LOCALLY ADAPTIVE MULTI-OBJECTIVE LEARNING

Jivat Neet Kaur*, **Isaac Gibbs***, **Michael I. Jordan***†
*University of California, Berkeley †Inria, Paris

ABSTRACT

We consider the problem of learning a predictor that satisfies multiple objectives of interest simultaneously, a general framework that captures a range of problem formulations including calibration, regret, and multiaccuracy. We work in an online setting where the data distribution can change arbitrarily over time. Existing approaches to this problem aim to minimize the set of objectives over the *entire time horizon* in a worst-case sense, and in practice they do not necessarily adapt to distribution shifts. Earlier work has aimed to alleviate this problem by incorporating additional objectives that target local guarantees over contiguous subintervals. Empirical evaluation of these proposals is, however, scarce. In this article, we consider an alternative procedure that achieves local adaptivity by replacing one part of the multi-objective learning method with an adaptive online algorithm. Empirical evaluations on datasets from energy forecasting and algorithmic fairness show that our proposed method improves upon existing approaches and achieves unbiased predictions over subgroups, while remaining robust under distribution shift.

1 INTRODUCTION

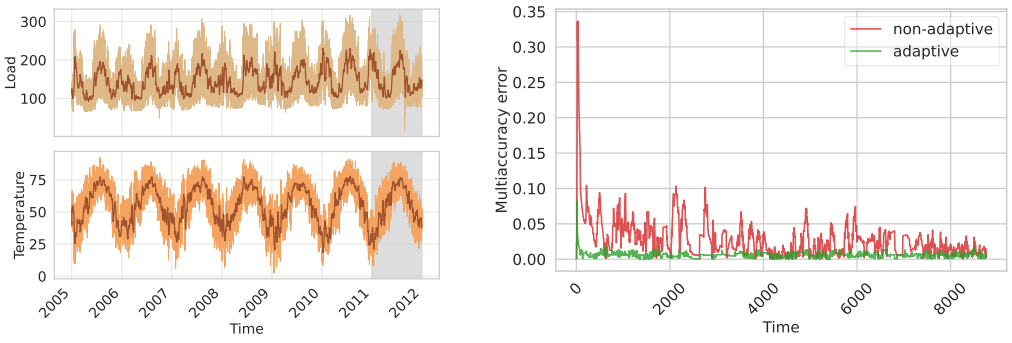
In an ever-changing world, real-time decision making necessitates coping with arbitrary distribution shifts and adversarial behavior. These shifts can arise from seasonality, change in the data distribution induced by feedback loops or policy changes, and exogenous shocks such as pandemics or economic crises. Online learning is a powerful framework for analyzing sequential data that makes no assumptions on the data distribution.

Multi-objective learning is a generic framework that refers to any task in which a predictor must satisfy multiple objectives or criterion of interest simultaneously (Lee et al., 2022). In the online setting, this encompasses many previously studied problems such as multicalibration (Hebert-Johnson et al., 2018), multivalid conformal prediction (Gupta et al., 2022), and multi-group learning (Deng et al., 2024). Despite being a desirable and promising notion, methods from the online multi-objective learning literature have had little influence on the practice of machine learning.

We attribute this to two shortcomings. First, many of the algorithms proposed in the literature are not adaptive to abrupt changes in the data distribution: they learn a predictor that minimizes the objectives over the *entire time horizon*. In changing environments and in the presence of adversarial behavior, such algorithms will fail to cope with distribution shifts. Second, most prior work is purely theoretical with scant empirical evaluation. As a result, the practical aspects of multi-objective online algorithms have received limited consideration.

In this work, we aim to overcome the above shortcomings. We propose a locally adaptive multi-objective learning algorithm that outputs predictors which (approximately) satisfy a set of objectives over all local time intervals $I \subseteq [T]$. Previously, Lee et al. (2022) suggested a method that lends adaptivity to existing algorithms by including additional objectives for all contiguous subintervals. We present an alternative approach that directly modifies the multi-objective algorithm by replacing one part of the scheme with an adaptive online learning method. We provide a meta-algorithm that, given an adaptive online learner, minimizes the worst case multi-objective loss across time intervals. For concreteness, we instantiate it with the Fixed Share method (Herbster & Warmuth, 1998), which is guaranteed to provide adaptivity over all intervals of a fixed target width. Other possible instantiations of our approach that target alternative adaptive guarantees are discussed in Section 2.2.

To close the empirical gap in this literature, we provide extensive empirical evaluations comparing the performance of various adaptive methods in practice. This includes experiments on electricity demand



(a) (b)

Figure 1: **GEFCom14-L electric load forecasting dataset.** On the left hand side are the time series for the raw load (light brown) and temperature (light orange) data. The dark brown curves indicate the weekly (168-hourly) moving average. The shaded grey region shows the competition duration. On the right-hand side, we plot a weekly moving average of the local multiaccuracy error.

forecasting and predicting recidivism over time in which our goal is to remove biases presenting in existing baseline predictors. Across all our empirical benchmarks we find that our proposed method consistently outperforms the previous proposals of Lee et al. (2022). We will release a codebase that implements our algorithm and all the baselines used in the paper.

