Agnostic Learning, Multicalibration, and Omniprediction

Lecturer: Varun Kanade

These lecture notes are written to provide additional material for two lectures given at the Copenhagen Learning Theory Summer School in June 2026. They are not intended to be complete and it is recommended that the interested reader should read the references given in these notes.

1 Agnostic Learning

We'll begin by introducing a framework called *agnostic learning*. This term was defined by Kearns et al. (1994) though essentially variants of these existed earlier including, e.g., (Haussler, 1990; Vapnik, 1998).

Let $\mathcal{X}$ be some input space within which our data representations lie, e.g., $\mathcal{X} = \mathbb{R}^d$ and let $\mathcal{Y}$ be the output space, e.g., $\mathcal{Y} = \{0, 1\}$ (binary classification), or $\mathcal{Y} = [k]$ (multiclass classification), or $\mathcal{Y} = \mathbb{R}$ (regression). Suppose there is an underlying distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$ and the data that our learning algorithms receive will be an i.i.d. sample from this distribution.

We will also assume that there is an *action* space $\mathcal{A}$ such that $\mathcal{A} \supseteq \mathcal{Y}$. We will consider loss functions,

$$\ell : \mathcal{Y} \times \mathcal{A} \to \mathbb{R}^+.$$

For some function, $h : \mathcal{X} \to \mathcal{A}$ for a loss function ℓ, we can associate an expected loss, or *risk*, which is defined as,

$$R_\ell(h) = \mathop{\mathbb{E}}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}} \left[\ell(\mathbf{y}, h(\mathbf{x}))\right].$$

Before we continue, let us consider a few examples.

- Let $\mathcal{Y} = \{0, 1\}$ and $\mathcal{A} = \{0, 1\}$. The loss $\ell(y, a) = \mathbb{1}(y \neq a)$ is called the *classification* loss or *zero-one* loss.
- Let $\mathcal{Y} = \mathcal{A} = \mathbb{R}$. The loss $\ell(y, a) = (y - a)^2$ is called the squared loss and is often used for regression problems.
- Let $\mathcal{Y} = \mathcal{A} = \mathbb{R}$. The loss $\ell(y, a) = |y - a|$ is called the absolute loss or ℓ_1-loss and is used in regression problems when we expect outliers.

What should the goal be from the point of view of learning? In a way, we'd like to find h such that $R_\ell(h)$ is as small as possible. However, it is hard to decide what value of *small* is good enough. We have not made any assumptions on the distribution $\mathcal{D}$, in particular, we've not made any functional assumptions between y and x for (x, y) in the support of $\mathcal{D}$. It is this lack of assumption that motivates the name *agnostic* learning. What we'd like is that if the *distribution* is structured, then we can recover some of the structure. On the other hand if $\mathbf{x}$ and $\mathbf{y}$ are independent, then there is nothing to be learnt. In agnostic learning, we can characterize this in terms of a reference class of functions, $\mathcal{F}$. For some such class of functions $\mathcal{F}$, we'd like to design algorithms that produce a hypothesis function h that can compete with the *best* function from the class $\mathcal{F}$ in terms of risk.

More formally, we can state our goal as follows: Identify some $h : \mathcal{X} \to \mathcal{A}$, such that,

$$R_\ell(h) \leq \min_{f \in \mathcal{F}} R_\ell(f) + \epsilon,$$

for some suitably small ϵ.[1]

In practice, we don't have direct access to the distribution $\mathcal{D}$. Any learning algorithm only has access to a sample $S \sim \mathcal{D}^n$ for some $n = n(\epsilon, \delta)$ and outputs some estimate $\widehat{h} = \widehat{h}(S)$ and we would like that with probability at least $1 - \delta$,

$$R_\ell(\widehat{h}) \leq \min_{f \in \mathcal{F}} R_\ell(f) + \epsilon,$$

where $n = n(\epsilon, \delta)$ is the sample complexity of the learning algorithm.

We can formalise the definition, but we will try and avoid being too formal in these lectures in order to focus on the key ideas and intuition. We will assume that the readers can fill in the details required to make the definitions and theorems fully rigorous.

Definition 1 (Agnostic Learning)**.** *We say that a class of functions $\mathcal{F}$, where $f \in \mathcal{F}$ are functions from $\mathcal{X} \to \mathcal{A}$, is agnostically learnable for a loss function $\ell : \mathcal{Y} \times \mathcal{A} \to \mathbb{R}^+$ with sample complexity $n(\cdot, \cdot)$, if there exists a learning algorithm, L, such that for all $\epsilon, \delta > 0$, and for every distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$, L when given as input $S \sim \mathcal{D}^{n(\epsilon,\delta)}$ outputs $\widehat{h}$ such that with probability at least $1 - \delta$,*

$$R_\ell(\widehat{h}) \leq \min_{f \in \mathcal{F}} R_\ell(f) + \epsilon.$$

It is worth clarifying a few points before proceeding as we've omitted several important details from the definition above.

- We have allowed the sample complexity $n = n(\epsilon, \delta)$ to depend on the desired accuracy (ϵ) and confidence level (δ); however, we may also want this to depend on some characterization on the input space $\mathcal{X}$ and the function class $\mathcal{F}$. It stands to reason that if the inputs are more complex or the class of functions that we are attempting to compete against is more complex, then we should expect to require more data.
- So far we've only put information-theoretic or statistical requirements on the learning algorithm. However, we may also insist that the learning algorithms are efficient, i.e. they run in polynomial time. In fact, as we shall see next, it is often the case that computation is the bigger bottleneck for agnostic learning than the amount of data required.

We've considered a fairly general framework so far. Let's consider some specific instances of loss functions and function classes to understand the picture better. Suppose we consider the setting where the outcomes are binary, i.e., $\mathcal{Y} = \{-1, 1\}$.[2] Suppose that $\mathcal{A} = [-1, 1]$. Consider the following loss function that we'll call the *negative correlation loss*,

$$\ell(y, a) = \frac{1}{2}(1 - ya).$$

This loss function is closely related to the *zero-one* loss function. For example, if we consider $\mathcal{A} = \{-1, 1\}$, then, $\ell(y, a)$ as defined above is exactly $\mathbb{1}(y \neq a)$.

[1] For cases where the minimum is not achieved, we might consider inf, however, since we are already allowing for some slack ϵ, this does not usually change anything.

[2] We'll sometimes use $\mathcal{Y} = \{-1, 1\}$ and sometimes $\mathcal{Y} = \{0, 1\}$ for binary classification problems. The actual labels don't actually matter, but the maths will be slightly more intuitive depending on what we choose as the label names.

Why is agnostic learning hard?

It turns out that, in general, agnostic learning is often a *hard* problem in a computational sense, particularly when we consider non-convex loss functions such as the *zero-one* loss or even the negative correlation loss. In almost all cases, the hardness is for computational reasons. For example, assuming the class of functions $\mathcal{F}$ is not too complex, say in terms of VC dimension or Rademacher complexity, then finding a function $f \in \mathcal{F}$ that minimizes the empirical risk functional is usually sufficient to guarantee agnostic learning. Unfortunately, the problem of minimizing the empirical risk often turns out to be NP-hard.

Example: Linear classifiers

Suppose that $\mathcal{F}$ is the class of linear classifiers over $\mathbb{R}^d$, i.e.,

$$\mathcal{F} = \{x \mapsto \text{sign}(\langle w, x \rangle) \mid w \in \mathbb{R}^d\}.$$

Provided we can find a linear separator that minimizes the number of mistakes, we'd be able to do agnostic learning as the VC dimension of $\mathcal{F}$ is only $d + 1$. However, the problem of finding a linear separator that separates a given set of labelled examples is NP-complete.

