Calibrating an online predictor via Approachability

Princewill Okoroafor[*]
Department of Computer Science
Cornell University
Ithaca, NY 14850
pco9@cornell.edu

Wen Sun
Department of Computer Science
Cornell University
Ithaca, NY 14850
ws455@cornell.edu

Robert Kleinberg
Department of Computer Science
Cornell University
Ithaca, NY 14850
rdk@cs.cornell.edu

Abstract

Predictive models in ML need to be trustworthy and reliable, which often at the very least means outputting calibrated probabilities. This can be particularly difficult to guarantee in the online prediction setting when the outcome sequence can be generated adversarially. In this paper we introduce a technique using Blackwell's approachability theorem for taking an online predictive model which might not be calibrated and transforming its predictions to calibrated predictions without much increase to the loss of the original model. Our proposed algorithm achieves calibration and accuracy at a faster rate than existing techniques [Kuleshov and Ermon, 2017] and is the first algorithm to offer a flexible tradeoff between calibration error and accuracy in the online setting. We demonstrate this by characterizing the space of jointly achievable calibration and regret using our technique.

1 Introduction

In the online learning setting, a predictive model, also known as a forecaster, gives a probability value prediction at each time step, and its performance is evaluated based on a loss function. For the class of loss function known as a proper scoring rule, the only way to minimize that score is to predict the true probabilities of an outcome.

However, most training methods for predictive models do not guarantee calibrated probability values. There has been a large body of work highlighting the need for calibrated probability estimates (i.e., models that are able to assess their uncertainty) [Kleinberg, 2018] and on how to obtain these calibrated probability estimates Foster [1999]. In the offline setting, this is generally done by some post-processing of the data to remap the probability values to calibrated probability estimates in a way that minimizes the increase in loss, such as by post-hoc calibration or recalibration (cite). In contrast, in the online prediction setting, little work has been done on this subject. Recently, Kuleshov and Ermon [2017] and Foster and Hart [2021] have presented various approaches for taking an online predictive model and transforming its predictions without major increase in loss. Kuleshov and Ermon [2017] introduces this problem as an online recalibration problem, and provides an algorithm for achieving epsilon accuracy relative to the loss function using a connection between calibration and internal regret. In this paper, we show that their result can be significantly improved by using

[*] Use footnote for providing further information about author (webpage, alternative address)—*not* for acknowledging funding agencies.

Preprint. Under review.

Blackwell's Approachability Theorem. We present an algorithm, making use of approachability, that achieves recalibration at a much faster rate than the internal regret minimization algorithm by Kuleshov and Ermon [2017]. We also characterize the achievable amount of calibration and regret as a function of the time horizon; more precisely, we study for which exponents a, b does there exist a forecasting algorithm that guarantees at most T^a calibration error and no more than T^b regret relative to scoring rule loss functions. We provide the first algorithm that offers a flexible tradeoff between calibration error and regret in the online setting. We generalize our result to L_p calibration for $p \geq 1$.

1.1 Motivation

Calibrating probability predictions As the prevalence of machine learning systems in decision-making settings grows, it is essential that the predictions they provide are trustworthy, especially in applications where the confidence associated with the prediction is at least as important as the prediction itself. Neural networks have been found to be poor at assessing their own uncertainty [Guo et al., 2017], and as a result, may output probability values that do not match the true probabilities of outcomes. This can have serious consequences; machine learning systems have been known to propagate unintended but harmful discrimination, as shown by Buolamwini and Gebru [2018] for image classification and Bolukbasi et al. [2016] for natural language tasks. One proposed method for addressing the issue of assessing uncertainty is calibration [Hebert-Johnson et al., 2018]. Calibration requires that the probability estimates from the ML model match the true distribution of the outcome; for example, for a binary class, if a model outputs a probability of 0.3 a certain number of times, the proportion of true outcomes should be 30 percent across the total instances when the model predicted 0.3. In the online setting, many works have proposed techniques for how to achieve calibrated probability estimates, even in the adversarial setting [Foster, 1999, Mannor and Stoltz, 2010, Abernethy et al., 2011].

Limitations of calibration While calibration is a useful property for online predictors to have, calibration is not sufficient and does not fully reflect domain specific knowledge. For example, consider two ML weather forecasters. Suppose the true outcome is that it rains once every two days. Forecaster 1 predicts 50 percent chance of rain every day, and Forecaster 2 predicts 0 percent chance of rain on the days it does not rain and 100 percent on the days it does. Observe that both of these forecasters are equally calibrated; however, the second forecaster is a better predictor of the likelihood of rain. Calibration does not capture this fact. Although calibration does not imply accuracy, accuracy does imply calibration, simply because being accurate requires an understanding of the outcome distribution. This is why, in practice, proper scoring rules are used to assess the accuracy of predictions [Gneiting and Raftery, 2007].

Incorporating expert/domain-specific knowledge in online prediction models Forecaster 2 is an example of a forecaster that reflects domain-specific knowledge and is also calibrated. However, it is also possible for a forecaster that acts on domain specific knowledge to be poorly calibrated. Consider a third forecaster in the same weather prediction setting which predicts 20 percent chance of rain on the days it does not rain, and 80 percent chance of rain on the days that it does. This predictor is poorly calibrated, because it incurs a calibration error of 0.2 for every decision. However, compared to Forecaster 1, its predictions still reflect a domain-specific understanding of the probability distribution. The goal of our work is to take a model such as this third forecaster and transform its predictions in an online setting to achieve calibration while still making decisions that are informed by domain knowledge.

1.2 Problem formulation

In this paper, we focus on a class of loss functions known as strictly proper scoring rules. We refer the reader to Section 2.2 for a introduction on the subject.

Consider an online prediction environment where the timing of each round of the prediction process is as follows.

1. An oracle reveals a prediction q_t.
2. The algorithm must make a prediction p_t.
3. The actual label $y_t \in \{0, 1\}$ is revealed.

4. The algorithm receives a score $S(p_t, y_t)$ is revealed.

At the end of T rounds, the following quantities are calculated.

- The forecaster's cumulative score is $S_f = \sum_{t=1}^{T} S(p_t, y_t)$.
- The oracle's cumulative score is $S_o = \sum_{t=1}^{T} S(q_t, y_t)$.
- The forecaster's average regret is $\frac{1}{T}(S_f - S_o)$.
- The forecaster's ℓ_1-calibration error is

$$\sum_{p \in [0,1]} \left| \frac{1}{T} \sum_{t=1}^{T} (y_t - p) \cdot \mathbb{1}_{p_t = p} \right| .$$

(Although written as a sum over all $p \in [0, 1]$, the sum is actually finite because there are only finitely many p for which the summand is nonzero.)

For the sake of generality, our model makes no assumptions about how the oracle's predictions are generated, except that if the algorithm is randomized the oracle cannot anticipate the algorithm's *future* coin-tosses. This means, for example, that our simple prediction model subsumes more elaborate models in which the predictions q_t are generated by a contextual bandit algorithm, or by a pre-trained model such as a deep neural network, using domain-specific features observed at time t or earlier.

Our work addresses the question: For which exponent pairs (a, b) is there a forecasting algorithm that guarantees regret $\tilde{O}(T^a)$ and calibration error $\tilde{O}(T^b)$? The purpose of this paper is to propose a method of tackling this question using Blackwell's Approachability Theorem.

1.3 Our results

For the notion of regret described above, we show that there is a family of approachability-based algorithms, parameterized by $\varepsilon > 0$, that simultaneously achieves calibration $O(\varepsilon + 1/\sqrt{\varepsilon T})$ and average regret $O(\varepsilon^2 + 1/\sqrt{\varepsilon T})$. This is a significant improvement from the result by Kuleshov and Ermon [2017], which achieves calibration $O(\varepsilon + 1/\sqrt{\varepsilon^2 T})$ and average regret $O(\varepsilon + 1/\sqrt{\varepsilon^2 T})$. The improved dependence on ε is significant in practice because it impacts how many samples, T, are required in order to make the average regret less than some specified upper bound, δ. For example, to make $\varepsilon^2 + 1/\sqrt{\varepsilon T}$ less than δ one would set $T = O(\delta^{-5/2})$ and $\varepsilon = O(\delta^{1/2})$, whereas to make $\varepsilon + 1/\sqrt{\varepsilon^2 T}$ less than δ requires $T = O(\delta^{-4})$ and $\varepsilon = O(\delta)$. For $\delta = 0.1$ this amounts to the difference between a few hundred samples versus more than ten thousand.