As we discussed above, multi-objective learning can be used to address many common prediction tasks. As a case study, in this work, we focus on the multiaccuracy problem in which the goal is to learn predictors which are simultaneously unbiased under a set of covariate shifts of interest. We seek a small multiaccuracy error while preserving accuracy relative to a given sequence of baseline predictions. This is a problem of significant and broad interest across real-time decision-making and deployed machine learning systems. We show that our proposed algorithm has low multiaccuracy error over all intervals while the baselines have poor adaptivity. An alternative objective to multiaccuracy that is popular in the literature is multicalibration (Haghtalab et al., 2023a; Garg et al., 2024). Despite being a stronger condition, we show that in practice existing online multicalibration algorithms only achieve multiaccuracy at relatively slow rates. Adaptive extensions of the multicalibration algorithm yield improvements in local multiaccuracy error, however are unable to close the performance gap.

We note that although we focus on multiaccuracy in this paper, our general algorithm extends to other multi-objective learning problems including multi-group learning (Tosh & Hsu, 2022) and omniprediction (Gopalan et al., 2022). We discuss these extensions in Appendix D.

1.1 Peek at results

To demonstrate the significance of local adaptivity in practice, we consider the probabilistic electricity load forecasting track of the Global Energy Forecasting Competition 2014 (GEFCom2014) (Hong et al., 2016). The aim in the load forecasting track GEFCom2014-L is to forecast month-ahead quantiles of hourly loads for a US utility from January 1, 2011 to December 31, 2011 based on historical load and temperature data (Figure 1a). We consider the binary task of predicting whether the electricity demand exceeds 150MW at hour t and evaluate whether the predictions are multiaccurate with respect to discrete temperature groups $\{[0, 20), [20, 40), \ldots, [80, 100)\}$ (in °F). Informally, obtaining multiaccuracy with respect to temperature ensures our predictions are accurate at different times of day and across seasons. Figure 1b shows the multiaccuracy error of our proposed locally adaptive algorithm compared to a non-adaptive multiaccuracy algorithm, plotted as a weekly (168-hourly) moving average. We can see that the multiaccuracy error of the adaptive algorithm is close to zero across all time intervals, while the non-adaptive variant has high variance.

1.2 Preliminaries

We use $\mathcal{X}$ to denote the feature space and $\mathcal{Y} = [a, b]$ to denote the label space, which we assume to be a bounded interval. Our goal is to learn a sequence of predictors $p_t \in \mathcal{Y}, t = 1, 2, \ldots, T$ that guarantee loss minimization simultaneously for every objective within a set $\mathcal{L}$ over time. Each

objective, or criterion, is a function $\ell : \mathcal{Y} \times \mathcal{X} \times \mathcal{Y} \to [-1, 1]$ that takes as input a predictor $p_t(x_t)$, features $x_t \in \mathcal{X}$, and label $y_t \in \mathcal{Y}$ and returns a value in $[-1, 1]$. We will use $[T]$ to denote the set $\{1, 2, \ldots, T\}$. The sequence of data points (x_t, y_t), $t \in [T]$ can be generated adversarially dependent on the entire history of data and predictions up to time t.

The objectives we consider can be quite general and we will give some examples of specific choices shortly. Broadly, our only restriction is that the objectives should be consistent with one another in the sense that for any distribution on y_t there is a single optimal predictor $p_t(x_t)$ that minimizes all the objectives simultaneously. Formally, we assume the following.

Assumption 1. *For any $x \in \mathcal{X}$ and distribution P_Y on $\mathcal{Y}$ there exists $p^* \in \mathcal{Y}$ such that for all $\ell \in \mathcal{L}$,*

$$p^* \in \operatorname*{argmin}_{p \in \mathcal{Y}} \mathbb{E}_{Y \sim P_Y}[\ell(p, x, Y)].$$

Moreover, for all $\ell \in \mathcal{L}$, p^* guarantees the loss bound

$$\mathbb{E}_{Y \sim P_Y}[\ell(p^*, x, Y)] \leq 0. \tag{1}$$

The assumption that p^* produces a negative objective value is not strictly necessary and previous work in multiobjective learning has considered slightly more general settings (Lee et al., 2022). We have chosen to add this condition because it simplifies the notation and is satisfied by many common problems of interest. For instance, as we will discuss in the sections that follow, multiaccuracy, multicalibration, omniprediction, and multi-group learning can all be formulated in a way that meets this condition.

Using this assumption, our goal in online multi-objective learning will be to learn a sequence of predictors p_t that (approximately) matches the optimal bound (1):

$$\max_{\ell \in \mathcal{L}} \frac{1}{T} \sum_{t=1}^{T} \ell(p_t(x_t), x_t, y_t) \lessapprox 0.$$

As an example, we now define two instantiations of multi-objective problems that are commonly studied in the literature and which we will focus on—multiaccuracy and multicalibration. The offline version of multiaccuracy was introduced in Kim et al. (2019). We parameterize the multiaccuracy criterion by a function class $\mathcal{F}$ and the goal is to be unbiased for all $f \in \mathcal{F}$, i.e., there is no systematic correlation between the prediction residuals and any $f \in \mathcal{F}$.