Remark 2. *It is probably worth mentioning that it is not known (or indeed believed to be true) that the task of agnostically learning linear classifiers is NP-hard. This is because we do not require that the output of the learning algorithm is actually a linear classifier. This is sometimes called as* improper *learning. The problem is still conjectured to be computationally hard. On the other hand, if we require the output of the algorithm to be a linear classifier, the so called* proper learning *setting, then it is indeed the case that the problem is* NP*-hard.*

What would we do in practice? Rather than trying to minimize a non-convex loss function such as the *zero-one* loss, we'd consider minimizing a *convex* loss function. Let's still have $\mathcal{Y} = \{-1, 1\}$, but instead of considering the class of linear classifiers, $\mathcal{F}$, let's look at the class of linear functions,

$$\mathcal{H} = \{x \mapsto \langle w, x \rangle \mid w \in \mathbb{R}^d\}.$$

Let $\mathcal{A} = \mathbb{R}$. We can consider the *hinge loss* function defined as,

$$\ell(y, a) = (1 - ya)_+,$$

where $(x)_+ = x$ if $x \geq 0$ and 0 otherwise. Note that the hinge loss implicitly imposes a margin condition. So if $h(x)y \geq 1$ for all (x, y) in the support of $\mathcal{D}$, then the risk would be 0; however $h(x)$ agreeing in sign with y everywhere is not sufficient. By additionally minimizing the $\|w\|_2$, we get the SVM formulation.

Why is this a good idea?

If we can make the hinge loss really small, then we also get good classification error by considering the classifier $\text{sign} \circ h$. This is because,

$$\mathbb{1}(y \neq \text{sign}(a)) \leq (1 - ya)_+.$$

On the other hand, it is possible that there is some linear classifier $f \in \mathcal{F}$ that has classification error, but no linear function $h \in \mathcal{H}$ has low hinge loss. This is likely to be the case if there are a small number of outliers far away from the classification boundary.

What can we do about hard agnostic learning problems?

Restricted Problem Settings

Perhaps the problem we are trying to solve is too general. We are considering an arbitrary distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$. We can look at the marginal distribution μ of $\mathcal{D}$ over $\mathcal{X}$. What if μ is structured, e.g., $\mu = \mathcal{N}(0, \sigma^2 I_d)$? We do have some slightly more efficient algorithms in this setting. Note that we could also make assumptions about the conditional distribution $\mathbf{y}|\mathbf{x} = x$; however, strong assumptions in that direction take us to the *realizable* setting and we move away from *agnostic* learning.

Reductions

Computer scientists like reducing problems to each other. In particular, this is useful to understand relative hardness of different problems. We can try to understand if the problem of agnostic learning can be reduced to any "easier" problem. In particular, we want to understand whether a *boosting* type result is possible, i.e. if there is a *weak agnostic* learning algorithm, then can we do *boosting* to get an agnostic learning algorithm? The answer is yes, as we'll see next. In fact, the notion of a weak agnostic learner and boosting algorithms turn out to have a lot of applications beyond what was originally conceived.

2 Agnostic Boosting

In this section, we'll focus on minimizing the negative correlation loss or equivalently maximizing correlation. In these lectures, we'll also cheat a bit by assuming that we can compute expectations of quantities that we care about exactly. In practice, we can only estimate these expectations using a sample. However, our definitions are robust to the parameters and a standard application of Chernoff-Hoeffding bounds will suffice in all cases (at a small loss in confidence and accuracy parameters). We'll point out how to do this in a few instances, but then assume that the reader is able to fill in these details as required.

For a function $h : \mathcal{X} \to [-1, 1]$ and a distribution $\mathcal{D}$ over $\mathcal{X} \times \{-1, 1\}$, define,

$$\mathrm{corr}(h, \mathcal{D}) = \mathop{\mathbb{E}}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}} \left[h(\mathbf{x})\mathbf{y}\right].$$

Let $\mathcal{C}$ be a class of functions where each $c \in \mathcal{C}$ is a function from $\mathcal{X} \to \{-1, 1\}$. Then note that for every $c \in \mathcal{C}$,

$$\mathrm{corr}(c, \mathcal{D}) = 1 - 2\mathrm{err}(c, \mathcal{D}),$$

where $\mathrm{err}(c, \mathcal{D}) = \mathbb{P}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}} \left[c(\mathbf{x}) \neq \mathbf{y}\right]$. So minimizing the error and maximizing the correlation are equivalent. We can allow functions in $\mathcal{C}$ to take values in the interval $[-1, 1]$, but then we lose the direct relationship with the classification error.

Definition 3 (Weak Agnostic Learning)**.** *For $\alpha \geq \beta > 0$, a learning algorithm,* WeakLearn*, is an (α, β)-weak agnostic learner for $\mathcal{C}$ and distribution μ over $\mathcal{X}$, if for any distribution $\mathcal{D}$ over $\mathcal{X} \times \{-1, 1\}$ with a marginal distribution μ over $\mathcal{X}$, if $\max_{c\in\mathcal{C}} \mathrm{corr}(c, \mathcal{D}) \geq \alpha$, then given $S \sim \mathcal{D}^n$ as input, with probability at least $1 - \delta$,* WeakLearn *outputs $h : \mathcal{X} \to [-1, 1]$, such that $\mathrm{corr}(h, \mathcal{D}) \geq \beta$.*

Let's start by making a few remarks about this definition before proving a boosting theorem.

- The (α, β)-weak learning definition says that if there is some function $c \in \mathcal{C}$ that has sufficiently high correlation, then we'd like the weak learner to identify some hypothesis that has a non-trivial correlation. In order to have most utility, we'd like such algorithms

to exist for all $\alpha > 0$ (with sample complexity and running time increasing with $1/\alpha$) and for β to be at least polynomially related to α, e.g. $\beta = \alpha^2$.

- The failure probability is required to take into account the possibility that the algorithm may receive a completely unrepresentative sample S.
- Of course, we'd like the sample complexity and running time of the algorithm to be small. However, our interest in this lecture is to understand how such an algorithm can be effectively used as a black-box or oracle, rather than designing specific weak learning algorithms.
- The definition restricts the marginal distribution over $\mathcal{X}$ because in the original applications, weak learning algorithms could only be designed in cases where the marginal distribution was either Gaussian in $\mathbb{R}^d$ or uniform over $\{0,1\}^d$. What is interesting is that unlike in the case of, say AdaBoost, changing the weight of the data is not required since we can change the labels to modify the data distribution.

2.1 Boosting Algorithm

We'll define the following function which will help us define a potential to analyse the boosting algorithm.

Define the function $\phi : \mathbb{R} \to \mathbb{R}^+$ as,

$$\phi(z) = \begin{cases} 1 - z & \text{for } z \leq 0 \\ e^{-z} & \text{otherwise.} \end{cases}$$

The function ϕ is differentiable and its derivative is given by,

$$\phi'(z) = \begin{cases} -1 & \text{for } z \leq 0 \\ -e^{-z} & \text{otherwise.} \end{cases}$$

Observe that ϕ' is increasing and 1-Lipschitz and as a consequence ϕ is convex and 1-smooth. This immediately implies the following lemma.