By choosing ε appropriately, we show that our algorithm can be designed to achieve the best known calibration upper bound of $T^{-\frac{1}{3}}$ while limiting regret to no more than $T^{-\frac{1}{3}}$. If one is more interested in minimizing regret, we also show that ε can be chosen to achieve regret of $T^{-\frac{2}{5}}$ while limiting calibration error to no more than $T^{-\frac{1}{5}}$. The algorithm allows for a linear interpolation between these two bounds. That is, for any x in the interval $\left[\frac{1}{3}, \frac{2}{5}\right]$, we can set $\varepsilon = T^{-2x}$ to achieve calibration $O(T^{2x-1})$ while simultaneously achieving regret $O(T^{-x})$. We generalize this to ℓ_p calibration for $p \geq 1$.

1.4 Comparison to prior work

Calibration and proper scoring rules Foster [1999] first reduced calibration to approachability. Since then, a number of alternative proofs of calibration using reductions to approachability have emerged [Mannor and Stoltz, 2010, Abernethy et al., 2011]. Unlike in the standard setting of calibrated binary sequence prediction, our setting ("recalibration") incorporates side information in the form of an oracle who makes a prediction at each timestep. In the standard calibration problem, the goal is to minimize calibration error. In the recalibration problem, the algorithm must attain two goals simultaneously: sublinear calibration error and sublinear regret relative to the oracle's predictions. To achieve both of these objectives we need to modify the vector payoffs and the approachable set used in the standard reduction from calibration to approachability. The main technical challenge we had to overcome in this work is verifying that the modified set is indeed approachable in the game with the modified vector payoffs. After showing that the modified set is indeed approachable, we rely on a reduction from approachability to Online Linear Optimization by Abernethy et al. [2011] to construct an algorithm for recalibration.

Recalibration in offline setting In the offline setting, calibrated predictions are usually constructed using methods such as Platt Scaling [Platt, 1999] and isotonic regression [Niculescu-Mizil and Caruana, 2005]. In the context of binary classification, these methods reduce the problem of outputting calibrated predictions to a one-dimensional regression problem. Given data $\{(x_i, y_i)\}_{i=1}^{n}$, train a model $f(s)$ to predict $p_i = f(s)$ from uncalibrated scores $s_i = g(x_i)$ produced by a classifier g. These techniques are particularly suited for the offline setting where the training and the calibration phases of the algorithm can be separated and thus, do not apply in the online setting and can fail when the test distribution does not match the training distribution. Our results, on the other hand, are robust to adversarial manipulations.

Recalibration in online setting Kuleshov and Ermon [2017] present an algorithm for recalibration, that is, for achieving ε calibration and ε regret simultaneously at a rate of $1/\varepsilon\sqrt{T}$. They achieve this by running $1/\varepsilon$ many calibration algorithms in parallel for each prediction interval that the expert (called "oracle" in our work, "blackbox predictor" in theirs) makes. This method works because calibrated predictors have been shown to minimize internal regret [Cesa-Bianchi and Lugosi, 2006]. They are able to bound the regret by the internal regret, which is bounded by calibration error, which itself is bounded by ε. The two main issues with their approach are first, the additional cost of running $1/\varepsilon$ calibration algorithms in parallel; and second, having to rely on the calibration error bound in order to bound the regret. Our technique bypasses these constraints by appealing to Blackwell's Approachability Theorem. With Blackwell's Approachability Theorem, we can treat this problem as a vector-valued game where one tries to simultaneously minimize the calibration and regret components of the vector. Instead of having $1/\varepsilon$ different calibration algorithms, we have only a single calibration algorithm which also takes regret into account. The single calibration algorithm achieves a stronger guarantee by leveraging the fact that proper scoring rules incentivize calibration. We also take this a step further by giving precise error bounds as a function of the time horizon, and allowing a trade-off between calibration error and regret.

Another closely related result is contained a preprint by Foster and Hart [2021]. In their paper on "calibeating," they present a method for transforming expert predictions to calibrated predictions, while measuring accuracy against an even more strict benchmark than ours: they compare the algorithm's loss to that of the expert after the calibration error has been removed, a benchmark called the "refinement score". They prove this for the loss function known as the Brier score, when calibration is quantified using the ℓ_2 objective. Our work pertains to a much broader class of scoring rules and permits calibration to be quantified using ℓ_p for any $p \geq 1$, but we measure accuracy with respect to a weaker benchmark, namely the expert's average loss.

2 Background

2.1 Calibration

Let $y_1, y_2, \ldots \in \{0, 1\}$ be a sequence of outcomes, and $p_1, p_2, \ldots \in [0, 1]$ a sequence of probability predictions by a forecaster. We define for every T and every probability interval $[a, b]$, where $0 \leq a \leq b \leq 1$, the quantities

$$n_T(p, \varepsilon) := \sum_{t=1}^{T} \mathbb{I}[p_t \in (p - \varepsilon/2, p + \varepsilon/2)], \qquad \rho_T(p, \varepsilon) := \frac{\sum_{t=1}^{T} y_t \mathbb{I}[p_t \in (p - \varepsilon/2, p + \varepsilon/2)]}{n_T(p, \varepsilon)}.$$

The quantity $\rho_T(p - \varepsilon/2, p + \varepsilon/2)$ should be interpreted as the empirical frequency of $y_t = 1$, up to round T, on only those rounds where the forecaster's prediction was "roughly" equal to p. The goal of calibration, of course, is to have this empirical frequency $\rho_T(p, \varepsilon)$ be close to the estimated frequency p. To capture how close an algorithm $\mathcal{A}$ to being ε-calibrated, we use a notion of rate below.

Definition 1. *Let $\mathcal{P}(\varepsilon)$ denote the set of midpoints of the intervals $[i\varepsilon, (i+1)\varepsilon]$ for $i = 0, 1, \ldots, \lfloor \varepsilon^{-1} \rfloor$. Let the (ℓ_p, ε)-calibration rate for forecaster $\mathcal{A}$ be*

$$C^{\varepsilon}_{p,T}(\mathcal{A}) = \max \left\{ 0,\ \frac{1}{T} \left(\sum_{z \in \mathcal{P}(\varepsilon)} n_T(z, \varepsilon) \cdot |z - \rho_T(z, \varepsilon)|^p \right)^{1/p} - \frac{\varepsilon}{2} \right\} \tag{1}$$

*We say that a forecaster is (ℓ_p, ε)-*calibrated *if $C^{\varepsilon}_{p,T}(\mathcal{A}) = o(1)$. This in turn implies $\limsup_{T \to \infty} C^{\varepsilon}_{p,T}(\mathcal{A}) = 0$.*

2.2 Proper Scoring Rules, Regret, and Recalibration

Kuleshov and Ermon [2017] define the problem of online recalibration in which the task is to transform a sequence of uncalibrated forecasts q_t into predictions p_t that are calibrated and almost as accurate as the original q_t. They show that this objective is achievable if and only if the loss function used to measure forecast accuracy is a *proper scoring rule*, a term which we now define.

Suppose there is a future event denoted by a random variable X with a finite set $\mathcal{Y}$ of possible outcomes. For example: $\mathcal{Y} = \{\text{rain}, \text{no rain}\}$. Let $\Delta_{\mathcal{Y}}$ be the set of probability distributions on $\mathcal{Y}$. An algorithm reports a probability distribution $p \in \Delta_{\mathcal{Y}}$, observes the outcome $y \in \mathcal{Y}$ and receives a score $S(p, y)$.

Definition 2. *A scoring rule is a function $S : \Delta_{\mathcal{Y}} \times \mathcal{Y} \mapsto \mathbb{R}$. It is proper if accurately reporting the distribution of X minimizes the expected score: that is, for all distributions $p, q \in \Delta_{\mathcal{Y}}$*

$$\mathbb{E}_{X \sim p}\left[S(p, X)\right] \leq \mathbb{E}_{X \sim p}\left[S(q, X)\right]. \tag{2}$$

Scoring rule S is strictly proper *if Inequality* (2) *is strict whenever $p \neq q$.*

Note that we adopt the convention that the scoring rule is a loss function rather than a payoff function, i.e. p is the unique probability that minimizes $S(\cdot, p)$ rather than maximizing it. We extend S to the domain $\Delta_{\mathcal{Y}} \times \Delta_{\mathcal{Y}}$ by making it linear in the second variable. In other words, $S(q, p)$ is shorthand for $\mathbb{E}_{X \sim p}\left[S(q, X)\right]$. We assume the scoring rule S is Lipschitz-continuous in its first variable, with Lipschitz constant L_S, i.e.

$$\forall p, q \in \Delta_{\mathcal{Y}}\ \forall y \in \mathcal{Y} \qquad |S(p, y) - S(q, y)| \leq L_S \cdot \|p - q\|,$$

where $\|p - q\|$ denotes the total variation distance between p and q.