Definition 1 (Online multiaccuracy)**.** Let $\mathcal{F} = \{f : \mathcal{X} \to [0, 1]\}$ be a class of functions on $\mathcal{X}$. In online multiaccuracy, we instantiate $\ell_{\text{MA}_{f,\sigma}}(p_t(x_t), x_t, y_t) = \sigma f(x_t) \cdot (y_t - p_t(x_t))$ for every sign $\sigma = \{\pm\}$ and $f \in \mathcal{F}$ and define the multiaccuracy error ℓ_{MA} in the sup-norm as

$$\ell_{\text{MA}}(p_t(x_t), x_t, y_t) = \sup_{f,\sigma} \frac{1}{T} \sum_{t=1}^{T} \sigma f(x_t) \cdot (y_t - p_t(x_t)). \tag{2}$$

Another popular online prediction formulation is multicalibration (Hebert-Johnson et al., 2018). In a binary classification task, calibration asks that among instances with predicted probability p, a fraction p of them are observed to be truly labeled as 1. Multicalibration is a strengthening of calibration that additionally requires the predictor to be multiaccurate conditional on its realized value. To implement this in practice, we discretize the label interval $[0, 1]$ into m bins $V_m := \{[0, 1/m), [1/m, 2/m), \ldots, [(m-1)/m, 1]\}$ and define a representative value for each bin as the midpoint $v_j = \frac{2j-1}{2m}$ for $j = 1, \ldots, m$. We then define an approximate notion of multicalibration that asks for v_j to be an unbiased prediction of $y_t = 1$ over all reweightings in $\mathcal{F}$ and all timepoints where $p_t \in [v_j - \frac{1}{2m}, v_j + \frac{1}{2m})$.

Definition 2 (Online multicalibration)**.** Fix a set of functions $\mathcal{F}$ and $m \geq 1$. In online multicalibration we instantiate $\ell_{\text{MC}_{f,\sigma,v}}(p_t(x_t), x_t, y_t) = \sigma f(x_t) \cdot \mathbb{1}\{p_t(x_t) \in v\} \cdot (y_t - v_j)$ for every sign $\sigma = \{\pm\}$, $f \in \mathcal{F}$, and $v \in V_m$ and define the multicalibration error ℓ_{MC} in the sup-norm as

$$\ell_{\text{MC}}(p_t(x_t), x_t, y_t) = \sup_{f,\sigma,v} \frac{1}{T} \sum_{t=1}^{T} \sigma f(x_t) \cdot \mathbb{1}\{p_t(x_t) \in v\} \cdot (y_t - v_j). \tag{3}$$

A direct calculation shows that the online multicalibration error always upper bounds the multiaccuracy error; specifically, $\ell_{\text{MA}} \leq m \cdot \ell_{\text{MC}}$.

In this work, we will give a multi-objective learning algorithm that achieves small multiaccuracy error while preserving accuracy relative to a base predictor sequence $\tilde{p}_t(x_t), t \in [T]$. While improving multiaccuracy, it is important that we do not degrade the accuracy of $\tilde{p}_t(x_t)$, leaving its predictions less useful. We discuss this in more detail in Section 4.1. We define the latter accuracy objective as prediction error. In what follows, we let $c : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}_{\geq 0}$ denote any proper loss for the mean, i.e., any loss such that $\mathbb{E}_{y\sim P}[y] \in \operatorname{argmin}_p \mathbb{E}_{y\sim P}[c(p, y)]$ for all distributions P on $\mathcal{Y}$. A common example that we will work with in our experiments is the squared error/Brier score $c(p, y) = (y - p)^2$.

Definition 3 (Online prediction error)**.** Given a base predictor sequence $\tilde{p}_t(x_t), t \in [T]$, define the prediction error ℓ_{pred} as

$$\ell_{\text{pred}}(p_t(x_t), x_t, y_t) := \frac{1}{T}\sum_{t=1}^{T} c(p_t(x_t), y_t) - c(\tilde{p}_t(x_t), y_t), \tag{4}$$

2 METHODS

2.1 ONLINE MULTI-OBJECTIVE LEARNING

The online multi-objective learning problem is a sequential prediction task over T rounds. A standard framework introduced in Lee et al. (2022) is to consider a two-player game between a learner, who observes $x_t \in \mathcal{X}$ and chooses a predictor $p_t(x_t)$, and an adversary who maintains a distribution $q^{(t)} \in \Delta(\mathcal{L})$, where we use the notation $\Delta(S)$ to denote the set of probability distributions over the set S. At each time step, the learner observes the adversary's current mixture and the covariates x_t and chooses its (randomized) prediction as $p_t(x_t) \sim P_t(x_t)$, where

$$P_t(x_t) = \operatorname*{argmin}_{P\in\Delta(\mathcal{Y})} \max_{y\in\mathcal{Y}} \mathbb{E}_{p\sim P}\left[\sum_{\ell} q_\ell^{(t)} \ell(p, x_t, y)\right].$$

This choice is designed to guarantee that the learner obtains the best possible performance under the adversarial value of y_t with respect to the mixture loss specified by $q^{(t)}$. As an aside, we note that although generic multiobjective learning problems require randomized predictors, our methods will often produce deterministic values. This is due to the fact that for many of the problems we are interested in (e.g., multiaccuracy, low predictive accuracy) the objectives are convex and thus the minimax program above admits a solution $P_t(x_t)$ that is supported on a singleton.