Lemma 4. *For all $z, \delta \in \mathbb{R}$, $|\phi(z+\delta) - \phi(z) - \delta\phi'(z)| \leq \delta^2/2$.*

Algorithm 1 Agnostic Boosting Algorithm

1. Let $H_0 : \mathbb{R} \to \mathbb{R}$ be the constant 0 function.
2. For $t = 1, 2, \ldots$
   (a) For $(x, y) \in \mathcal{X} \times \{-1, 1\}$, let $w_t(x, y) = -\phi'(H_{t-1}(x)y)$.
   (b) Let $\mathcal{D}_t$ be the distribution defined as follows: draw $(x, y) \sim \mathcal{D}$, output (x, y) w.p. $(1 + w_t(x, y))/2$ and $(x, -y)$ otherwise.
   (c) Call WeakLearn with distribution $\mathcal{D}_t$ to obtain g_t. (Note that the marginal μ over $\mathcal{X}$ is unchanged.)
   (d) Define

$$a_t = \mathop{\mathbb{E}}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}} \left[g_t(\mathbf{x})\mathbf{y}w_t(\mathbf{x},\mathbf{y}) \right]; \quad b_t = \mathop{\mathbb{E}}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}} \left[-\text{sign}(H_{t-1}(\mathbf{x}))\mathbf{y}w_t(\mathbf{x},\mathbf{y}) \right].$$

   (e) If $a_t \geq b_t$, let $h_t = g_t$ and $\gamma_t = a_t$; else $h_t = -\text{sign} \circ H_{t-1}$ and $\gamma_t = b_t$.
   (f) If $\gamma_t \leq \min\{\epsilon, \beta\}$, then exit loop. Else, $H_t = H_{t-1} + \gamma_t h_t$.
3. Output $\text{sign} \circ H_t$.

Theorem 5 ((Kalai and Kanade, 2009)). *The agnostic boosting algorithm (Algorithm 1) after $T = O(1/\min\{\epsilon^2, \beta^2\})$, outputs a hypothesis $h : \mathcal{X} \to \{-1, 1\}$, such that with high probability,*

$$\operatorname{corr}(h, \mathcal{D}) \geq \sup_{c \in \mathcal{C}} \operatorname{corr}(c, \mathcal{D}) - \alpha - \epsilon.$$

In order to prove the theorem, it will be helpful to establish a couple of lemmas.

Lemma 6. *For any $h : \mathcal{X} \to [-1, 1]$, for the distribution $\mathcal{D}_t$ defined at iteration t,*

$$\operatorname{corr}(h, \mathcal{D}_t) = \mathop{\mathbb{E}}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}} \left[h(\mathbf{x})\mathbf{y}w_t(\mathbf{x}, \mathbf{y})\right] = \mathop{\mathbb{E}}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}} \left[-h(\mathbf{x})\mathbf{y}\phi'(H_{t-1}(\mathbf{x})\mathbf{y})\right].$$

The proof of the above lemma is a simple calculation.

Lemma 7. *Let $H : \mathcal{X} \to \mathbb{R}$, then for any $c \in \mathcal{C}$, if $w(x, y) = -\phi'(H(x)y)$,*

$$\mathop{\mathbb{E}}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}} \left[c(\mathbf{x})\mathbf{y}w(\mathbf{x}, \mathbf{y})\right] - \mathop{\mathbb{E}}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}} \left[\operatorname{sign}(H(\mathbf{x}))\mathbf{y}w(\mathbf{x}, \mathbf{y})\right] \geq \operatorname{corr}(c, \mathcal{D}) - \operatorname{corr}(\operatorname{sign} \circ H, \mathcal{D}).$$

Proof. We will observe that,

$$\mathop{\mathbb{E}}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}} \left[(c(\mathbf{x}) - \operatorname{sign}(H(\mathbf{x})))\mathbf{y}(w(\mathbf{x}, \mathbf{y}) - 1)\right] \geq 0.$$

Note that $0 \leq w(x, y) \leq 1$ for all (x, y) and $w(x, y) = 1$ when $\operatorname{sign}(H(x)) = -y$. It follows that if $w(x, y) - 1 < 0$, then $\operatorname{sign}(H(x)) = y$, so $c(x)y \leq \operatorname{sign}(H(x))y$. So the conclusion follows. The proof of the lemma then follows from expanding out the expression in the expectation. □

We can now complete the proof using the potential $\Phi(H) = \mathop{\mathbb{E}}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}} \left[\phi(H(\mathbf{x})\mathbf{y})\right]$.

Proof of Theorem 5. Suppose at time step $t + 1$, it is not the case that

$$\operatorname{corr}(\operatorname{sign} \circ H_t, \mathcal{D}) \geq \mathsf{opt} - \alpha - \epsilon,$$

where $\mathsf{opt} = \sup_{c\in\mathcal{C}} \operatorname{corr}(c, \mathcal{D})$. With slight abuse of notation, we'll assume that $c^* \in \mathcal{C}$ satisfies $\operatorname{corr}(c^*, \mathcal{D}) = \mathsf{opt}$. Then using Lemma 7 we know that $\operatorname{corr}(c^*, \mathcal{D}_{t+1}) - \operatorname{corr}(\operatorname{sign} \circ H_t, \mathcal{D}_{t+1}) > \alpha + \epsilon$. Now either $\operatorname{corr}(c^*, \mathcal{D}_{t+1}) \geq \alpha$ and so WeakLearn returns a g_{t+1} with $a_{t+1} \geq \beta$, or $\operatorname{corr}(-\operatorname{sign} \circ H_t, \mathcal{D}_{t+1}) \geq \epsilon$. Thus, $\gamma_{t+1} \geq \min\{\beta, \epsilon\}$ and the algorithm moves to the next iteration.

Then using Lemma 4, after taking expectations, we have $\Phi(H_t) - \Phi(H_{t+1}) \geq \min\{\beta, \epsilon\}^2/2$. However, $\Phi(H_0) = 1$, so this can only happen for at most $2/\min\{\beta, \epsilon\}^2$ iterations. □

3 Multigroup Fairness Notions

We're going to change tack a bit and look at some concepts that originated in the algorithmic theory of fairness literature. We will soon make connections to agnostic learning and in particular see how the fundamental idea of a weak agnostic learner along with potential-based boosting is a key approach to achieve the solution concepts that satisfy the desired fairness criteria. Suppose we have some underlying population $\mathcal{X}$ and there is some (eventually) observable outcome $y \in \{0, 1\}$ with every $x \in \mathcal{X}$. Let's write $\mathcal{Y} = \{0, 1\}$. For example, this could indicate whether an individual is likely to succeed in some standardized test, or repay a loan, etc. We assume that there is some underlying ground-truth function $p^* : \mathcal{X} \to [0, 1]$ such that for every $x \in \mathcal{X}$, $\mathbb{E}\left[\mathbf{y}|\mathbf{x}\right] = p^*(\mathbf{x})$. This ground-truth p^* may be arbitrarily complex and we don't necessarily expect to learn p^* well from a finite sample. Nevertheless, we might apply a machine learning

method to design a predictor $p : \mathcal{X} \to [0,1]$ that makes a prediction for each individual that their observed outcome will be 1.

As we've already discussed, it may be unrealistic to expect guarantees of the form $p \approx p^*$. What might we reasonably expect p to satisfy? Let $\mathcal{D}$ be the distribution over $\mathcal{X} \times \{0,1\}$ such that $\mathbb{E}\left[\mathbf{y}|\mathbf{x}\right] = p^*(\mathbf{x})$ – that is our underlying data generation process. One minimal guarantee that we might desire is *accuracy in expectation*, i.e., the expected accuracy error is small, or $\mathbb{E}\left[\mathbf{y}\right] \approx \mathbb{E}\left[p(\mathbf{x})\right]$.

Definition 8. *The expected accuracy error of a predictor $p : \mathcal{X} \to [0,1]$ with respect to a distribution $\mathcal{D}$ over $\mathcal{X} \times \{0,1\}$ is defined as*

$$\mathsf{EAE}(p, \mathcal{D}) = |\mathop{\mathbb{E}}_{\mathcal{D}}\left[\mathbf{y}\right] - \mathop{\mathbb{E}}_{\mathcal{D}}\left[p(\mathbf{x})\right]|.$$

A slightly stronger notion that is well-studied in the statistics literature is that of *calibration*. Informally, we would say that a predictor p is (approximately) calibrated if for all v in the range of p, $\mathbb{E}\left[\mathbf{y}|p(\mathbf{x}) = v\right] \approx v$. We can define the expected calibration error as follows.