We measure a forecaster's accuracy by comparing with the score of the oracle. Let $q_1, q_2, \ldots \in [0, 1]$ be a sequence of probability predictions by an oracle.

Definition 3. *Let the regret at timestep t for forecaster $\mathcal{A}$ be*

$$r_t(p_t, q_t) = S(p_t, y_t) - S(q_t, y_t)$$

This leads to an average regret *of $R_T(A) = \frac{1}{T}\sum_{t=1}^{T} r_t(p_t, q_t)$. We say that a forecaster has* no-regret *if $R_T(\mathcal{A}) = o(1)$. This in turn implies $\limsup_{T \to \infty} R_T(\mathcal{A}) = 0$. We also say a forecaster has* δ-regret rate *if $R_T(\mathcal{A}) \leq \delta$.*

Definition 4. *Let the $(\ell_p, \varepsilon, \delta)$-recalibration rate for forecaster $\mathcal{A}$ be*

$$C^{\varepsilon,\delta}_{p,T}(\mathcal{A}) = \max \left\{ 0, C^{\varepsilon}_{p,T}(\mathcal{A}), R_T(\mathcal{A}) - \frac{\delta}{2} \right\} \tag{3}$$

*We say that a forecaster is $(\ell_p, \varepsilon, \delta)$-*recalibrated *if $C^{\varepsilon,\delta}_{p,T}(\mathcal{A}) = o(1)$. This in turn implies $\limsup_{T \to \infty} C^{\varepsilon,\delta}_{p,T}(\mathcal{A}) = 0$.*

This definition is analogous to Definition 4 in Kuleshov and Ermon [2017], except that we have quantified the calibration and accuracy using two parameters, ε and δ, whereas they use ε for both.

2.3 Blackwell's Approachability Theorem

Blackwell approachability [Blackwell, 1956] generalizes the problem of playing a repeated two-player zero-sum game to games whose payoffs are vectors instead of scalars. In a Blackwell approachability game, at all times t, two players interact in this order: first, Player 1 selects an action $x_t \in X$; then, Player 2 selects an action $y_t \in Y$; finally, Player 1 incurs the vector-valued payoff $u(x_t, y_t) \in \mathbb{R}^d$. The sets X, Y of player actions are assumed to be compact convex subsets of finite-dimensional vector spaces, and u is assumed to be a biaffine function on $X \times Y$. Player 1's objective is to guarantee that the average payoff converges to some desired closed convex target set $\mathcal{S} \subseteq \mathbb{R}^d$. Formally, given target set $\mathcal{S} \subseteq \mathbb{R}^d$, Player 1's goal is to pick actions $x_1, x_2, \ldots \in X$ such that no matter the actions $y_1, y_2, \ldots \in Y$ played by Player 2,

$$\texttt{dist}\left(\frac{1}{T}\sum_{t=1}^{T} u(x_t, y_t), \mathcal{S}\right) \to 0 \quad \text{as} \quad T \to \infty \tag{4}$$

The action x_t is allowed to depend on the realized payoff vectors $u_s(x_s, y_s)$ for $s = 1, 2, \ldots, t-1$. We say the set S is approachable if Player 1 has a strategy that attains the goal (4) no matter how Player 2 plays. Blackwell's Approachability Theorem asserts that a convex set $\mathcal{S} \subset \mathbb{R}^d$ is approachable if and only if every closed halfspace containing $\mathcal{S}$ is approachable. Henceforth we say refer to this necessary and sufficient condition as *halfspace-approachability*.

In this paper, we shall adopt the notation, $\texttt{dist}_p(x, \mathcal{S})$ to be the $\min_{s \in S} \|x - s\|_p$. Due to the nature of our vector payoff formulation, we will also use a compounded notation of distance: when $x = (x_1, x_2) \in \mathbb{R}^d \times \mathbb{R}$ and $\mathcal{S} = \mathcal{S}_1 \times \mathcal{S}_2 \subseteq \mathbb{R}^d \times \mathbb{R}$, we will write $\texttt{dist}_p^r(x, \mathcal{S})$ to denote $||(\texttt{dist}_p(x_1, \mathcal{S}_1), \texttt{dist}_1(x_2, \mathcal{S}_2))||_r$. We will refer to the ℓ_p ball $\in R^d$ of radius r centered at the origin as $B_p^d(r)$. To specify the target convex set for our Blackwell's instance, we shall refer to the following set definitions.

Definition 5. *For a fixed $\varepsilon, \delta, m > 0$ and distance metric ℓ_p, let*

$$\mathcal{S}_p^m(\varepsilon, \delta) = \left\{(x, z) \mid \ x \in \mathbb{R}^{m+1}, z \in \mathbb{R} \quad s.t \quad ||x||_p \leq \frac{\varepsilon}{2},\ z \leq \frac{\delta}{2}\right\} \tag{5}$$

let

$$\mathcal{K}_q^m = \left\{(a, b) \mid \ a \in \mathbb{R}^{m+1}, b \in \mathbb{R} \quad s.t \quad ||a||_q \leq 1,\ 0 \leq b \leq 1\right\} \tag{6}$$

Observe that $\mathcal{S}_p^m(\varepsilon, \delta)$ can be thought of as $B_p^{m+1}(\varepsilon/2) \oplus \left(-\infty, \frac{\delta}{2}\right]$. Similarly, $\mathcal{K}_q^m$ can be expressed as $B_q^{m+1}(1) \oplus [0, 1]$

We now give an equivalent and alternative characterization of the definition of recalibration rate (Definition 4): let the *recalibration vector* at time T denoted $\mathbf{v}_T$ be given by: $\mathbf{v}_T = \mathbf{c}_T \oplus R_T$ where $\mathbf{c}_T(i) = \frac{n_T(i\varepsilon, \varepsilon)}{T} \left|i\varepsilon - \rho_T(i\varepsilon, \varepsilon)\right|^p$, and $R_T = \frac{1}{T}\sum_{t=0}^{T} S(p_t, y_t) - S(q_t, y_t)$

Lemma 6.

$$C_{p,T}^{\varepsilon,\delta}(\mathcal{A}) = \texttt{dist}_p^{\infty}\left(\mathbf{v}_T, \mathcal{S}_p^{\lfloor \varepsilon^{-1} \rfloor}(\varepsilon, \delta)\right) \tag{7}$$

3 Vector-valued payoff game

We now describe the construction of the payoff game that allows us to reduce recalibration to approachability. This payoff game modifies the standard construction for calibration in [Foster, 1999, Abernethy et al., 2011] by adding an additional dimension for regret.

For any $m \geq \sqrt{4L_s}$ where L_s is the lipschitz constant of the scoring rule, we will show how to construct an $(\ell_1, \varepsilon, \delta)$-recalibrated forecaster for $\varepsilon = \frac{1}{m}$ and $\delta = \frac{4L_s}{m^2}$. On each round t, after observing the oracle's prediction q_t, a forecaster will randomly predict a probability $p_t \in \{0/m, 1/m, 2/m, \ldots, (m-1)/m, 1\}$, according to the distribution $\mathbf{w}_t$, that is $\Pr(p_t = i/m) = w_t(i)$. We define a vector-valued game. Let the player choose $\mathbf{w}_t \in \mathcal{X} := \Delta_{m+1}$, and the adversary choose $y_t \in \mathcal{Y} := [0, 1]$, and the payoff vector will be $\boldsymbol{\ell}_t(\mathbf{w}_t, y_t) = \mathbf{c}(\mathbf{w}_t, y_t) \oplus r_t(\mathbf{w}_t, y_t)$ [2] defined as follows:

$$\mathbf{c}(\mathbf{w}_t, y_t) := \left\langle \mathbf{w}_t(0)\left(y_t - \frac{0}{m}\right), \mathbf{w}_t(1)\left(y_t - \frac{1}{m}\right), \ldots, \mathbf{w}_t(m)(y_t - 1)\right\rangle \tag{8}$$

[2] Blackwell's original theorem only applies to fixed losses. Our reduction remains valid because Abernethy et al. [2011]'s proof applies to changing losses.

$$r_t(\mathbf{w}_t, y_t) := \sum_{i=0}^{m} \mathbf{w}_t(i) r_t\left(\frac{i}{m}, q_t\right) = \sum_{i=0}^{m} \mathbf{w}_t(i)\left(S\left(\frac{i}{m}, y_t\right) - S(q_t, y_t)\right) \tag{9}$$

The set we wish to approach is

$$\left\{(x, z) \mid \; x \in \mathbb{R}^{m+1}, z \in \mathbb{R} \quad \text{s.t} \quad ||x||_p \le \frac{1}{m},\; z \le \frac{4L_s}{m^2}\right\} \tag{10}$$

denoted by as $\mathcal{S}_p^m\left(\frac{1}{m}, \frac{4L_s}{m}\right)$ by Definition 5. For simplicity of notation, henceforth, we shall refer to this set as $\mathcal{S}_p^m$ (or $\mathcal{S}$).