After the learner makes its selection, the true value of y_t is revealed and the adversary updates its mixture distribution. In the original work of Lee et al. (2022), the adversary sets its weights using the Hedge updates

$$q_\ell^{(t+1)} \propto q_\ell^{(t)} \exp(\eta \ell(p_t(x_t), x_t, y_t)),$$

for some $\eta = \Theta(\sqrt{\log(|\mathcal{L}|)/T})$. This is designed to ensure that the mixture distribution with respect to $q^{(t)}$ is a good proxy for the maximum multiobjective error. More formally, this choice of weights has the following well-known error bound (see, e.g., Theorem 1.5 of Hazan (2016)),

$$\max_{\ell\in\mathcal{L}} \sum_{t=1}^{T} \mathbb{E}_{p\sim P_t(x_t)}[\ell(p, x_t, y_t)] \leq \sum_{\ell\in\mathcal{L}}\sum_{t=1}^{T} q_\ell^{(t)} \mathbb{E}_{p\sim P_t(x_t)}[\ell(p, x_t, y_t)] + O(\sqrt{T\log(|\mathcal{L}|)}).$$

By combining this bound with the choice of $p_t(x_t)$ we obtain the following multiobjective error bound.

Theorem 1 (Theorem 2.1 in Lee et al. (2022))**.** *Under Assumption 1, Algorithm 1 with Hedge as the method for learning $q^{(t)}$ obtains the multiobjective learning bound*

$$\max_{\ell\in\mathcal{L}} \sum_{t=1}^{T} \mathbb{E}_{p\sim P_t(x_t)}[\ell(p, x_t, y_t)] \leq O(\sqrt{T\log(|\mathcal{L}|)}).$$

2.2 LOCALLY ADAPTIVE MULTI-OBJECTIVE LEARNING

The result of Theorem 1 ceases to be useful when environments are changing and the data distribution shifts arbitrarily over time. As a simple example, fix the singleton function class $\mathcal{F}_{\text{MA}} = \{x \mapsto 1\}$

and consider targeting just the multiaccuracy error (i.e., set $\mathcal{L} = \{\ell_{\mathrm{MA}_{f,\sigma}} : f \in \mathcal{F}_{\mathrm{MA}}, \sigma \in \{\pm\}\}$). Let the labels be given as $y_t = 1$ for the first $T/2$ rounds and $y_t = 0$ for the last $T/2$ rounds. Here, the constant predictor $p_t = 1/2$ minimizes the multiaccuracy error in (2). Nevertheless, this predictor performs poorly in the individual intervals $1 \leq t \leq T/2$ and $t > T/2$ compared to the optimal predictor that switches from $p_t = 1$ to $p_t = 0$ after $t = T/2$.

To account for distribution shifts in changing environments, we will now modify the method of Lee et al. (2022) by replacing the Hedge algorithm with a locally adaptive method. Informally, this will allow us to bound the worst case multi-objective loss over local subintervals given by

$$\sup_{I=[r,s]} \left[\max_{\ell \in \mathcal{L}} \sum_{t=r}^{s} \mathbb{E}_{p \sim P_t(x_t)}[\ell(p, x_t, y_t)] \right], \tag{5}$$

where the supremum is over some appropriate set of intervals I that we will specify shorty. Algorithm 1 gives our generic method. Here, WL denotes any procedure for learning the weights $q^{(t)}$. From here on, we use the shorthand $\ell^{(t)} := \mathbb{E}_{p \sim P_t(x_t)}[\ell(p, x_t, y_t)]$ to denote the expected loss of our randomized predictor at time step t.

Algorithm 1 Locally adaptive multi-objective learning

Input: Set of objectives $\mathcal{L}$, learning method WL
Input: Sequence of samples $\{(x_1, y_1), \ldots, (x_T, y_T)\}$
$q_\ell^{(1)} = \frac{1}{|\mathcal{L}|}, \quad \forall \ell \in \mathcal{L}.$
for each $t \in [T]$ **do**
  $P_t(x_t) = \operatorname*{argmin}_{P \in \Delta(\mathcal{Y})} \max_{y \in \mathcal{Y}} \mathbb{E}_{p \sim P} \left[\sum_{\ell \in \mathcal{L}} q_\ell^{(t)} \ell(p, x_t, y_t) \right]$
  Output $p_t(x_t) \sim P_t(x_t)$
  $q_\ell^{(t+1)} = \mathtt{WL}(\{q^{(s)}\}_{s \leq t}, \{\mathbb{E}_{p \sim P_t(x_t)}[\ell(p, x_t, y_t)]\}_{\ell \in \mathcal{L}})$

As a concrete instantiation, we will perform empirical experiments on the Fixed Share method introduced in Herbster & Warmuth (1998) that modifies the Hedge update by adding an exploration term that prevents any of the weights from collapsing to zero. A formal statement of this procedure is given in Algorithm 2. As we will discuss in the next section, Fixed Share provides a multiobjective learning guarantee *locally* on any interval of a fixed width. There are many possible alternative methods that one could implement in the place of Fixed Share. For instance, one may consider the *strongly adaptive* learning procedure of Daniely et al. (2015) and Jun et al. (2017) that guarantee a stronger notion of adaptive regret with dependency over the interval width $|I|$ for all intervals $I \subseteq [T]$. We have chosen to focus on Fixed Share due to its strong empirical performance.