Definition 9. *The expected calibration error of a predictor $p : \mathcal{X} \to [0,1]$ with respect to a distribution $\mathcal{D}$ over $\mathcal{X} \times \{0,1\}$ is defined as*

$$\mathsf{ECE}(p, \mathcal{D}) = \mathbb{E}\left[\left|\mathbb{E}\left[\mathbf{y} - p(\mathbf{x})|p(\mathbf{x})\right]\right|\right].$$

It should be clear that requiring calibration is a *stronger* guarantee than simply requiring accuracy in expectation. As an exercise the reader is invited to show that $\mathsf{EAE}(p, \mathcal{D}) \leq \mathsf{ECE}(p, \mathcal{D})$.

It is worth observing at this stage that calibration only *decreases* squared error. There is a minor caveat that in order to calibrate using a finite sample, then we will have to discretize the predictor which causes some increase in the squared error.

Recall that p^* is the underlying ground-truth function. Let $p : \mathcal{X} \to [0,1]$ be a predictor. Define a predictor $p^\delta : \mathcal{X} \to [0,1]$, where $p^\delta(x) = i\delta$ if $p(x) \in [i\delta, (i+1)\delta)$.

Exercise: Show that

$$\mathbb{E}\left[(p^* - p^\delta)^2\right] \leq \mathbb{E}\left[(p^* - p)^2\right] + 2\delta.$$

Let us now define $\bar{p} : \mathcal{X} \to [0,1]$, where $\bar{p}(x) = v_i$ for $x \in (p^\delta)^{-1}(i\delta)$ and $v_i = \mathbb{E}\left[\mathbf{y}|p^\delta(\mathbf{x}) = i\delta\right]$.

Claim 10.

$$\mathop{\mathbb{E}}_{\mathcal{D}}\left[(p^* - p^\delta)^2\right] - \mathop{\mathbb{E}}_{\mathcal{D}}\left[(p^* - \bar{p})^2\right] \geq \mathsf{ECE}(p^\delta, \mathcal{D})^2.$$

Proof. First, we'll make the following observation,

$$\begin{aligned}
\mathbb{E}\left[\left|\bar{p}(\mathbf{x}) - p^\delta(\mathbf{x})\right|\right] &= \mathbb{E}\left[\mathbb{E}\left[\left|\bar{p}(\mathbf{x}) - p^\delta(\mathbf{x})\right| \;\middle|\; p^\delta(\mathbf{x}) = i\delta\right]\right] \\
&= \mathbb{E}\left[\left|\mathbb{E}\left[\mathbf{y} - p^\delta(\mathbf{x}) \mid p^\delta(\mathbf{x}) = i\delta\right]\right|\right] = \mathsf{ECE}(p^\delta, \mathcal{D}) \qquad (1)
\end{aligned}$$

Second, we have the following,

$$\begin{aligned}
\mathbb{E}\left[(p^*(\mathbf{x}) - p^\delta(\mathbf{x}))^2\right] &= \mathbb{E}\left[(p^*(\mathbf{x}) - \bar{p}(\mathbf{x}) + \bar{p}(\mathbf{x}) - p^\delta(\mathbf{x}))^2\right] \\
&= \mathbb{E}\left[(p^*(\mathbf{x}) - \bar{p}(\mathbf{x}))^2\right] + \mathbb{E}\left[(\bar{p}(\mathbf{x}) - p^\delta(\mathbf{x}))^2\right] + 2\mathbb{E}\left[(p^*(\mathbf{x}) - \bar{p}(\mathbf{x}))(\bar{p}(\mathbf{x}) - p^\delta(\mathbf{x}))\right] \\
& \qquad (2)
\end{aligned}$$

We observe that

$$\mathbb{E}\left[(p^*(\mathbf{x}) - \bar{p}(\mathbf{x}))(\bar{p}(\mathbf{x}) - p^\delta(\mathbf{x}))\right] = \mathbb{E}\left[\mathbb{E}\left[(p^*(\mathbf{x}) - \bar{p}(\mathbf{x}))(\bar{p}(\mathbf{x}) - p^\delta(\mathbf{x})) \mid p^\delta(\mathbf{x}) = i\delta\right]\right] = 0.$$

Plugging this back in Eq. (2), doing some rearranging and using Eq. (1),

$$\begin{aligned}\mathop{\mathbb{E}}_{\mathcal{D}}\left[(p^*(\mathbf{x}) - p^\delta(\mathbf{x}))^2\right] - \mathop{\mathbb{E}}_{\mathcal{D}}\left[(p^*(\mathbf{x}) - \bar{p}(\mathbf{x}))^2\right] &= \mathbb{E}\left[(\bar{p}(\mathbf{x}) - p^\delta(\mathbf{x}))^2\right] \\ &\geq \mathbb{E}\left[\left|\bar{p}(\mathbf{x}) - p^\delta(\mathbf{x})\right|\right]^2 = \mathsf{ECE}(p^\delta, \mathcal{D})^2.\end{aligned}$$

□

Fairness Criteria

So far we've not talked about fairness or *groups*. Suppose that our underlying population $\mathcal{X}$ has subgroups and we want to be sure that our predictors are not biased, in favour of, or against, any of these groups. To set up some notation, let us say $\mathcal{G} = \{g\}$ is a family of subsets of $\mathcal{X}$, so each $g \subseteq \mathcal{X}$. By slight abuse of notation we'll also denote by g, the indicator function of group membership, i.e., $g : \mathcal{X} \to \{0, 1\}$, $g(x) = \mathbb{1}(x \in g)$. Note that the number of groups could be very large and also potentially overlapping. We will inevitably have difficulties dealing with very small groups, especially when working with a small finite sample, and we'll see how the fairness notions are degrading with respect to these.

Multiaccuracy

Definition 11 (Multiaccuracy, (Hébert-Johnson et al., 2018))**.** *We say that a predictor $p : \mathcal{X} \to [0, 1]$ is τ-multiaccurate for a distribution $\mathcal{D}$ over $\mathcal{X} \times \{0, 1\}$ with respect to a family of groups $\mathcal{G}$ if*

$$\max_{g \in \mathcal{G}} \left|\mathbb{E}\left[g(\mathbf{x})(\mathbf{y} - p(\mathbf{x}))\right]\right| \leq \tau.$$

Let's see how this is a fairness notion. Suppose we fix some group g, then we can look at the *expected* accuracy of the predictor p condition on the group.

We have

$$\mathbb{E}\left[g(\mathbf{x})(\mathbf{y} - p(\mathbf{x}))\right] = \mathbb{E}\left[\mathbf{y} - p(\mathbf{x}) \mid \mathbf{x} \in g\right] \cdot \mathbb{P}\left[\mathbf{x} \in g\right],$$

so then we get for a τ-multiaccurate p,

$$\left|\mathbb{E}\left[\mathbf{y} - p(\mathbf{x}) \mid \mathbf{x} \in g\right]\right| \leq \frac{\tau}{\mathbb{P}\left[\mathbf{x} \in g\right]}.$$

Above, we've assumed obviously that $\mathbb{P}\left[\mathbf{x} \in g\right] \neq 0$. What this says is that the predictor p satisfies accuracy in expectation even conditioned on groups. This guarantee is meaningless for very small groups, but holds for all large groups which may even be overlapping.

We can add the condition that p is calibrated to get the notion of a calibrated multiaccurate predictor.