Theorem 7. *For the vector-valued game defined in* (8)*, the set $\mathcal{S} = \mathcal{S}_p^m$ is approachable.*

Proof. To show that the set $\mathcal{S}$ is approachable, we need to show that every halfspace containing $\mathcal{S}$ is approachable. We do this in Lemma 8 □

Lemma 8 (Halfspace Lemma)**.** *Let H be a halfspace containing $\mathcal{S}$. H is approachable.*

Proof of Lemma 8. Let H be a halfspace containing $\mathcal{S}$ defined by the equation $\langle a, x\rangle + bz \le \theta$ for $x \in \mathbb{R}^{m+1}, z \in \mathbb{R}$. It must be the case that

$$\max\left\{\langle a, x\rangle + bz \mid \; ||x||_p \le \frac{1}{m}, z \le \frac{4L_s}{m^2}\right\} \le \theta$$

First, we need $b \ge 0$, since we can choose z to violate this constraint otherwise. Secondly, we need $\theta \ge \left(\frac{||a||_p}{m} + \frac{4bL_s}{m^2}\right)$, since we can choose x and z to violate this constraint otherwise. Thus, if $\mathcal{S} \subseteq H$, then both conditions $b \ge 0$ and $\theta \ge \left(\frac{||a||_p}{m} + \frac{4bL_s}{m^2}\right)$ must hold for H. Conversely, if both conditions $b \ge 0$ and $\theta \ge \left(\frac{||a||_p}{m} + \frac{4bL_s}{m^2}\right)$ hold for H, then $\mathcal{S} \subseteq H$. This is because for any $(x, z) \in \mathcal{S}$, $\langle a, x\rangle + bz \le \left(\frac{||a||_p}{m} + \frac{4bL_s}{m^2}\right) \le \theta$ and if $b \le 0$, we can obtain a contradiction by choosing $z < -\frac{\theta}{b}$. WLOG, we will assume $\theta = \left(\frac{||a||_p}{m} + \frac{4bL_s}{m^2}\right)$, since approachability of a halfspace defined by $\langle a, x\rangle + bz \le \left(\frac{||a||_p}{m} + \frac{4bL_s}{m^2}\right)$ implies approachability of $\langle a, x\rangle + bz \le \theta$ for $\theta \ge \left(\frac{||a||_p}{m} + \frac{4bL_s}{m^2}\right)$. That is, we will only concern ourselves with proving halfspace-approachability for halfspaces such that $\theta = \left(\frac{||a||_p}{m} + \frac{4bL_s}{m^2}\right)$. For a halfspace such that $a = \mathbf{0}$, we can approach it by simply following the oracle's predictions. This gives us zero regret, and thus guarantees that $bz = 0 \le \frac{4L_s}{m^2}$. If $a \ne \mathbf{0}$, then we can consider the halfspace normalized by $||a||_p$, that is, the halfspace defined by $a' = \frac{a}{||a||_p}, b' = \frac{b}{||a||_p}$ and $\theta = \frac{1}{m} + \frac{4b' L_s}{m^2}$. Since $||a'|| = 1$ and $b' \ge 0$, by Lemma 9, this halfspace is approachable. Consequently, any halfspace containing $\mathcal{S}$ is approachable. □

Lemma 9. *For a fixed (a, b) such that $\|a\|_p = 1$ and $b \ge 0$, the halfspace H_1 parameterized by (a, b) defined by below is approachable*

$$H_1 := \left\{(x, z) \mid \; \langle a, x\rangle + bz \le \frac{1}{m} + \frac{4bL_s}{m^2}\right\} \tag{11}$$

The full proof can be found in Section A.1. We provide a proof sketch here. To show that H_1 is approachable, we will find a mixed distribution for the forecaster (i.e, a probability distribution over $p \in \{0/m, 1/m, 2/m, \ldots, (m-1)/m, 1\}$) such that $\mathbb{E}_p\left[\langle a, \boldsymbol{\ell}_c(p, y)\rangle + b\boldsymbol{\ell}_r(p, y)\right] \le \frac{1}{m} + \frac{4bL_S}{m^2}$ for any $y \in \{0, 1\}$. For simplicity, define

$$f(i, y) = \left\langle a, \boldsymbol{\ell}_c\left(\frac{i}{m}, y\right)\right\rangle + b\boldsymbol{\ell}_r\left(\frac{i}{m}, y\right) \quad \text{and} \quad F_i = \begin{bmatrix} f(i,0) \\ f(i,1) \end{bmatrix}$$

so our objective becomes to show that there exists a distribution p over $\frac{i}{m} \in \{0, \ldots, m\}$ such that $\mathbb{E}_p\, f(i, y) \le \frac{1}{m} + \frac{4bL_S}{m^2}$ for $y \in \{0, 1\}$, or equivalently that the vector $\mathbb{E}_p\, F_i$ belongs to the quadrant-shaped set $(-\infty, \frac{1}{m} + \frac{4bL_S}{m^2}] \times (-\infty, \frac{1}{m} + \frac{4bL_S}{m^2}]$. We will be choosing p to be either a point-mass on $\frac{i}{m}$ for some i, or a distribution on two consecutive values in the set $\{0, \frac{1}{m}, \frac{2}{m}, \ldots, 1\}$. For $p \in [0,1]$ let $D(p)$ denote the vector corresponding to the scoring rule term in F_i.

$$D(p) = b \cdot \begin{bmatrix} S(p,0) - S(q_t,0) \\ S(p,1) - S(q_t,1) \end{bmatrix}$$

As a result of the fact that S is a proper scoring rule, an important observation is that the curve formed by $D(p)$ is convex and its tangent lines are parallel to $\begin{bmatrix} p \\ p-1 \end{bmatrix}$. Thus, $F_0, F_1, \ldots, F_m$ are points on a sequence of tangent lines to the convex curve formed by $D(p)$. Additionally, we can show that F_0 lies in the left half-plane while F_m must belong to the lower half-plane. Thus, $F_0, F_1, \ldots, F_m$ are always in the second, third or fourth quadrants and lie on lines with a slopes that are slowly changing from negative to positive. If F_i belongs to the third quadrant — that is, the set $(-\infty, 0] \times (-\infty, 0]$ — then we choose p to be a point-mass on i. This guarantees that $\mathbb{E}_p F_i \leq \frac{1}{m} + \frac{4bL_S}{m^2}$. Otherwise, there must be at least one index j such that F_j lies in the second quadrant while F_{j+1} lies in the fourth quadrant. Using plane geometry, we show that the line segment joining F_j and F_{j+1} intersects the set $(-\infty, \frac{1}{m} + \frac{4bL_S}{m^2}] \times (-\infty, \frac{1}{m} + \frac{4bL_S}{m^2}]$ as required.

4 Efficient Algorithm via Online Linear Optimization

We now show how the results in the previous section lead to an efficient algorithm for online recalibration.

Theorem 10. *For any m, there exists a $(\ell_1, \frac{1}{m}, \frac{4L_s}{m^2})$-online recalibration algorithm that runs in time $O(\log m)$ per iteration and guarantees a recalibration rate of $O\left(\sqrt{\frac{m}{T}}\right)$*

We proceed according to the steps of the reduction from Approachability to OLO outlined in Abernethy et al. [2011]. To prove the result, we provide a convex set $\mathcal{K}$ and and express the distance of a loss vector to the set $\mathcal{S}$ we wish to approach as an optimization over the convex set $\mathcal{K}$. We do so in Lemma 11. Then, we present an algorithm (halfspace oracle) such that given a halfspace $\boldsymbol{\theta}_t \in \mathcal{K}$, it returns a distribution $\mathbf{w}_t \in \Delta_{m+1}$ with the guarantee that $\langle \boldsymbol{\ell}_t(\mathbf{w}_t, y_t), \boldsymbol{\theta}_t \rangle \leq \frac{1}{m} + \frac{4L_s}{m^2}$. Lastly, we present an algorithm for recalibration that uses Online Gradient Descent Zinkevich [2003] to select the halfspace $\boldsymbol{\theta}_t \in \mathcal{K}$ to approach at each timestep.