Comparison to adaptive algorithms in literature. Previously, Lee et al. (2022) proposed an adaptive extension of their multi-objective learning algorithm that included additional objectives for all subintervals. Formally, given an initial set of objectives $\mathcal{L}$ they consider the augmented collection $\mathcal{L}_{\text{adapt.}} = \{\ell(p_t(x_t), x_t, y_t)1\{t \in I\} \mid \ell \in \mathcal{L}, I = [r, s] \subseteq [T]\}$ and show that using these objectives in the algorithm described in Section 2.1 guarantees the local bound

$$\sup_{I=[r,s] \subseteq [T]} \left[\max_{\ell \in \mathcal{L}} \sum_{t \in [I]} \ell^{(t)} \right] \leq O(\sqrt{T(\log(|\mathcal{L}|) + 2\log T)},$$

where the supremum is over all contiguous intervals $I \subseteq [T]$. In our work, we propose to hold the set of objectives fixed and instead use a locally adaptive procedure WL to learn the weights $q^{(t)}$.

3 THEORY

We will now state a theoretical guarantee for Algorithm 1. For concreteness, we will focus on the case where the adversary learns the weights $q^{(t)}$ using the Fixed Share method given in Algorithm 2. Similar results for other adaptive learning methods can be obtained in an identical fashion by replacing the regret bound for Fixed Share (Lemma 1) with the associated bound for that method.

The theory has two parts: a guarantee for the adversary's distribution q and a guarantee on the learner's response. We denote the $|\mathcal{L}|$-dimensional vector of losses as $\ell_{\mathcal{L}}^{(t)} = (\ell^{(t)})_{\ell \in \mathcal{L}}$. All proofs are deferred to Appendix B. We first show that the maximum objective value over any time interval I is upper bounded by the average value of the individual objectives taken with respect to weights $q^{(t)}$.

Lemma 1. *Consider Algorithm 1 with weights learned using Algorithm 2. Assume that $\gamma \leq 1/2$. Then, for any interval $I = [r, s] \subseteq [T]$,*

$$\sum_{t=r}^{s} q^{(t)\top} \ell_{\mathcal{L}}^{(t)} \geq \max_{\ell \in \mathcal{L}} \sum_{t=r}^{s} \ell^{(t)} - \eta \left(\sum_{t=r}^{s} q^{(t)\top} (\ell_{\mathcal{L}}^{(t)})^2 \right) - \frac{1}{\eta} \left(\log \left(\frac{|\mathcal{L}|}{\gamma} \right) + |I| 2\gamma \right). \tag{6}$$

Next, we show that the average value of the objectives is non-positive over any interval I. This lemma follows from the minimax-optimal strategy of the learner and has been shown to hold previously in Lee et al. (2022).

Lemma 2. *Suppose the objectives satisfy Assumption 1. Then, for any interval $I = [r, s] \subseteq [T]$,*

$$\sum_{t=r}^{s} q^{(t)\top} \ell_{\mathcal{L}}^{(t)} \leq 0.$$

We combine the previous two lemmas to get our main result.

Theorem 2. *Let $\gamma \leq 1/2$ and assume that the objectives satisfy Assumption 1. Then, for any $I = [r, s] \subseteq [T]$,*

$$\max_{\ell \in \mathcal{L}} \frac{1}{|I|} \sum_{t=r}^{s} \ell^{(t)} \leq \frac{\eta}{|I|} \left(\sum_{t=r}^{s} q^{(t)\top} (\ell_{\mathcal{L}}^{(t)})^2 \right) + \frac{1}{\eta |I|} \left(\log \left(\frac{|\mathcal{L}|}{\gamma} \right) + |I| 2\gamma \right). \tag{7}$$

The guarantee of Theorem 2 depends on the values of the fixed share hyperparameters γ, η. To set the best upper bound for a given interval I, we would ideally substitute the optimal values $\gamma = \frac{1}{2|I|}$ and $\eta = \sqrt{\frac{\log(|\mathcal{L}| \cdot 2|I|) + 1}{\sum_{t=r}^{s} q^{(t)\top} (\ell_{\mathcal{L}}^{(t)})^2}}$ in (7) and obtain

$$\max_{\ell \in \mathcal{L}} \frac{1}{|I|} \sum_{t=r}^{s} \ell^{(t)} \leq \frac{2}{|I|} \sqrt{\left(\log(|\mathcal{L}| \cdot 2|I|) + 1 \right)} \cdot \sqrt{\sum_{t=r}^{s} q^{(t)\top} (\ell_{\mathcal{L}}^{(t)})^2} = O\left(\sqrt{\frac{\log(|\mathcal{L}| \cdot |I|)}{|I|}} \right).$$

In practice, we can only use one setting of these parameters and cannot specialize γ and η to a specific interval. To mimic these optimal choices, we let the user pick a fixed target interval width $|I| = \tau$, noting that a smaller choice of τ gives stronger locally adaptive guarantees at the cost of a looser upper bound. Since the optimal value for η used above depends on the expected squared objective $\sum_{t=r}^{s} q^{(t)\top} (\ell_{\mathcal{L}}^{(t)})^2$ which is unknown in practice, we follow Gibbs & Candès (2024) in selecting an adaptive value of η that updates online as $\eta = \eta_t := \sqrt{\frac{\log(|\mathcal{L}| \cdot 2\tau) + 1}{\sum_{s=t-\tau+1}^{t} q^{(s)\top} (\ell_{\mathcal{L}}^{(s)})^2}}$. This lets the algorithm adaptively track changes in the moving average of the expected squared objective over the most recent τ time steps. We also demonstrate the importance of this choice empirically in Appendix G.1.

