and Q^0 is the maximum entropy distribution satisfying these constraints since the co-ordinates are independent.

Now consider the set $C = \{x_0 = 0\}$. Let $B(p)$ denote the Bernoulli distribution with parameter p. It follows that $R|_C = P|_C = B(1/2)$, whereas $Q^0|_C = B(3/4)$. Hence

$$D\left(R|_C \| P|_C\right) = 0, D\left(R|_C \| Q|_C\right) = D\left(Q|_C \| P|_C\right) = d(3/4, 1/2).$$

Assume that Q satisfies (α, β) multi-group attribution, so that

$$2 \cdot d(3/4, 1/2) = \left| D\left(R|_C \| P|_C\right) - D\left(R|_C \| Q|_C\right) - D\left(Q|_C \| P|_C\right) \right| \leq \frac{\beta}{R(C)}$$

Since $R(C) = 1/2$, this implies $\beta \geq d(3/4, 1/2)$ as desired. □

The work of [**?**] introduced sandwiching bounds for importance weights. The setting is that the learnt importance weights w are meant to approximate the true importance weights w^* of the distribution R relative to P. In analogy with completeness and soundness for proof systems, they ask that for every $C \in \mathcal{C}$, the importance weights satisfy

$$\frac{R(C)}{P(C)} \leq \mathop{\mathbb{E}}_{R|_C}[w(x)] \leq \mathop{\mathbb{E}}_{R|_C}[w^*(x)]. \tag{55}$$

We show that multi-group attribution implies similar sandwiching bounds for $\log(w)$. Formally, if $Q = w \cdot P$ satisfies Definition VIII.4, then

$$\log\left(\frac{R(C)}{P(C)}\right) \leq \mathop{\mathbb{E}}_{R|_C}[\log(w(x))] \leq \mathop{\mathbb{E}}_{R|_C}[\log(w^*(x))]. \tag{56}$$

Equations (55) and (56) are similar in form, yet neither of them implies the other, and indeed there are some subtle differences. We show that the upper bound in Equation (56) only requires multi-accuracy (as opposed to full multi-group attribution). In contrast, [**?**] showed that neither direction of Equation (55) is implied by multiaccuracy alone. The connection to the Pythagorean property makes our proof technically simpler.

Corollary A.15. *If Q satisfies (α, β) multi-group attribution for $(P, R, \mathcal{C})$, then for every $C \in \mathcal{C}$ where $R(C) > \alpha$,*

$$\log\left(\frac{R(C)}{P(C)}\right) - \frac{\beta}{R(C)} - \frac{\alpha}{R(C) - \alpha} \leq \mathop{\mathbb{E}}_{R|_C}[\log(w(\mathbf{x}))] \leq \mathop{\mathbb{E}}_{R|_C}[\log(w^*(\mathbf{x}))] + \frac{\alpha}{R(C)}. \tag{57}$$

Proof of Corollary A.15. We start by relating the central quantity to conditional KL divergence. For $x \in C$,

$$w(x) = \frac{Q(x)}{P(x)} = \frac{Q|_C(x)}{P|_C(x)} \frac{Q(C)}{P(C)}$$

Hence

$$\mathop{\mathbb{E}}_{R|_C}[\log(w(\mathbf{x}))] = \mathop{\mathbb{E}}_{R|_C} \log\left(\frac{Q|_C(\mathbf{x})}{P|_C(x)}\right) + \log\left(\frac{Q(C)}{P(C)}\right) \tag{58}$$

We can upper bound this as

$$\begin{aligned}
\mathop{\mathbb{E}}_{R|_C}\left[\log\left(\frac{Q|_C(\mathbf{x})}{P|_C(x)}\right)\right] &= D\left(R|_C \| P|_C\right) - D\left(R|_C \| Q|_C\right) \\
&\leq D\left(R|_C \| P|_C\right) \\
&= \mathop{\mathbb{E}}_{R|_C}[\log(w^*(x))] + \log\left(\frac{P(C)}{R(C)}\right).
\end{aligned}$$

Hence using this in Equation (58),

$$\begin{aligned}
\mathop{\mathbb{E}}_{R|_C}[\log(w(\mathbf{x}))] &\leq \mathop{\mathbb{E}}_{R|_C}[\log(w^*(x))] + \log\left(\frac{P(C)}{R(C)}\right) + \log\left(\frac{Q(C)}{P(C)}\right) \\
&= \mathop{\mathbb{E}}_{R|_C}[\log(w^*(x))] + \log\left(\frac{Q(C)}{R(C)}\right)
\end{aligned} \tag{59}$$

By multiaccuracy, $Q(C) \leq R(C) + \alpha$. Hence

$$\log\left(\frac{Q(C)}{R(C)}\right) \leq \log\left(1 + \frac{\alpha}{R(C)}\right) \leq \frac{\alpha}{R(C)}. \tag{60}$$

Plugging Equation (60) into (59) gives the upper bound.