We set the convex set $\mathcal{K}$ to be $\mathcal{K}_\infty^m$. This is defined in 5 as

$$\mathcal{K}_\infty^m := \left\{(a,b) \mid \ a \in \mathbb{R}^{m+1}, b \in \mathbb{R} \quad \text{s.t} \quad ||a||_\infty \leq 1, \ 0 \leq b \leq 1\right\} \tag{12}$$

This is an appropriate choice of $\mathcal{K}$ due to Lemma 11, since it allows us to upper bound the distance to S in terms of linear optimization objective over the set $\mathcal{K}_\infty^m$.

Lemma 11. *For any vector $\mathbf{x} \in \mathbb{R}^{m+2}$ such that $\|\mathbf{x}_{1:m+1}\|_p \geq 1/m$, and $|\mathbf{x}_{m+2}| \geq \frac{4L_s}{m^2}$,*

$$\mathtt{dist}_p^1\left(\mathbf{x}, \mathcal{S}_p^m\right) = -\frac{1}{m} - \frac{4L_s}{m^2} - \min_{\theta \in K_q^m} \langle -\mathbf{x}, \theta \rangle \tag{13}$$

where q is such that $\frac{1}{p} + \frac{1}{q} = 1$

We defer the proof of the lemma 11 to the appendix. The usefulness of the lemma above is that it allows us to combine the approachability guarantee of 7 to upper bound the distance to the target convex set in terms of regret of an online linear optimization algorithm.

$$\mathtt{dist}_p^1\left(\frac{1}{T}\sum_{t=1}^T \boldsymbol{\ell}_t(\mathbf{w}_t, y_t), \mathcal{S}_p^m\right) = -\frac{1}{m} - \frac{4L_s}{m^2} - \min_{\theta \in \mathcal{K}_q^m} \left\langle -\frac{1}{T}\sum_{t=1}^T \boldsymbol{\ell}_t(w_t, y_t), \boldsymbol{\theta} \right\rangle \tag{14}$$

$$\leq \frac{1}{T}\left(\sum_{t=1}^T \langle -\boldsymbol{\ell}_t(\mathbf{w}_t, y_t), \boldsymbol{\theta}_t \rangle - \min_{\boldsymbol{\theta} \in \mathcal{K}_q^m} \sum_{t=1}^T \langle -\boldsymbol{\ell}_t(\mathbf{w}_t, y_t), \boldsymbol{\theta} \rangle\right) \tag{15}$$

where the first inequality and the second inequality follow from the approachability guarantee of 7: for any halfspace θ_t, there exists a distribution $\mathbf{w}_t$ such that $\langle \boldsymbol{\ell}_t(\mathbf{w}_t, y_t) \rangle \leq \frac{1}{m} + \frac{4L_s}{m^2}$ for any $y_t \in \{0,1\}$

Constructing the Halfspace Oracle Given any $\boldsymbol{\theta}_t \in \mathcal{K}$, we must construct $\mathbf{w} \in \Delta_{m+1}$ so that $\langle \boldsymbol{\ell}_t(\mathbf{w}_t, y_t), \boldsymbol{\theta}_t \rangle \leq \frac{1}{m} + \frac{4L_s}{m^2}$ for any y_t. The proof of approachability for 9 is a constructive one and describes how to choose $\mathbf{w}_t \in \Delta_{m+1}$ given $\boldsymbol{\theta}_t$. However, we need to make the process more precise and show that the construction can be done in $O(\log m)$ time. We do so in Section A.2 of the Appendix.

The Learning Algorithm We use the Online Gradient Descent algorithm Zinkevich [2003]. As Abernethy et al. [2011] points out in the calibration version, the vectors $\boldsymbol{\ell}_t(\mathbf{w}_t, y_t)$ are sparse and have at most two nonzero coordinates. This is also true for our recalibration payoff game since $\mathbf{w}_t$ is a distribution over two consecutive probabilities values. Hence, the Gradient Descent Step requires only $O(1)$ computation and the Projection Step can be performed $O(1)$ (for $p = 1$). Since $\boldsymbol{\theta}$ is the only state the OGD algorithm needs to store, the storage space required is $O(\min\{T, m\})$.

Algorithm 1 Online Recalibration Algorithm

```
Input: some natural number $m \geq \sqrt{4L_s}$
Initialize: $\boldsymbol{\theta}_1 = \mathbf{0}, \mathbf{w}_1 \in \Delta_{m+1}$
for $t = 1, \ldots, T$ do
    Observe $q_t$ from black-box prediction oracle
    Sample $i_t \sim \mathbf{w}_t$, predict $p_t = \frac{i_t}{m}$, observe $y_t$
    Set $l_t := -\boldsymbol{\ell}_t(w_t, y_t)$
    Query learning algorithm: $\boldsymbol{\theta}_{t+1} \leftarrow \text{Update}(\boldsymbol{\theta}_t | l_t)$    // Online Gradient Descent step
    Query halfspace oracle: $\mathbf{w}_{t+1} \leftarrow \text{Approach}(\boldsymbol{\theta}_{t+1})$  // Obtain $\mathbf{w}_{t+1} \in \Delta_{m+1}$ from $\boldsymbol{\theta}_{t+1}$
end for
```

OGD guarantees that the regret is no more than $DG\sqrt{T}$ where D is the ℓ_2 diameter of the set and G is the ℓ_2-norm of the largest cost vector. For the convex set $\mathcal{K}_\infty^m$, the ℓ_2 diameter is $O(\sqrt{m})$. The ℓ_2-norm of the calibration component of the vector is bounded by $\sqrt{2}$. To make the size of the regret at time t small and at most 1, we normalize by the lipschitz-constant L_s

$$
\begin{aligned}
C_{p,T}^{\varepsilon,\delta}(\mathcal{A}) &\leq \mathtt{dist}_p^1 \left(\frac{1}{T} \sum_{t=1}^{T} \boldsymbol{\ell}_t(\mathbf{w}_t, y_t), \mathcal{S}_p^m \right) && (16)\\
&\leq \frac{\text{Regret}_t}{T} && (17)\\
&\leq \frac{GD}{\sqrt{T}} && (18)\\
&= O\left(\sqrt{\frac{m}{T}} \right) \quad (\text{for } p = 1) && (19)
\end{aligned}
$$

Theorem 12 (ℓ_p generalization of Theorem 10)**.** *For any m, there exists a $(\ell_p, \frac{1}{m}, \frac{4L_s}{m^2})$-online recalibration algorithm that guarantees a recalibration rate of $O\left(\frac{m^c}{\sqrt{T}}\right)$ where $c = \left(\frac{1}{2} - \frac{1}{q}\right)$ for $q > 2$ and $c = 2$ for $q \leq 2$*

We prove this result in the appendix similar to how we show the ℓ_1 version of the theorem. The convex set $\mathcal{K}_\infty^m$ is replaced with a more general $\mathcal{K}_q^m$ for q such that $\frac{1}{p} + \frac{1}{q} = 1$. The results from Lemma 11 and 14 hold for any p. The halfspace oracle algorithm also applies for any p. We obtain our bounds using Online Gradient Descent. However, the learning algorithm can be chosen as Online Mirror Descent for a regularizer optimized for the corresponding convex set $\mathcal{K}_q^m$.

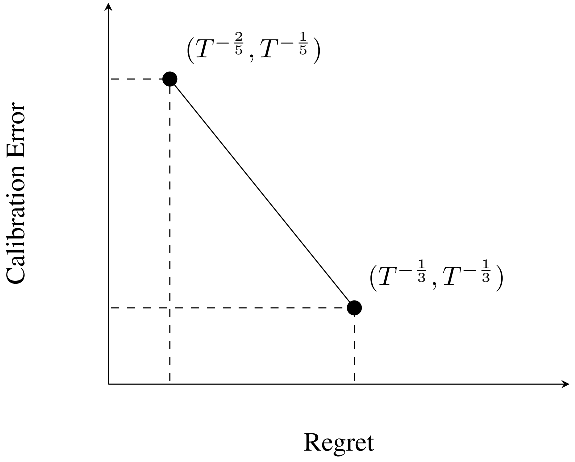
5 Convergence Rates

In this section, we describe how the results from the previous sections can be used to obtain bounds on calibration error and regret.