4 Applications to Mean Estimation and Quantile Estimation

As a case study, we focus on the multiaccuracy problem in this work. We consider two example applications of Algorithm 1 to multiaccurate mean and quantile estimation. We defer the discussion on quantile estimation to Appendix C.3. Our goal in multiaccurate mean estimation is to learn predictors that have small multiaccuracy error (2) while guaranteeing the prediction error (4) is low relative to a given sequence of baseline predictions. We fix a function class $\mathcal{F}_{\text{MA}} \subseteq \{f : \mathcal{X} \to [0, 1]\}$ that we desire multiaccuracy with respect to and define $\mathcal{L} := \{\ell_{\text{MA}_{f,\sigma}} : f \in \mathcal{F}_{\text{MA}}, \sigma \in \{\pm\}\} \cup \{\ell_{\text{pred}}\}$ including the prediction error objective. We provide an algorithm for locally adaptive multiaccurate mean estimation in Alg. 3 and its guarantee in Corollary C.1. The weights $q_{\text{MA},f,\sigma}^{(t)}$ and $q_{\text{pred}}^{(t)}$ denote the entries of $q^{(t)}$ associated with the multiaccuracy and prediction error objectives, respectively. Next, we discuss the importance of including the prediction error objective in multiaccuracy problems.

4.1 SIGNIFICANCE OF THE PREDICTION ERROR OBJECTIVE

In our applications, we will start with a base forecaster, $\tilde{p}_t(x_t)$ that was constructed in advance for that application. Our goal will be to improve $\tilde{p}_t(x_t)$ to be multiaccurate. While doing this, it is important that we do not degrade the accuracy of $\tilde{p}_t(x_t)$, thereby rendering its predictions less useful. Our algorithm achieves small multiaccuracy error while preserving the accuracy relative to a base predictor by including an additional prediction error objective (4). If such a base forecaster is not available, one might consider omitting the prediction error objective and running our method with just the multiaccuracy objectives $\mathcal{L} = \{\ell_{\text{MA}_{f,\sigma}} : f \in \mathcal{F}_{\text{MA}}, \sigma \in \{\pm\}\}$. In general, this is not advisable. Indeed, if we exclude the regret objective in Algorithm 3 one can show that the best response of the learner yields the predictor: $p_t(x_t) = b\mathbb{1}\big\{\sum_{f,\sigma} q^{(t)}_{\text{MA}_{f,\sigma}} \sigma f(x_t) > 0\big\} + a\mathbb{1}\big\{\sum_{f,\sigma} q^{(t)}_{\text{MA}_{f,\sigma}} \sigma f(x_t) \leq 0\big\}$. This solution has a pathological behavior where the predictor will only take the extreme values a or b at every step. This makes the predictions less useful and interpretable for real-time decision-making in an online setting. Our prediction error objective recovers the predictor from this problem by enforcing solutions that do not lie in the extremes. In practical settings where $\tilde{p}_t(x_t)$ is not available in advance, we recommend combining our procedure with a standard online learning algorithm (e.g., online gradient or mirror descent) that provides an appropriate baseline (see, e.g., Algorithm 4).

5 EXPERIMENTS

In this section, we present a set of empirical evaluations on real applications. In each example, we define a baseline predictor sequence $\tilde{p}_t(x_t)$ and a set of objectives we evaluate. We learn locally adaptive predictors using the general recipe in Alg. 2 and compare with baseline approaches we define in Section 5.2. In Section 5.1, we specify for each dataset a practically and societally meaningful set of covariates that define the function class $\mathcal{F}$. We consider simulated examples in Appendix H and compare the local adaptivity of all methods under different magnitudes of distribution shift.

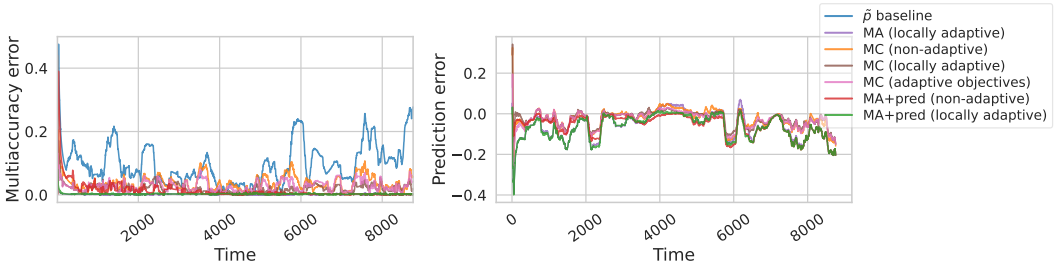
5.1 DATASETS

GEFCom2014 electric load forecasting. In Section 1.1, we introduced the binary load prediction task and displayed the load and temperature trends over time (Figure 1a). We set function class $\mathcal{F}$ to be the indicator functions for the temperature groups $\{[0, 20), [20, 40), \ldots, [80, 100)\}$.We construct our baseline predictions $\tilde{p}_t$ by linearly interpolating the quantiles forecasts of Ziel & Liu (2016), whose method outperforms the top entries in the competition. See Appendix E.2 for further details.