Definition 12 (Calibrated multiaccuracy)**.** *We say that a predictor $p : \mathcal{X} \to [0, 1]$ is τ-multiaccurate and τ-calibrated for a distribution $\mathcal{D}$ with respect to a family of groups $\mathcal{G}$, if p is τ-multiaccurate and $\mathsf{ECE}(p, \mathcal{D}) \leq \tau$.*

Multicalibration

Finally, we'll define an even stronger notion called multicalibration.

Definition 13 (Multicalibration, (Hébert-Johnson et al., 2018)). *We say that a predictor $p : \mathcal{X} \to [0,1]$ is τ-multicalibrated for a distribution $\mathcal{D}$ with respect to a family of groups $\mathcal{G}$, if*

$$\max_{g \in \mathcal{G}} \mathbb{E}\left[\left|\mathbb{E}\left[g(\mathbf{x})(\mathbf{y} - p(\mathbf{x})) \mid p(\mathbf{x})\right]\right|\right] \leq \tau.$$

We've said that this is a stronger notion. It can be easily checked that if $\mathcal{X} \in \mathcal{G}$, i.e., the entire population is considered as one of the subgroups, then p being multicalibrated implies global calibration of p. On the other hand, we can easily observe that,

$$\left|\mathbb{E}\left[g(\mathbf{x})(\mathbf{y} - p(\mathbf{x}))\right]\right| \leq \mathbb{E}\left[\left|\mathbb{E}\left[g(\mathbf{x})(\mathbf{y} - p(\mathbf{x})) \mid p(\mathbf{x})\right]\right|\right],$$

and so τ-multicalibration implies τ-multiaccuracy.

While the original motivation for defining multiaccuracy and multicalibration comes from the algorithmic theory of fairness literature,[3] we will see that these notions have a deep connection to ideas in learning theory.

We've already alluded to the fact that the groups $g \in \mathcal{G}$ can be thought of as functions $g : \mathcal{X} \to \{0,1\}$. Let's see what it would mean to *test* whether a predictor p is multiaccurate. In order to identify a violation, we would have to determine whether there is some $g \in \mathcal{G}$, such that,

$$\left|\mathbb{E}\left[g(\mathbf{x})(\mathbf{y} - p(\mathbf{x}))\right]\right| > \tau.$$

Essentially this is a question of whether the function $p^*(x) - p(x)$ is correlated with some function $g \in \mathcal{G}$. Define a distribution $\mathcal{D}'$ over $\mathcal{X} \times \{-1, 1\}$, such that the marginal distribution of $\mathcal{D}'$ over $\mathcal{X}$ is the same as that of $\mathcal{D}$, and

$$\mathop{\mathbb{E}}_{(\mathbf{x},\mathbf{z}) \sim \mathcal{D}'}\left[\mathbf{z}|\mathbf{x}\right] = (p^*(\mathbf{x}) - p(\mathbf{x})).$$

This means that if there is a (τ, β)-weak agnostic learning algorithm for the class $\mathcal{G}$, then we can test for multiaccuracy violations.

We will show how we can construct predictors with these fairness guarantees given access to a weak agnostic learner for $\mathcal{G}$.

3.1 Multiaccurate predictors from Weak Agnostic Learning

We will show first how to construct a multiaccurate predictor starting from some predictor $p : \mathcal{X} \to [0,1]$ using access to an (α, β)-weak agnostic learning algorithm for the class $\mathcal{C}$ (we will now refer to the set of groups as $\mathcal{C}$ rather than $\mathcal{G}$ to be consistent with the learning theory notation). We will also assume that $c \in \mathcal{C}$ takes values in $\{-1, 1\}$; assuming that the constant 1 function is included, we can consider the transformation $2f - 1$ of a function $f : \mathcal{X} \to \{0,1\}$, which degrades the multiaccuracy parameter by a factor of 3. Let us denote by τ the desired multiaccuracy parameter; we will require that $\alpha \leq \tau$, i.e. the weak learning algorithm should be capable of detecting correlations as small as τ at least.

As a technical tool, we'll use the clip function. For an interval $[a, b]$ and $h : \mathcal{X} \to \mathbb{R}$, define $\mathrm{clip}_{[a,b]}(h) : \mathcal{X} \to [a, b]$ to be the function that maps $h(x)$ to the closest value in the interval $[a, b]$.

[3] Actually the notion of multiaccuracy implicitly was defined much earlier in computational complexity theory as a form of indistinguishability.

We will analyse Algorithm 2, again, using a potential function argument. As a reminder, we are ignoring statistical issues arising from working with a finite sample rather population level quantities. However, as we've said before these can be handled using Chernoff-Hoeffding bounds with only minor losses in parameters. This slight cheating will make our analysis considerably cleaner.

Algorithm 2 Algorithm for MA

1. Let $p_0 = p$ be the input predictor. Let τ be the multiaccuracy parameter and let WeakLearn be the (α, β)-weak agnostic learner with $\alpha \leq \tau$.
2. For $t = 1, 2, \ldots$
   (a) Define the distribution $\mathcal{D}_t$ as follows. Draw $(\mathbf{x}, \mathbf{y}) \sim \mathcal{D}$, and output $(\mathbf{x}, \mathbf{z})$ where $\mathbf{z}$ takes values in $\{-1, 1\}$, such that $\mathbb{E}\left[\mathbf{z}|\mathbf{x}\right] = p^*(\mathbf{x}) - p_{t-1}(\mathbf{x})$.
   (b) Let h_t be the hypothesis returned by WeakLearn when given access to a sample from $\mathcal{D}_t$.
   (c) If $\mathop{\mathbb{E}}_{(\mathbf{x},\mathbf{z})\sim\mathcal{D}_t}\left[h_t(\mathbf{x})\mathbf{z}\right] \geq \beta$, $p_t = \text{clip}_{[0,1]}(p_{t-1} + \beta h_t)$; else break and return p_t

Let's begin by a very simple observation that clipping a function to the interval $[0, 1]$ only reduces the squared error with respect to any function $p^* : \mathcal{X} \to [0, 1]$.

Claim 14. *Suppose $p^* : \mathcal{X} \to [0, 1]$ and $h : \mathcal{X} \to \mathbb{R}$, then,*

$$\mathbb{E}\left[(p^*(\mathbf{x}) - \text{clip}_{[0,1]}(h)(\mathbf{x}))^2\right] \leq \mathbb{E}\left[(p^*(\mathbf{x}) - h(\mathbf{x}))^2\right].$$

Next, we will show that every iteration of Algorithm 2 also decreases the squared error with respect to p^*.

Lemma 15. *Suppose at iteration t, $\mathop{\mathbb{E}}_{(\mathbf{x},\mathbf{z})\sim\mathcal{D}_t}\left[\mathbf{z}h_t(\mathbf{x})\right] \geq \beta$, then,*

$$\mathbb{E}\left[(p^*(\mathbf{x}) - p_t(\mathbf{x}))^2\right] \leq \mathbb{E}\left[(p^*(\mathbf{x}) - p_{t-1}(\mathbf{x}))^2\right] - \beta^2.$$

Proof. Since $\|h_t\|_\infty \leq 1$, we have

$$\mathbb{E}\left[(p^*(\mathbf{x}) - p_{t-1}(\mathbf{x}) - \beta h_t(\mathbf{x}))^2\right] \leq \mathbb{E}\left[(p^*(\mathbf{x}) - p_{t-1}(\mathbf{x}))^2\right] - 2\beta\mathbb{E}\left[(p^*(\mathbf{x}) - p_{t-1}(\mathbf{x}))h_t(\mathbf{x})\right] + \beta^2.$$

Since the algorithm did not terminate at iteration t, we also have,

$$\mathbb{E}\left[(p^*(\mathbf{x}) - p_{t-1}(\mathbf{x}) - \beta h_t(\mathbf{x}))^2\right] \leq \mathbb{E}\left[(p^*(\mathbf{x}) - p_{t-1}(\mathbf{x}))^2\right] - 2\beta^2 + \beta^2.$$

Finally, appealing to Claim 14 we get the desired result. □

If we consider the potential function $\mathbb{E}\left[(p^*(\mathbf{x}) - p_t)^2\right]$, we know that $\mathbb{E}\left[(p^*(\mathbf{x}) - p_0(\mathbf{x}))^2\right] \leq 1$ and that the potential-based algorithm can only continue for $O(1/\beta^2)$ iterations. However, if p_t is not τ-multiaccurate, then the weak learning algorithm is guaranteed to succeed, so we will get a τ-multiaccurate p_t in $O(1/\beta^2)$ iterations.