Theorem 13. *For any $x \in [\frac{1}{3}, \frac{2}{5}]$, given a black-box prediction oracle, there exists a forecasting algorithm that simultaneously achieves regret $O(T^{-x})$ while keeping ℓ_1-calibration error less than T^{2x-1}.*

Proof. In Theorem 10, we show that for any m, there exists an $(\ell_1, \frac{1}{m}, \frac{4L_s}{m^2})$-online recalibration algorithm which satisfies a recalibration rate of $O\left(\sqrt{\frac{m}{T}}\right)$. By definition 4, this implies that the ℓ_1-calibration error is upper bounded by $O(\frac{1}{m} + \sqrt{\frac{m}{T}})$ and the regret is upper bounded by $O(\frac{1}{m^2} + \sqrt{\frac{m}{T}})$. Setting $m = T^{1-2x}$, we obtain an algorithm that guarantees regret of $O(T^{-x})$ and calibration error $O(T^{2x-1})$ □

Figure 1: The graph below captures the linear tradeoff between regret and ℓ_1 calibration error.



Theorem 13 generalizes to ℓ_p calibration as follows.

Theorem 14 (ℓ_p generalization of Theorem 13)**.** *Given a black-box prediction oracle, there exist forecasting algorithms that satisfy the following guarantees. For any $p \geq 2$, the algorithm simultaneously achieves regret and ℓ_p-calibration error less than $O(T^{1/2})$. For any $1 \leq p < 2$ and $x \in [\frac{p}{p+2}, \frac{2p}{3p+2}]$, there exists a forecasting algorithm that simultaneously achieves regret $O(T^{-x})$ while keeping ℓ_p-calibration error less than $O(T^{(2x-1)/(2/p-1)})$.*

The proof of the theorem parallels the proof of Theorem 13, substituting $m = \sqrt{T}$ in case $p \geq 2$, and $m = T^{(1-2x)/(2/p-1)}$ in case $1 \leq p < 2$.

References

Jacob Abernethy, Peter L. Bartlett, and Elad Hazan. Blackwell approachability and no-regret learning are equivalent. In Sham M. Kakade and Ulrike von Luxburg, editors, *Proceedings of the 24th Annual Conference on Learning Theory*, volume 19 of *Proceedings of Machine Learning Research*, pages 27–46, Budapest, Hungary, 09–11 Jun 2011. PMLR. URL https://proceedings.mlr.press/v19/abernethy11b.html.

David Blackwell. An analog of the minimax theorem for vector payoffs. *Pacific Journal of Mathematics*, 6(1):1 – 8, 1956. doi: pjm/1103044235. URL https://doi.org/.

Tolga Bolukbasi, Kai-Wei Chang, James Zou, Venkatesh Saligrama, and Adam Kalai. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, page 4356–4364, Red Hook, NY, USA, 2016. Curran Associates Inc. ISBN 9781510838819.

Joy Buolamwini and Timnit Gebru. Gender shades: Intersectional accuracy disparities in commercial gender classification. In Sorelle A. Friedler and Christo Wilson, editors, *Proceedings of the 1st Conference on Fairness, Accountability and Transparency*, volume 81 of *Proceedings of Machine Learning Research*, pages 77–91. PMLR, 23–24 Feb 2018. URL https://proceedings.mlr.press/v81/buolamwini18a.html.

Nicolo Cesa-Bianchi and Gábor Lugosi. *Prediction, learning, and games*. Cambridge university press, 2006.

Dean P Foster. A proof of calibration via blackwell's approachability theorem. *Games and Economic Behavior*, 29(1):73–78, 1999. ISSN 0899-8256. doi: https://doi.org/10.1006/game.1999.0719. URL `https://www.sciencedirect.com/science/article/pii/S0899825699907194`.

Dean P Foster and Sergiu Hart. Calibeating: Beating forecasters at their own game, 2021. URL `preprint:http://www.ma.huji.ac.il/hart/papers/calib-beat.pdf?`

Tilmann Gneiting and Adrian E Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102(477):359–378, 2007. doi: 10.1198/016214506000001437. URL `https://doi.org/10.1198/016214506000001437`.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 1321–1330. PMLR, 06–11 Aug 2017. URL `https://proceedings.mlr.press/v70/guo17a.html`.

Ursula Hebert-Johnson, Michael Kim, Omer Reingold, and Guy Rothblum. Multicalibration: Calibration for the (Computationally-identifiable) masses. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 1939–1948. PMLR, 10–15 Jul 2018. URL `https://proceedings.mlr.press/v80/hebert-johnson18a.html`.

Jon Kleinberg. Inherent trade-offs in algorithmic fairness. In *Abstracts of the 2018 ACM International Conference on Measurement and Modeling of Computer Systems*, SIGMETRICS '18, page 40, New York, NY, USA, 2018. Association for Computing Machinery. ISBN 9781450358460. doi: 10.1145/3219617.3219634. URL `https://doi.org/10.1145/3219617.3219634`.

Volodymyr Kuleshov and Stefano Ermon. Estimating uncertainty online against an adversary. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, AAAI'17, page 2110–2116. AAAI Press, 2017.

Shie Mannor and Gilles Stoltz. A geometric proof of calibration. *Math. Oper. Res.*, 35(4):721–727, nov 2010. ISSN 0364-765X. doi: 10.1287/moor.1100.0465. URL `https://doi.org/10.1287/moor.1100.0465`.

Alexandru Niculescu-Mizil and Rich Caruana. Predicting good probabilities with supervised learning. In *Proceedings of the 22nd International Conference on Machine Learning*, ICML '05, page 625–632, New York, NY, USA, 2005. Association for Computing Machinery. ISBN 1595931805. doi: 10.1145/1102351.1102430. URL `https://doi.org/10.1145/1102351.1102430`.

John C. Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In *ADVANCES IN LARGE MARGIN CLASSIFIERS*, pages 61–74. MIT Press, 1999.

Martin Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In *Proceedings of the Twentieth International Conference on International Conference on Machine Learning*, ICML'03, page 928–935. AAAI Press, 2003. ISBN 1577351894.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes] See Section 1
   (b) Did you describe the limitations of your work? [Yes] See Section 1
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] See Section 1

(d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes] See Section 1

2. If you are including theoretical results...

   (a) Did you state the full set of assumptions of all theoretical results? [Yes] See Section **??**

   (b) Did you include complete proofs of all theoretical results? [Yes] See Sections 4, 5

3. If you ran experiments...

   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [N/A]

   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [N/A]

   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [N/A]

   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [N/A]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...

   (a) If your work uses existing assets, did you cite the creators? [N/A]

   (b) Did you mention the license of the assets? [N/A]

   (c) Did you include any new assets either in the supplemental material or as a URL? [N/A]

   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]

   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]

5. If you used crowdsourcing or conducted research with human subjects...

   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]

   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]

   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

A Appendix

A.1 Proof of Lemma 9

Lemma 15 (Restating Lemma 9)**.** *For a fixed (a, b) such that $\|a\|_\infty = 1$ and $b \geq 0$, the halfspace H_1 parametrized by (a, b) defined by below is approachable*

$$H_1 := \left\{(x, z) \mid \langle a, x\rangle + bz \leq \frac{1}{m} + \frac{4bL_S}{m^2}\right\} \tag{20}$$

Proof. To show that H_1 is approachable, we will find a mixed distribution for the forecaster (i.e, a probability distribution over $p \in \{0/m, 1/m, 2/m, \dots, (m-1)/m, 1\}$) such that $\mathbb{E}_p\left[\langle a, \boldsymbol{\ell}_c(p, y)\rangle + b\boldsymbol{\ell}_r(p, y)\right] \leq \frac{1}{m} + \frac{4bL_S}{m^2}$ for any $y \in \{0, 1\}$. For simplicity, define

$$c(i, y) = \frac{i}{m} - y \quad \text{and} \quad C_i = \begin{bmatrix} c(i, 0) \\ c(i, 1) \end{bmatrix}$$

$$d(i, y) = b \cdot S\left(\frac{i}{m}, y\right) - b \cdot S(q_t, y) \quad \text{and} \quad D_i = \begin{bmatrix} d(i, 0) \\ d(i, 1) \end{bmatrix}$$

$$f(i, y) = a_i c(i, y) + d(i, y) \quad \text{and} \quad F_i = \begin{bmatrix} f(i, 0) \\ f(i, 1) \end{bmatrix} = a_i C_i + D_i$$