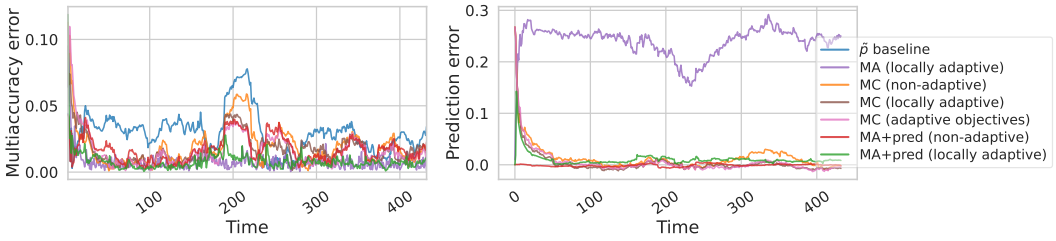
COMPAS dataset. Larson et al. (2016) analyzed the COMPAS tool used to predict recidivism for criminal defendants in Broward County, Florida and found that certain groups of defendants are more likely to be incorrectly judged as high risk of recidivism. In Figure 3, we plot the true recidivism rate over time for different racial groups. We consider the recidivism prediction task and evaluate the local multiaccuracy of predictors with respect to the African-American, Caucasian, and Hispanic subgroups that constitute over 90% of the dataset. We use the COMPAS recidivism risk scores in the dataset as our baseline predictions. The scores take integer values between 1–10 and we rescale to $[0, 1]$. Following the analysis of Barenstein (2019) who point the data processing error in the two-year sample cutoff rule for recidivists, we drop the data points with screen date after April 1, 2014.

5.2 BASELINES

We consider baselines that differ in their *adaptivity* and the *set of objectives in* $\mathcal{L}$. **MA+pred** denotes the algorithm with the multiaccuracy and prediction error objectives $\mathcal{L} := \{\ell_{\text{MA}_{f,\sigma}} : f \in \mathcal{F}_{\text{MA}}, \sigma \in \{\pm\}\} \cup \{\ell_{\text{pred}}\}$. We now explain the baselines. **Baseline predictors** $\tilde{p}_t$ are predictions that were constructed in advance for the application and are our input to Algorithm 3. **Multiaccuracy (MA)** with $\mathcal{L} := \{\ell_{\text{MA}_{f,\sigma}} : f \in \mathcal{F}_{\text{MA}}, \sigma \in \{\pm\}\}$ is a specific case of Algorithm 3 where $\mathcal{L}$ does not include ℓ_{pred}. We implement the online **Multicalibration (MC)** algorithm from Lee et al. (2022). This is a competitive algorithm as multicalibration is a stronger condition than multiaccuracy. Lee et al. (2022) show that their algorithm can guarantee that predictions satisfy an accuracy objective (specifically, low squared error) on subgroups in addition to multicalibration. Hence, we consider c as the squared error in our prediction error objective ℓ_{pred}. We take number of bins $m = 10$ as it is a reasonable



(a) GEFCom2014-L



(b) COMPAS

Figure 2: **Local multiaccuracy error (left) and prediction error (right)**, (a) GEFCom2014-L and (b) COMPAS. We skip the initial time steps (10 for (a), 2 for (b)) to improve readability. For clarity, comparison with MA+pred (adaptive objectives) is in Appendix F.2. Ablations are in Appendix G.

target for multicalibration. Lower values will give better multiaccuracy results at the cost of a much weaker multicalibration guarantee. We evaluate for varying m in Appendix E.3.

We consider three variants for the algorithms: **non-adaptive, locally adaptive**, and **adaptive objectives**. The non-adaptive variant corresponds to using Hedge to learn the weights in Algorithm 1; the locally adaptive variant corresponds to using the Fixed Share update as stated in Algorithm 2; and the adaptive objectives variant corresponds to using Hedge with additional objectives for all subintervals. Specifically, the adaptive objectives method augments the objectives as $\mathcal{L}_{\text{adapt.}} = \{\ell(p_t(x_t), x_t, y_t)\mathbb{1}\{t \in I\}, \ell \in \mathcal{L}, I = [r, s] \subseteq [T]\}$.

5.3 Local multiaccuracy and prediction error evaluation

In this section, we evaluate the local multiaccuracy (ℓ_{MA}) and prediction error (ℓ_{pred}) incurred by the algorithms. First, we consider results on GEFCom2014-L (Figure 2a) using an interval width $\tau = 336$ hours (2 weeks). While the baseline predictor $\tilde{p}_t$ exhibits high ℓ_{MA}, all algorithms improve upon it; specifically, locally adaptive algorithms MA and MA+pred achieve near-zero ℓ_{MA} over all local intervals, whereas non-adaptive variants show high local variability. Notably, all MC variants have significantly slower multiaccuracy rates in practice. Observing the prediction error (right panel), the MA baseline has non-zero ℓ_{pred}, whereas MA+pred consistently preserves or improves accuracy over $\tilde{p}_t$. MC generally yields negative prediction error, though with poorer adaptivity than MA+pred.

On COMPAS (Figure 2b, $\tau = 50$ days), non-adaptive methods again show minimal adaptivity to underlying shifts, performing poorly on subgroups locally. Conversely, our proposed algorithm achieves significantly better local multiaccuracy. While adaptive MC variants improve over non-adaptive MC, their multiaccuracy rates remain substantially worse than MA+pred (locally adaptive). Although MA (locally adaptive) yields slightly better multiaccuracy than MA+pred, it suffers significantly higher prediction error (2b (right)). We evaluate ℓ_{MA} across a wider range of widths $|I|$ in Appendix F.1.