An algorithm for getting a predictor that is τ-calibrated and τ-multiaccurate is not much more complicated.

We won't analyse this algorithm in detail, but the general idea is the same. We've already seen that recalibrating only reduces the squared error as does the multiaccuracy algorithm. Thus, the only step that potentially increases the potential is the discretization, but the parameter δ is chosen so that the increase in potential is always offset by a reduction due to multiaccuracy and (re)-calibration. In fact, getting a calibrated multiaccurate predictor is not more expensive than getting a multiaccurate predictor.

Algorithm 3 Algorithm for Calibrated Multiaccuracy

1. Receive as input $q_0 : \mathcal{X} \to [0,1]$, a parameter τ and an algorithm for achieving multiaccuracy, MA.
2. Let $\delta = \tau^2/8$.
3. For $t = 1, 2, \ldots,$
   (a) $p_t \leftarrow \mathsf{MA}(q_{t-1}, \tau - \delta)$.
   (b) If $\mathsf{ECE}(p_t^\delta, \mathcal{D}) > 3\tau/4$, then recalibrate p_t^δ to get q_t.
   (c) Else, let $q_t = p_t^\delta$, break and return q_t.

3.2 From multigroup fairness to learning

In the previous section, we've seen how access to a weak agnostic learner is sufficient to derive predictors that satisfy different levels of multigroup fairness guarantees. In fact, we've seen that *weak* agnostic learning is a remarkably powerful primitive and suffices to derive results in learning and fairness.

We'll try to understand whether multigroup fairness notions themselves result in any learning guarantees. We've generally thought of the groups $\mathcal{G}$ as sub-populations, but we could equally consider the multigroup fairness guarantees as providing some promise with respect to a class of functions over the domain $\mathcal{X}$. In order to be consistent with the notation in learning theory, we'll refer to multiaccuracy/calibration with respect to a class of functions $\mathcal{C}$ rather than $\mathcal{G}$. In fact, let's think of a function $c \in \mathcal{C}$ as being a function from $\mathcal{X} \to \{-1, 1\}$.

Multiaccuracy alone is weak!

Let's first consider a predictor $p : \mathcal{X} \to [0,1]$ that is multiaccurate with respect to some class of functions $\mathcal{C}$. In particular, this means that for every $c \in \mathcal{C}$,

$$\left| \mathop{\mathbb{E}}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}} \left[c(\mathbf{x})(\mathbf{y} - p(\mathbf{x})) \right] \right| \leq \tau.$$

If we wanted to consider the *correlation* (or negative correlation loss), we'll look at

$$\max_{c\in\mathcal{C}} \mathbb{E}\left[(2\mathbf{y} - 1)c(\mathbf{x}) \right],$$

and ask if the predictor p can be processed to achieve a correlation that is almost as high. The most natural processing is to consider $\text{sign}(p(x) - 1/2)$. So, is it the case that,

$$\mathbb{E}\left[(2\mathbf{y} - 1)\text{sign}\left(p(\mathbf{x}) - \frac{1}{2} \right) \right] \geq \max_{c\in\mathcal{C}} \mathbb{E}\left[(2\mathbf{y} - 1)c(\mathbf{x}) \right] - O(\tau)?$$

Somewhat surprisingly, the answer is no!

We'll see a simple counterexample and then try and understand what the main problem is. Suppose $\mathcal{X} = \{(x_1, x_2, x_3) \mid x_i \in \{0,1\}\}$ be the Boolean hypercube. Let $\mathcal{C}$ be the set of all functions from $\mathcal{X}$ to $\{-1, 1\}$ that depend only on x_1 and x_2. Suppose that $p^*(x) = \text{majority}(x_1, x_2, x_3)$ and $p(x) = \text{majority}(x_1, x_2, 1 - x_3)$. Let $\mathcal{D}$ be a distribution such that the marginal distribution over $\mathcal{X}$ is uniform.

Let's establish some claims.

$$\begin{aligned} p^*(x) &= \text{majority}(x_1, x_2, x_3) = x_1x_2 + x_2x_3 + x_3x_1 - 2x_1x_2x_3 \\ p(x) &= \text{majority}(x_1, x_2, 1 - x_3) = x_1x_2 + x_2(1 - x_3) + (1 - x_3)x_1 - 2x_1x_2(1 - x_3) \\ p^*(x) - p(x) &= (2x_3 - 1)(x_1 + x_2 - 2x_1x_2) \end{aligned}$$

Then it is clear that for every $c \in \mathcal{C}$,

$$\mathbb{E}\left[c(\mathbf{x})(p^*(\mathbf{x}) - p(\mathbf{x}))\right] = 0,$$

as $\mathbb{E}\left[2x_3 - 1\right] = 0$.

This establishes that p is 0-multiaccurate for $\mathcal{C}$. It is then easy to check that

$$\mathbb{P}\left[p^*(\mathbf{x}) \neq p(\mathbf{x})\right] = \frac{1}{2}.$$

On the other hand, if $c^*(x) = x_1 x_2$, i.e. the conjunction of x_1 and x_2, then

$$\mathbb{P}\left[p^*(\mathbf{x}) \neq c^*(\mathbf{x})\right] = \frac{1}{4}.$$

Thus, we can see that while there is a $c^* \in \mathcal{C}$ which has positive correlation with p^*, p which is multiaccurate does not. In fact, p is *anti-calibrated*, i.e., knowing the value $p(x)$ reveals nothing about $p^*(x)$. We'll next show that adding *global calibration* suffices to give strong learning-theoretic guarantees.

Calibrated Multiaccuracy yields Agnostic Learning

We'll see now that a predictor p that is both multiaccurate and calibrated does yield agnostic learning. In fact, the contributions of multiaccuracy and calibration can be cleanly separated out in a transparent fashion to understand how together they *must* yield agnostic learning.

We'll set these out in the following two lemmas.