Observe that $f(i, y) = \langle a, \boldsymbol{\ell}_c(\frac{i}{m}, y)\rangle + b\boldsymbol{\ell}_r(\frac{i}{m}, y)$, so our objective becomes to show that there exists a distribution p over $\frac{i}{m} \in \{0, \dots, m\}$ such that $\mathbb{E}_p\, f(i, y) \leq \frac{1}{m} + \frac{4bL_S}{m^2}$ for $y \in \{0, 1\}$, or equivalently that the vector $\mathbb{E}_p\, F_i$ belongs to the quadrant-shaped set $(-\infty, \frac{1}{m} + \frac{4bL_S}{m^2}] \times (-\infty, \frac{1}{m} + \frac{4bL_S}{m^2}]$. We will be choosing p to be either a point-mass on $\frac{i}{m}$ for some i, or a distribution on two consecutive values in the set $\{0, \frac{1}{m}, \frac{2}{m}, \dots, 1\}$. Hence, the vector $\mathbb{E}_p\, F_i$ will belong to one of m closed line segments forming a polygonal path through the vectors $F_0, F_1, \dots, F_m$. Observe that F_0 belongs to the left half-plane, i.e. $f(0, 0) \leq 0$, because

$$f(0, 0) = a_0 c(0, 0) + d(0, 0) = b(S(0, 0) - S(q_t, 0)) \leq 0,$$

where the last inequality holds because $b \geq 0$ and S is a proper scoring rule. Similarly, F_m belongs to the lower half-plane, i.e. $f(m, 1) \leq 0$, because

$$f(m, 1) = a_m c(m, 1) + d(m, 1) = b(S(1, 1) - S(q_t, 1)) \leq 0.$$

If F_0 or, respectively, F_m belongs to the third quadrant — that is, the set $(-\infty, 0] \times (-\infty, 0]$ — then we choose p to be a point-mass on 0 or 1, respectively. The remaining case is that F_0 and F_m belong to the sets $(-\infty, 0] \times (0, \infty)$ and $(0, \infty) \times (-\infty, 0]$, respectively. In that case, F_0 and F_m lie on opposite sides of the line L consisting of all points $\begin{bmatrix} x_0 \\ x_1 \end{bmatrix}$ that satisfy $x_0 = x_1$; F_0 lies above L while F_m lies below it. Hence, there must be at least one index j such that F_j lies on or above L while F_{j+1} lies below it. We aim to construct a distribution p supported on $\{\frac{j}{m}, \frac{j+1}{m}\}$ such that $\mathbb{E}_p\, F_i$ belongs to the set $(-\infty, \frac{1}{m} + \frac{4bL_S}{m^2}] \times (-\infty, \frac{1}{m} + \frac{4bL_S}{m^2}]$. Assume without loss of generality that $j \geq m/2$. (The case $j \leq m/2$ is handled symmetrically, by exchanging the roles of the labels $y = 0$ and $y = 1$, i.e. the first and second coordinates of the vectors we are considering.)

For $p \in [0, 1]$ let $D(p)$ denote the vector

$$D(p) = b \cdot \begin{bmatrix} S(p, 0) - S(q_t, 0) \\ S(p, 1) - S(q_t, 1) \end{bmatrix}$$

and observe that the notation D_i defined earlier is equivalent to $D(i/m)$. The fact that S is a proper scoring rule ensures that when y is a random sample from $\{0, 1\}$ taking the value 1 with some probability p, the value of p' that minimizes $\mathbb{E}_y[S(p', y) - S(q_t, y)]$ is $p' = p$. Since the expected value $\mathbb{E}_y[S(p', y) - S(q_t, y)]$ is calculated by taking the inner product of the vector $D(p')$ with the probability vector

$$Y(p) = \begin{bmatrix} 1 - p \\ p \end{bmatrix},$$

this means that the curve $\mathcal{D} = \{D(p') \mid 0 \le p' \le 1\}$ is convex and that the line

$$L(p) = \{x \mid \langle Y(p), x\rangle = \langle Y(p), D(p)\rangle\}$$

is tangent to $\mathcal{D}$ at the point $D(p)$. The normal vector to this tangent line is $Y(p)$, so the vector $C(p) = \begin{bmatrix} p \\ p-1 \end{bmatrix}$, being orthogonal to $Y(p)$, is parallel to the tangent line at $D(p)$. When $p = i/m$, observe that the vector $C(p)$ defined here coincides with C_i defined earlier.

Summarizing the foregoing discussion, the line $L(j/m) = \{D_j + \lambda C_j \mid \lambda \in \mathbb{R}\}$ is tangent to the convex curve $\mathcal{D}$ at the point D_j, hence it lies (weakly) below that curve. In particular, recall the line L consisting of points whose first and second coordinates are equal, and consider the point I_j where L intersects $L(j/m)$. Since $L(j/m)$ lies (weakly) below $\mathcal{D}$ and $\mathcal{D}$ intersects L at $D(q_t) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$, the intersection of $L(j/m)$ with L must belong to the third quadrant. From these properties, it will follow that the line segment joining F_j to F_{j+1} intersects the set $(-\infty, \frac{1}{m} + \frac{4bL_S}{m^2}] \times (-\infty, \frac{1}{m} + \frac{4bL_S}{m^2}]$ as required.

Let E_j be the intersection point of $L(j/m)$ with a vertical line through D_{j+1}. Since $L(j/m)$ lies below $\mathcal{D}$, we know that E_j is situated directly below D_{j+1}. To reason about the distance between D_{j+1} and E_j, observe that the convexity of the curve $\mathcal{D}$ implies that the slope of the line segment joining D_j to D_{j+1} lies between the slopes of the tangent lines at D_j and D_{j+1}. Those slopes are $1 - m/j$ and $1 - m/(j+1)$, respectively. Hence, a pair of lines passing through D_j, with slopes $1 - m/j$ and $1 - m/(j+1)$, will intersect the vertical line through D_{j+1} in a line segment that contains D_{j+1}. The lower endpoint of that line segment is E_j. Its length is the difference between the slopes of the two lines, times the horizontal displacement between D_j and D_{j+1}. In other words, the length of the vertical line segment is

$$\frac{m}{j(j+1)} \cdot b \cdot \left[S\left(\frac{j}{m}, 0\right) - S\left(\frac{j+1}{m}, 0\right) \right] \;\le\; \frac{m}{(m/2)^2} \cdot \frac{bL_S}{m} = \frac{4bL_S}{m^2}.$$

Since the vertical line segment contains E_j and D_{j+1}, its length is an upper bound on their distance from one another.

Now define

$$G_j = E_j + a_{j+1}C_j = F_{j+1} + (E_j - D_{j+1}) + a_{j+1}(C_j - C_{j+1}).$$

Since E_j lies on $L(j/m)$ and C_j is parallel to $L(j/m)$, we know that G_j lies on $L(j/m)$. To determine the position of G_j relative to L, observe that $C_j - C_{j+1} = \begin{bmatrix} -1/m \\ -1/m \end{bmatrix}$ is parallel to L, $F_{j+1} + (E_j - D_{j+1})$ lies below F_{j+1}, and recall that F_{j+1} lies below L. Hence, G_j lies below L. As F_j lies on or above L it follows that the line segment joining F_j to G_j intersects L, and this intersection point must be I_j because the segment connecting F_j to G_j is contained in $L(j/m)$. Write $I_j = (1-t)F_j + tG_j$ for some parameter $t \in [0, 1]$.