To show the lower bound, we start from Equation (58). We lower bound the first term using the Pythagorean property as

$$\begin{aligned}\mathop{\mathbb{E}}_{R|_C} \log\left(\frac{Q|_C(\mathbf{x})}{P|_C(x)}\right) &= D\left(R|_C \| P|_C\right) - D\left(R|_C \| Q|_C\right)\\ &\geq D\left(Q|_C \| P|_C\right) - \frac{\beta}{R(C)} \geq -\frac{\beta}{R(C)} \end{aligned} \tag{61}$$

where the last inequality uses the non-negativity of KL divergence. For the last term, we use $Q(C) \geq R(C) - \alpha$ and the inequality $\log(1-x) \geq -x/(1-x)$ to get

$$\log\left(\frac{Q(C)}{R(C)}\right) \geq \log\left(1 - \frac{\alpha}{R(C)}\right) \geq -\frac{\alpha}{R(C) - \alpha}$$

hence

$$\log\left(\frac{Q(C)}{P(C)}\right) = \log\left(\frac{R(C)}{P(C)}\right) + \log\left(\frac{Q(C)}{R(C)}\right) \geq \log\left(\frac{R(C)}{P(C)}\right) - \frac{\alpha}{R(C) - \alpha} \tag{62}$$

Plugging Equations (61) and (62) into Equation (58) gives the lower bound

$$\mathop{\mathbb{E}}_{R|_C}[\log(w(\mathbf{x}))] \geq \log\left(\frac{R(C)}{P(C)}\right) - \frac{\beta}{R(C)} - \frac{\alpha}{R(C) - \alpha}$$

□

Recall $K^\alpha = K^\alpha(R, \mathcal{C})$ is the set of all α-multi-accurate distributions for $R, \mathcal{C}$. For every $\mathcal{C}$ and $\alpha \geq 0$, K^α is a convex set, since it is given by linear constraints, and it is non-empty since $R \in K^\alpha$. An important class of distributions is the set of Gibbs distributions.

Definition A.16. *The set of all* Gibbs distributions $\mathcal{G} = \mathcal{G}(P, \mathcal{C})$ *is all distributions of the form*

$$Q(x) = P(x) \exp\Big(\sum_{c \in \mathcal{C}} \lambda_c c(x) - \lambda_0\Big) \tag{63}$$

where we use $c(x)$ to denote the indicator function for the set c. Writing $Q = w \cdot P$, we have $\log(w(x)) = \sum_{c \in \mathcal{C}} \lambda_c c(x) - \lambda_0$. The free parameters are $\lambda_{\mathcal{C}} = \{\lambda_c\}_{c \in \mathcal{C}}$, from these, we set the parameter λ_0 so that $\mathbb{E}_P[w(\mathbf{x})] = 1$. We define $\ell_1(Q) = \sum_{c \in \mathcal{C}} |\lambda_c|$ to be the ℓ_1 norm of the free parameters. We now describe three approaches in the literature that lead to essentially the same algorithm, which finds a multi-accurate Gibbs distribution. This algorithm is known to out-perform other density-ratio estimation algorithms in the non-realizable setting [34].

1) [27, 3] **Log-linear KLIEP**: Find the Gibbs distribution $Q \in \mathcal{G}$ that minimizes $D\left(R \| Q\right)$. This goal is to find a good density-ratio estimate.
2) [14, 15] **Divergence estimation using Gibbs distributions:** Find the Gibbs distribution $Q = w \cdot P$ that maximizes the lower bound $\mathbb{E}_R[\log(w(\mathbf{x}))]$ on $D\left(R \| P\right)$. This algorithm is proposed in Section 4A of [15] for the goal of divergence estimation.
3) [28, 29, 32, 35, 30] **MaxEnt:** Learn a model $Q^\alpha \in K^\alpha(R, \mathcal{C})$ for R by finding the distribution Q^α that minimizes $D\left(Q \| P\right)$.