Lemma 16 ((Casacuberta et al., 2025)). *Suppose p is τ-calibrated for $\mathcal{D}$, then,*

$$\mathbb{E}\left[(2\mathbf{y} - 1)\text{sign}\left(p(\mathbf{x}) - \frac{1}{2}\right)\right] \geq 2\mathbb{E}\left[\left|p(\mathbf{x}) - \frac{1}{2}\right|\right] - 2\tau.$$

Proof.

$$\begin{aligned}
\mathbb{E}\left[(2\mathbf{y} - 1)\text{sign}\left(p(\mathbf{x}) - \frac{1}{2}\right)\right] &= \mathbb{E}\left[(2p(\mathbf{x}) - 1)\text{sign}\left(p(\mathbf{x}) - \frac{1}{2}\right)\right] + 2\mathbb{E}\left[(\mathbf{y} - p(\mathbf{x}))\text{sign}\left(p(\mathbf{x}) - \frac{1}{2}\right)\right] \\
&= 2\mathbb{E}\left[\left(p(\mathbf{x}) - \frac{1}{2}\right)\text{sign}\left(p(\mathbf{x}) - \frac{1}{2}\right)\right] \\
&\quad + 2\mathbb{E}\left[\text{sign}\left(p(\mathbf{x}) - \frac{1}{2}\right)\mathbb{E}\left[\mathbf{y} - p(\mathbf{x}) \mid p(\mathbf{x})\right]\right] \\
&\geq 2\mathbb{E}\left[\left|p(\mathbf{x}) - \frac{1}{2}\right|\right] - 2\mathbb{E}\left[\left|\mathbb{E}\left[\mathbf{y} - p(\mathbf{x}) \mid p(\mathbf{x})\right]\right|\right] \\
&\geq 2\mathbb{E}\left[\left|p(\mathbf{x}) - \frac{1}{2}\right|\right] - 2\tau.
\end{aligned}$$

□

What is interesting about the lemma is that it clearly identifies the role of calibration. For example, if the labels are balanced then $p(x) = 1/2$ everywhere will be a calibrated predictor, but might not have any correlation with the observed labels. However, if p is also multiaccurate, then it cannot be very close to the constant $1/2$ function, unless there is no $c \in \mathcal{C}$ that has a strong correlation with the labels. That is what the next lemma establishes.

Lemma 17 ((Casacuberta et al., 2025)). *Suppose p is τ-multiaccurate for $\mathcal{C}$ with respect to $\mathcal{D}$, then for every $c \in \mathcal{C}$,*

$$\mathbb{E}\left[c(\mathbf{x})(2\mathbf{y}-1)\right] \leq \mathbb{E}\left[\left|p(\mathbf{x}) - \frac{1}{2}\right|\right] + 2\tau.$$

Proof.

$$\begin{aligned}\mathbb{E}\left[c(\mathbf{x})(2\mathbf{y}-1)\right] &= \mathbb{E}\left[c(\mathbf{x})(2p(\mathbf{x})-1)\right] + 2\mathbb{E}\left[c(\mathbf{x})(\mathbf{y}-p(\mathbf{x}))\right] \\ &\leq \mathbb{E}\left[|c(\mathbf{x})||2p(\mathbf{x})-1|\right] + 2\tau \\ &= 2\mathbb{E}\left[\left|p(\mathbf{x}) - \frac{1}{2}\right|\right] + 2\tau.\end{aligned}$$

□

Theorem 18 ((Casacuberta et al., 2025)). *If p is τ-multiaccurate and τ-calibrated with respect to $\mathcal{D}$ for a class $\mathcal{C}$, then,*

$$\mathbb{E}\left[(2\mathbf{y}-1)\text{sign}\left(p(\mathbf{x}) - \frac{1}{2}\right)\right] \geq \max_{c\in\mathcal{C}} \mathbb{E}\left[c(\mathbf{x})(2\mathbf{y}-1)\right] - 4\tau.$$

4 Omniprediction

We will now define omnipredictors. Suppose $\mathcal{C}$ is a class of functions from $\mathcal{X} \to \mathbb{R}$. And let $\mathcal{D}$ over $\mathcal{X} \times \{0,1\}$ be the data generating distribution. Let $\mathcal{Y} = \{0,1\}$.

For a loss function $\ell : \mathcal{Y} \times \mathbb{R} \to \mathbb{R}^+$, the goal in agnostic learning is to find some $h : \mathcal{X} \to \mathbb{R}$, such that,

$$R_\ell(h) = \mathop{\mathbb{E}}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}}\left[\ell(\mathbf{y}, h(\mathbf{x}))\right] \leq \min_{c\in\mathcal{C}} \mathop{\mathbb{E}}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}}\left[\ell(\mathbf{y}, c(\mathbf{x}))\right] + \epsilon.$$

How would we choose a loss function? Should we choose the hinge-loss? Or squared loss? Or logistic loss? Or something else? The choice of loss function is often application dependent. If we discovered that we would want to minimise a different loss function later on, then we have no option but to start the training process again.

Omniprediction tries to get around this problem by constructing a single predictor whose output can be post-processed differently depending on the loss function. Such a post-processed predictor should be competitive (in terms of risk) across a wide range of loss functions.

Definition 19 ((Gopalan et al., 2022)). *Let $\mathcal{C}$ be a family of functions and let $\mathcal{L}$ be a family of loss functions. For a distribution $\mathcal{D}$ over $\mathcal{X} \times \{0,1\}$, a predictor $p : \mathcal{X} \to [0,1]$ is an $(\mathcal{L}, \mathcal{C}, \epsilon)$-omnipredictor if for every $\ell \in \mathcal{L}$, there exists a function $k_\ell : [0,1] \to \mathbb{R}$, such that,*

$$R_\ell(k_\ell \circ p) \leq \min_{c\in\mathcal{C}} R_\ell(c) + \epsilon.$$

The key point in the above definition is that k_ℓ is a relatively simple univariate post-processing function. Crucially, it is only a function of $p(x)$, not directly of x! What is a good postprocessor? Suppose $q \in [0,1]$, then for a loss function ℓ, we define,

$$k^*_\ell(q) = \mathop{\text{argmin}}_{t\in\mathbb{R}} \mathop{\mathbb{E}}_{\mathbf{y}\sim\text{Ber}(q)}\left[\ell(\mathbf{y}, t)\right] = \mathop{\text{argmin}}_{t\in\mathbb{R}}\left(q\ell(1,t) + (1-q)\ell(0,t)\right).$$

If the argmin is a set, we can pick any element of it arbitrarily. What this is saying is that since we predict the probability of the outcome to be $p(x)$, then the particular value $t \in \mathbb{R}$ that minimizes the loss is precisely given by minimizing the loss when $\mathbf{y}$ is generated as $\text{Ber}(p(x))$.

It is worth observing that the ground truth, $p^*(x) = \mathbb{E}\left[\mathbf{y} \mid \mathbf{x} = x\right]$, is an omnipredictor.

Lemma 20. *For any class of functions $\mathcal{C}$ and class of loss functions $\mathcal{L}$, p^* is an $(\mathcal{L}, \mathcal{C}, 0)$-omnipredictor.*

4.1 Constructing omnipredictors through multicalibration

We will now see how multicalibrated predictors of particular types yield omnipredictors. Suppose $p : \mathcal{X} \to [0, 1]$ is τ-multicalibrated. Furthermore, let us assume that p only takes finitely many values (and that this is a relatively small number). In some sense, we've already seen how discretized predictors essentially satisfy the same fairness guarantees with only minor loss in the parameters. Such a predictor p yields a partition of the domain $\mathcal{X}$, for $v \in \text{range}(p)$, let $S_v = p^{-1}(v) \subset \mathcal{X}$. Then $\mathcal{S} = \{S_v \mid v \in \text{range}(p)\}$ is a partition of $\mathcal{X}$. We may as well assume that $p(x) = \mathbb{E}\left[\mathbf{y} | p(\mathbf{x}) = v\right]$ for $x \in S_v$. We'll see that such a predictor p is an omnipredictor for the class of convex, Lipschitz loss functions.

Covariance view of multicalibration

For a pair of random variables $(\mathbf{z}_1, \mathbf{z}_2)$, their covariance is defined as,

$$\text{cov}(\mathbf{z}_1, \mathbf{z}_2) = \mathbb{E}\left[\mathbf{z}_1 \mathbf{z}_2\right] - \mathbb{E}\left[\mathbf{z}_1\right] \mathbb{E}\left[\mathbf{z}_2\right].$$

When one of the random variables only takes values in $\{0, 1\}$, we have a simpler characterization.