If p is the distribution that selects a random $\frac{i}{m} \in \{\frac{j}{m}, \frac{j+1}{m}\}$ by setting $\frac{i}{m} = \frac{j}{m}$ with probability $1 - t$ and $\frac{i}{m} = \frac{j+1}{m}$ with probability t, then

$$\begin{aligned} \mathbb{E}_p\, F_i = (1-t)F_j + tF_{j+1} &= (1-t)F_j + tG_j + t(F_{j+1} - G_j) \\ &= I_j + t(D_{j+1} - E_j) + ta_{j+1}(C_{j+1} - C_j) \\ &= I_j + t(D_{j+1} - E_j) + \tfrac{ta_{j+1}}{m} \begin{bmatrix} 1 \\ 1 \end{bmatrix}. \end{aligned} \tag{21}$$

We need to show that both coordinates of the vector in Equation (21) are less than or equal to $\frac{1}{m} + \frac{4bL_S}{m}$. The first coordinate of I_j is non-positive, the first coordinate of $D_{j+1} - E_j$ is zero, and the first coordinate of $\frac{ta_{j+1}}{m} \begin{bmatrix} 1 \\ 1 \end{bmatrix}$ is at most $\frac{1}{m}$ since $0 \le t \le 1$ and $|a_{j+1}| \le 1$. The second coordinate of I_j is non-positive, the second coordinate of $D_{j+1} - E_j$ is at most $\frac{4bL_S}{m^2}$, and the second coordinate of $\frac{ta_{j+1}}{m} \begin{bmatrix} 1 \\ 1 \end{bmatrix}$ is at most $\frac{1}{m}$. □

A.2 Constructing the Halfspace Oracle

Here we go into more detail about how to construct the oracle asserted in 4. Recall that in the proof of Lemma A.1, given a halfspace θ parameterized by (a, b), we defined the vector F_i as follows:

$$F_i = \begin{bmatrix} f(i,0) \\ f(i,1) \end{bmatrix} \quad \text{where} \quad f(i,y) = a_i \left(\frac{i}{m} - y \right) + b \left[S \left(\frac{i}{m}, y \right) - S(q_t, y) \right] \tag{22}$$

In the proof, we note that F_0 is either in the 2nd or 3rd quadrant. Similarly, F_m is either in the 3rd or 4th quadrant. Thus, we first check if F_0 or F_m is in the 3rd quadrant. If one of them is, then we output a point distribution at the corresponding probability value. If none of F_0 or F_m is in the 3rd quadrant, then we binary search for an index i with F_i in the 3rd quadrant or a pair of consecutive indices $j, j+1$ where F_j is in 2nd quadrant and F_{j+1} is in the 4th quadrant. In the first case, $w_t(i) = 1$ and 0 everywhere else. In the second case, we set

$$w_t(j) = \frac{f(j+1,1) - f(j+1,0)}{f(j,0) - f(j+1,0) - f(j,1) + f(j+1,1)} \tag{23}$$

$$w_t(j+1) = \frac{f(j,0) - f(j,1)}{f(j,0) - f(j+1,0) - f(j,1) + f(j+1,1)} \tag{24}$$

and 0 everywhere else.The correctness of this procedure follows from the proof of LemmaA.1. The formula is obtained by solving this system of equations below to obtain a convex combination of F_j and F_{j+1}:

$$\begin{bmatrix} 1 & 1 \end{bmatrix} \begin{bmatrix} w_t(j) \\ w_t(j+1) \end{bmatrix} = 1 \quad \text{and} \quad \begin{bmatrix} 1 & -1 \end{bmatrix} \begin{bmatrix} f(j,0) & f(j+1,0) \\ f(j,1) & f(j+1,1) \end{bmatrix} \begin{bmatrix} w_t(j) \\ w_t(j+1) \end{bmatrix} = 0$$

Note that F_i does not need to be pre-computed for every index. It can be computed online during the binary search steps. Thus, this halfspace oracle can be implemented in $O(\log m)$ steps.

A.3 Learning Algorithm for general ℓ_p

Here we describe the learning for general ℓ_p. Similar to that of the learning algorithm for $p = 2$, we use Online Gradient Descent algorithm from [Zinkevich, 2003]. The ℓ_2-diameter for the convex set $\mathcal{K}_q^m$. The ℓ_2 diameter of $\mathcal{K}_q^m$ for $q \leq 2$ (i.e $p \geq 2$) is upper bounded by $\sqrt{m}$. For $q > 2$ (i.e $p < 2$), the ℓ_2 diameter m^c where $c = \frac{1}{2} - \frac{1}{q}$. Since the vectors $\boldsymbol{\ell}_t(\mathbf{w}_t, y_t)$ are sparse and have at most two nonzero coordinates, a similar argument to the case where $p = 1$, allows us to show that the Gradient Descent Step and the Projection Step can be performed $O(1)$.

$$C_{p,T}^{\varepsilon,\delta}(\mathcal{A}) \leq \texttt{dist}_p^1 \left(\frac{1}{T} \sum_{t=1}^{T} \boldsymbol{\ell}_t(\mathbf{w}_t, y_t), \mathcal{S}_p^m \right) \tag{25}$$

$$\leq \frac{\text{Regret}_t}{T} \tag{26}$$

$$\leq \frac{GD}{\sqrt{T}} = O\left(\frac{m^c}{\sqrt{T}} \right) \tag{27}$$

where $c = \frac{1}{2} - \frac{1}{q}$ for $q > 2$ (i.e $p < 2$) and $c = \frac{1}{2}$ for $q \leq 2$ (i.e $p \geq 2$)

A.4 Proof of Lemma 11

Lemma 16 (Restating Lemma 11). *For any vector* $\mathbf{x} \in \mathbb{R}^{m+2}$ *such that* $\|\mathbf{x}_{1:m+1}\|_p \geq 1/m$, *and* $\mathbf{x}_{m+2} \geq \frac{L_s}{m^2}$,

$$\texttt{dist}_p^1 \left(\mathbf{x}, \mathcal{S}_p^m \right) = -\frac{1}{m} - \frac{4L_s}{m^2} - \min_{\theta \in K_q^m} \langle -\mathbf{x}, \theta \rangle \tag{28}$$

where q *is such that* $\frac{1}{p} + \frac{1}{q} = 1$

Proof.

$$\texttt{dist}_p^1\left(\mathbf{x}, \mathcal{S}_p^m\right) = \texttt{dist}_p\left(\mathbf{x}_{1:m+1}, B_p^{m+1}(1/m)\right) + \texttt{dist}_1\left(\mathbf{x}_{m+2}, \left(-\infty, \frac{4L_s}{m^2}\right]\right) \tag{29}$$

$$= -\frac{1}{m} - \min_{\theta:||\theta||_q\leq 1}\langle -\mathbf{x}_{1:m+1}, \theta\rangle - \frac{4L_s}{m^2} - \min_{\theta\in[0,1]}\langle -\mathbf{x}_{m+2}, \theta\rangle \tag{30}$$

$$= -\frac{1}{m} - \frac{4L_s}{m^2} - \min_{\theta\in K_q^m}\langle -\mathbf{x}, \theta\rangle \tag{31}$$

$$\tag{32}$$

Remark: We need $\|\mathbf{x}_{1:m+1}\|_p \geq 1/m$, and $|\mathbf{x}_{m+2}| \geq \frac{L_s}{m^2}$ mainly for technicality in order to ensure equality. If these didn't hold, just like in the proof of Approachability, if you're already in the set you wish to approach, you can just make an arbitrary move. Similarly, if $\|\mathbf{x}_{1:m+1}\|_p < 1/m$ (i.e calibration error is already less than $\frac{1}{m}$), the algorithm can just follow the oracle's predictions. On the other hand, if $\mathbf{x}_{m+2} < \frac{L_s}{m^2}$, then following the halfspace oracle still ensures expected calibration error of at most $\frac{1}{m}$ for the timestep. □

Lemma 17 (Restating Lemma 6)**.**

$$C_{p,T}^{\varepsilon,\delta}(\mathcal{A}) = \texttt{dist}_p^\infty\left(\mathbf{v}_T, \mathcal{S}_p^{\lfloor\varepsilon^{-1}\rfloor}(\varepsilon,\delta)\right) \tag{33}$$

Proof.

$$\texttt{dist}_p^\infty\left(\mathbf{v}_T, \mathcal{S}_p^{\lfloor\varepsilon^{-1}\rfloor}(\varepsilon,\delta)\right) = \max\left\{\texttt{dist}_p\left(\mathbf{c}_T, B_p^{\lfloor\varepsilon^{-1}\rfloor}(\varepsilon/2)\right), \texttt{dist}_1\left(R_T, \left(-\infty, \frac{\delta}{2}\right]\right)\right\} \tag{34}$$

$$= \max\left\{C_{p,T}^{\varepsilon}(\mathcal{A}), R_T(\mathcal{A}) - \frac{\delta}{2}\right\} \tag{35}$$

$$= C_{p,T}^{\varepsilon,\delta}(\mathcal{A}) \tag{36}$$

□