The equivalence of (1) and (2) is well-known (it follows from Equation (31)). A generalization to f-divergences may be found in [24], or Section 7.3.1 of [3]. We have not found the equivalence to (3) explicitly in the literature, but it follows from known convex duality results.

Lemma A.17. *[30] $Q^\alpha \in K^\alpha \cap \mathcal{G}$ is the optimal solution to the following programs*

$$\min_{Q \in K^\alpha} D\left(Q \| P\right), \tag{64}$$

$$\min_{Q \in \mathcal{G}} D\left(R \| Q\right) + \alpha \ell_1(Q). \tag{65}$$

The first program is the one solved by MaxEnt. The second is an ℓ_1-regularized version of the program considered by Log-linear KLIEP and [15]. We derive their exact program by setting $\alpha = 0$.

E. Additional results for mixture of Gaussians

We estimated $D\left(R \| P\right)$, when R, P are mixtures of k Gaussians. In Table III we take $N = 500,000$ samples of two dimensional Gaussians. uLSIF is not in the table since it creates a $N \times N$ kernel matrix and hence does not scale to this number of samples. In Table IV we take 5 dimensional Gaussians. In all these tables we see MC providing the most accurate estimate, especially when k increases. Surprisingly, MC estimates the divergence quite well even when the set of base classifiers is just threshold over basic featers (MC-DT1).

TABLE III: KL estimations for mixture of k Gaussians with $N = 500000$

k	KL	LLK-RF5	MC-RF5	LLK-DT1	MC-DT1	KLIEP
5	9.02	7.66	$\mathbf{7.79}$	2.43	7.54	4.95
10	9.02	6.09	$\mathbf{7.49}$	1.23	7.34	4.88
15	9.02	3.69	$\mathbf{7.23}$	0.40	6.61	5.07

TABLE IV: KL estimations for mixture of k Gaussians with $d = 5$, $N = 500000$

k	KL	LLK-RF5	MC-RF5	LLK-DT1	MC-DT1	KLIEP
5	8.11	$\mathbf{6.76}$	6.30	1.94	5.50	4.16
10	8.11	$\mathbf{6.12}$	6.05	0.80	4.74	3.34
15	8.11	4.35	$\mathbf{5.80}$	0.39	3.59	3.59

a) Contour plots.: Figures 4a, 4b show contours of the first 9 pairs of Gaussians (first 9 sub-populations) in the mixture of $k = 10$ Gaussians. In these images we sampled from a mixture of R and P and colored the points by their importance weight. We can clearly see that MC is consistent in assigning the importance weights, approximately separating the cluster by a diagonal hyper-plane. On the other hand, LL-KLIEP makes mistakes on a number of clusters and does not always succeed in separating the Gaussians well. separating the Gaussians well.

F. Computation time and resources

All our experiments are run on a standard laptop computer with 8 GB RAM. Our Python implementations of MC andLL-KLIEP have not been optimized at all for runtime efficiency. Still, we find the MC algorithm to be reasonably efficient. To provide some ballpark numbers for the runtimes, MC-RF5 on the mixture of Gaussians experiment with $d = 2, k = 15, N = 30000$ takes around 10 seconds to fit the data. The LL-KLIEP implementation is slower, and takes around 100 seconds to fit this data. The MATLAB implementations of KLIEP and uLSIF take around 1 second to fit this data.

G. Choice of hyperparameters

The hyperparameters in the MC algorithm are the choice of classifier family, the width of the branching program, and the advantage below which we do not split a node. The choice of the classifier family is mentioned in each of the experiments. We set the width of the program to be 60, and the advantage to be 0.02 for all the experiments. The hyperparameters for the LL-KLIEP algorithm are similar: the classifier family (as above, mentioned in the experiments), the advantage below which the program terminates (set to be 0.02 as before), and the learning rate (set to be 0.02). The hyperparameters for the KLIEP and uLSIF algorithms (width of the kernel and regularization) are optimized using the automatic hyperparameter tuning and cross-validation routines provided in the code.