Lemma 21. *Suppose $(\mathbf{z}, \mathbf{y})$ are random variables with $\mathbf{y}$ only taking values in $\{0, 1\}$. Then,*

$$\text{cov}(\mathbf{z}, \mathbf{y}) = \mathbb{P}\left[\mathbf{y} = 1\right] \left(\mathbb{E}\left[\mathbf{z} \mid \mathbf{y} = 1\right] - \mathbb{E}\left[\mathbf{z}\right]\right) = \mathbb{P}\left[\mathbf{y} = 0\right] \left(\mathbb{E}\left[\mathbf{z}\right] - \mathbb{E}\left[\mathbf{z} \mid \mathbf{y} = 0\right]\right).$$

Given a distribution $\mathcal{D}$ over $\mathcal{X} \times \{0, 1\}$ and a predictor p as defined above, define the random variable $\mathbf{v}$ which takes values in range(p). Let $\mathcal{D}_{\mathbf{v}}$ denote the conditional distribution $\mathcal{D}$ restricted to $\mathbf{x} \in S_{\mathbf{v}}$. Now, we can write the multicalibration condition as follows:

$$\begin{aligned}
\mathbb{E}\left[\left|\mathbb{E}\left[c(\mathbf{x})(\mathbf{y} - p(\mathbf{x})) | p(\mathbf{x})\right]\right|\right] &= \mathop{\mathbb{E}}_{\mathbf{v}}\left[\left|\mathop{\mathbb{E}}_{\mathcal{D}_{\mathbf{v}}}\left[c(\mathbf{x})\mathbf{y} - c(\mathbf{x})\mathbf{v}\right]\right|\right] \\
&= \mathop{\mathbb{E}}_{\mathbf{v}}\left[\left|\mathop{\mathbb{E}}_{\mathcal{D}_{\mathbf{v}}}\left[c(\mathbf{x})\mathbf{y} - c(\mathbf{x})\mathop{\mathbb{E}}_{\mathcal{D}_{\mathbf{v}}}\left[\mathbf{y}\right]\right]\right|\right] \\
&= \mathop{\mathbb{E}}_{\mathbf{v}}\left[\left|\text{cov}_{\mathcal{D}_{\mathbf{v}}}(c(\mathbf{x}), \mathbf{y})\right|\right].
\end{aligned}$$

Since p is τ-multicalibrated means that the LHS is at most τ for all $c \in \mathcal{C}$, this is also the case for the RHS.

We can now prove the following result.

Theorem 22 ((Gopalan et al., 2022))**.** *Let p be a τ-multicalibrated predictor as defined above for class $\mathcal{C}$ and distribution $\mathcal{D}$ over $\mathcal{X} \times \{0, 1\}$. Then if $\mathcal{L}$ is the class of convex, 1-Lipschitz loss functions, p is an $(\mathcal{L}, \mathcal{C}, 2\tau)$-omnipredictor.*

Proof. Consider any $c \in \mathcal{C}$ and $\ell \in \mathcal{L}$.

$$\begin{aligned}
R_\ell(c) &= \mathop{\mathbb{E}}_{\mathcal{D}}\left[\ell(\mathbf{y}, c(\mathbf{x}))\right] \\
&= \mathop{\mathbb{E}}_{\mathbf{v}}\left[\mathop{\mathbb{E}}_{\mathbf{y}\sim\mathcal{D}_\mathbf{v}}\left[\mathbb{E}\left[\ell(\mathbf{y}, c(\mathbf{x})) \mid \mathbf{y}\right]\right]\right] \\
&\geq \mathop{\mathbb{E}}_{\mathbf{v}}\left[\mathop{\mathbb{E}}_{\mathbf{y}\sim\mathcal{D}_\mathbf{v}}\left[\ell\left(\mathbf{y}, \mathbb{E}\left[c(\mathbf{x}) \mid \mathbf{y}\right]\right)\right]\right] && \text{(Convexity)} \qquad (3) \\
&\geq \mathop{\mathbb{E}}_{\mathbf{v}}\left[\mathop{\mathbb{E}}_{\mathbf{y}\sim\mathcal{D}_\mathbf{v}}\left[\ell\left(\mathbf{y}, \mathop{\mathbb{E}}_{\mathcal{D}_\mathbf{v}}\left[c(\mathbf{x})\right]\right)\right]\right] - \mathop{\mathbb{E}}_{\mathbf{v}}\left[\mathop{\mathbb{E}}_{\mathbf{y}\sim\mathcal{D}_\mathbf{v}}\left[\left|\mathbb{E}\left[c(\mathbf{x})|\mathbf{y}\right] - \mathop{\mathbb{E}}_{\mathcal{D}_\mathbf{v}}\left[c(\mathbf{x})\right]\right|\right]\right] && \text{(Lipschitzness)} \qquad (4) \\
&\geq \mathop{\mathbb{E}}_{\mathbf{v}}\left[\mathop{\mathbb{E}}_{\mathbf{y}\sim\mathcal{D}_\mathbf{v}}\left[\ell(\mathbf{y}, k^*_\ell(\mathbf{v}))\right]\right] - 2\tau = R_\ell(k^*_\ell \circ p) - 2\tau.
\end{aligned}$$

We've used the optimality of k^*_ℓ as for any v, $\mathop{\mathbb{E}}_{\mathcal{D}_v}[\mathbf{y}] = v$. We've also bounded the second term in Eq. (4) by 2τ using the multicalibration condition that requires some justification. We justify this below.

$$\begin{aligned}
\mathop{\mathbb{E}}_{\mathbf{v}}\left[\mathop{\mathbb{E}}_{\mathbf{y}\sim\mathcal{D}_\mathbf{v}}\left[\left|\mathbb{E}\left[c(\mathbf{x}) \mid \mathbf{y}\right] - \mathop{\mathbb{E}}_{\mathcal{D}_\mathbf{v}}\left[c(\mathbf{x})\right]\right|\right]\right] &= \mathop{\mathbb{E}}_{\mathbf{v}}\left[\sum_{b\in\{0,1\}}\left|\mathbb{P}_{\mathcal{D}_\mathbf{v}}\left[\mathbf{y} = b\right]\left(\mathbb{E}\left[c(\mathbf{x})|\mathbf{y}\right] - \mathop{\mathbb{E}}_{\mathcal{D}_\mathbf{v}}\left[c(\mathbf{x})\right]\right)\right|\right] \\
&= \mathop{\mathbb{E}}_{\mathbf{v}}\left[2\left|\mathrm{cov}_{\mathcal{D}_\mathbf{v}}(c(\mathbf{x}), \mathbf{y})\right|\right] \leq 2\tau.
\end{aligned}$$

Above, we have appealed to Lemma 21. □

References

Sílvia Casacuberta, Parikshit Gopalan, Varun Kanade, and Omer Reingold. How global calibration strengthens multiaccuracy. In *66th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2025)*, 2025.

Parikshit Gopalan, Adam Tauman Kalai, Omer Reingold, Vatsal Sharan, and Udi Wieder. Omnipredictors. In *13th Innovations in Theoretical Computer Science Conference (ITCS 2022)*. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2022.

David Haussler. Decision theoretic generalizations of the pac model for neural net and other learning applications. In *Proc. of the 1st Workshop on Algorithmic Learning Theory*, pages 21–41, 1990.

Ursula Hébert-Johnson, Michael P Kim, Omer Reingold, and Guy N Rothblum. Multicalibration: Calibration for the (computationally-identifiable) masses. In *Proceedings of the 35th International Conference on Machine Learning*, pages 1939–1948. PMLR, 2018.

Adam Tauman Kalai and Varun Kanade. Potential-based agnostic boosting. In *Advances in Neural Information Processing Systems*, volume 22, pages 880–888, 2009.

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

Vladimir Vapnik. *Statistical learning theory. 1998*, volume 3. Wiley, New York, 1998.