One of the motivations of this work is fairness, and to mitigate disparities between sub-populations that should be treated similarly. By now it is well understood that there are tradeoffs between different notions of fairness, and the right definitions to be deployed are context-dependent and depend on societal norms. This is doubly true for a framework like ours, where a family of definitions is suggested parameterized by the collection $\mathcal{C}$. We do not give guidance about what choice of $\mathcal{C}$ is most suitable, it depends on the context. We believe that exploring the landscape of definitions and guarantees is critical even at the risk of abuse (a risk that exists for every work in this area).

TABLE V: Standard deviations across the k subgroups for KL estimations for mixture of k Gaussians with $N = 30000$, averaged over 5 trials. Algorithms with lowest standard deviations are in bold. We note that though KLIEP and ULSIF have low deviations, the divergence that they discover is also low (see Table I).

k	LLK-RF5	MC-RF5	LLK-DT1	MC-DT1	KLIEP	ULSIF
5	$\mathbf{0.42 \pm 0.15}$	$\mathbf{0.41 \pm 0.13}$	1.94 ± 0.65	$\mathbf{0.36 \pm 0.13}$	$\mathbf{0.35 \pm 0.24}$	$\mathbf{0.28 \pm 0.15}$
10	2.27 ± 0.24	$\mathbf{0.57 \pm 0.21}$	1.34 ± 0.24	$\mathbf{0.58 \pm 0.27}$	$\mathbf{0.49 \pm 0.16}$	$\mathbf{0.51 \pm 0.15}$
15	2.54 ± 0.11	$\mathbf{0.60 \pm 0.19}$	0.93 ± 0.18	1.16 ± 0.23	$\mathbf{0.92 \pm 0.30}$	$\mathbf{0.74 \pm 0.25}$

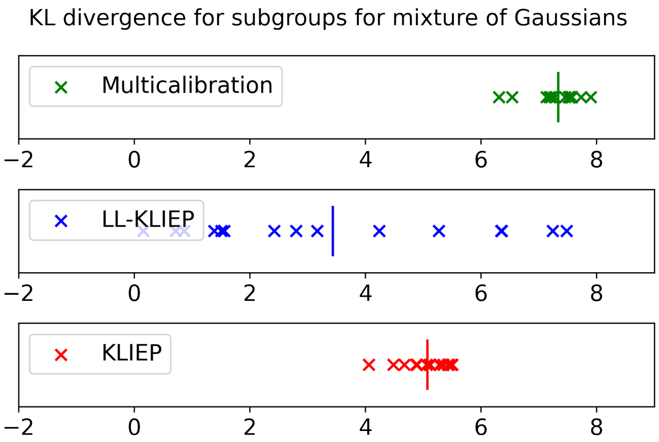
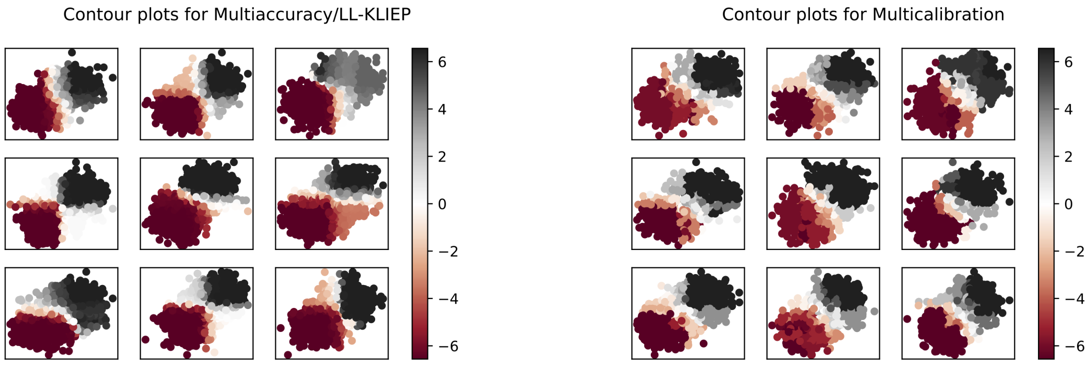


Fig. 3: KL estimations for the 15 subgroups for mixture of 15 Gaussians with $N = 500000$



(a) Color indicates the importance weight assigned by LL-KLIEP.

(b) Color indicates the importance weight assigned by the multi-calibration algorithm.

Fig. 4: The first 9 pairs of Gaussians in the mixture of $k = 10$ Gaussians.