6 Discussion

We hope our work serves as an initial step toward bridging the empirical gap in this growing literature. Our evaluation focuses on a subset of objectives and validation on broader problems is interesting future work. Further, empirical comparisons with other adaptive procedures for learning weights could help determine whether local errors can be further reduced in practice.

ACKNOWLEDGMENTS

We thank Nika Haghtalab, Eric Zhao, and Paula Gradu for helpful discussions. We thank Florian Ziel for sharing the GEFCom2014-L quantile forecasts from their paper (Ziel & Liu, 2016). This work was supported in part by the Office of Naval Research under grant number N00014-20-1-2787 and by the European Union (ERC-2022-SYG-OCEAN-101071601). Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Research Council Executive Agency. Neither the European Union nor the granting authority can be held responsible for them.

REFERENCES

Jacob Abernethy, Peter L. Bartlett, and Elad Hazan. Blackwell approachability and low-regret learning are equivalent. In *Proceedings of the Annual Conference on Learning Theory*, 2011.

Matias Barenstein. Propublica's compas data revisited. *arXiv preprint arXiv:1906.04711*, 2019.

David Blackwell. An analog of the minimax theorem for vector payoffs. *Pacific Journal of Mathematics*, 6(1):1 – 8, 1956.

Amit Daniely, Alon Gonen, and Shai Shalev-Shwartz. Strongly adaptive online learning. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 1405–1411. PMLR, 2015.

Samuel Deng, Jingwen Liu, and Daniel Hsu. Group-wise oracle-efficient algorithms for online multi-group learning. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

Sumegha Garg, Christopher Jung, Omer Reingold, and Aaron Roth. Oracle efficient online multicalibration and omniprediction. In *Proceedings of the 2024 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pp. 2725–2792, 2024.

Isaac Gibbs and Emmanuel J. Candès. Conformal inference for online prediction with arbitrary distribution shifts. *Journal of Machine Learning Research*, 25(162):1–36, 2024.

Ira Globus-Harris, Declan Harrison, Michael Kearns, Aaron Roth, and Jessica Sorrell. Multicalibration as boosting for regression. In *Proceedings of the 40th International Conference on Machine Learning*, Proceedings of Machine Learning Research, pp. 11459–11492, 2023.

Parikshit Gopalan, Adam Tauman Kalai, Omer Reingold, Vatsal Sharan, and Udi Wieder. Omnipredictors. In *13th Innovations in Theoretical Computer Science Conference (ITCS 2022)*, volume 215 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pp. 79:1–79:21, 2022.

Paula Gradu, Elad Hazan, and Edgar Minasyan. Adaptive regret for control of time-varying dynamics. In *Proceedings of The 5th Annual Learning for Dynamics and Control Conference*, volume 211, pp. 560–572. PMLR, 15–16 Jun 2023.

Varun Gupta, Christopher Jung, Georgy Noarov, Mallesh M. Pai, and Aaron Roth. Online Multivalid Learning: Means, Moments, and Prediction Intervals. In *13th Innovations in Theoretical Computer Science Conference (ITCS 2022)*, pp. 82:1–82:24, 2022.

Nika Haghtalab, Michael Jordan, and Eric Zhao. A unifying perspective on multi-calibration: Game dynamics for multi-objective learning. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023a.

Nika Haghtalab, Chara Podimata, and Kunhe Yang. Calibrated stackelberg games: Learning optimal commitments against calibrated agents. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023b.

Elad Hazan. *Introduction to Online Optimization*. Cambridge University Press, 2016.

Ursula Hebert-Johnson, Michael Kim, Omer Reingold, and Guy Rothblum. Multicalibration: Calibration for the (Computationally-identifiable) masses. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1939–1948, 2018.

Mark Herbster and Manfred K. Warmuth. Tracking the best expert. *Machine Learning*, 32(2): 151–178, 1998.

Tao Hong, Pierre Pinson, Shu Fan, Hamidreza Zareipour, Alberto Troccoli, and Rob J. Hyndman. Probabilistic energy forecasting: Global energy forecasting competition 2014 and beyond. *International Journal of Forecasting*, 32(3):896–913, 2016.

Kwang-Sung Jun, Francesco Orabona, Stephen Wright, and Rebecca Willett. Improved Strongly Adaptive Online Learning using Coin Betting. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54 of *Proceedings of Machine Learning Research*, pp. 943–951. PMLR, 2017.

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

Jeff Larson, Surya Mattu, Lauren Kirchner, and Julia Angwin. How we analyzed the compas recidivism algorithm. *ProPublica*, 2016. URL `https://www.propublica.org/article/how-we-analyzed-the-compas-recidivism-algorithm`.

Daniel Lee, Georgy Noarov, Mallesh Pai, and Aaron Roth. Online minimax multiobjective optimization: Multicalibeating and other applications. In *Advances in Neural Information Processing Systems*, 2022.

Georgy Noarov, Ramya Ramalingam, Aaron Roth, and Stephan Xie. High-dimensional prediction for sequential decision making. In *Forty-second International Conference on Machine Learning*, 2025.

Princewill Okoroafor, Robert Kleinberg, and Michael P. Kim. Near-optimal algorithms for omniprediction. *arXiv preprint arXiv:2501.17205*, 2025.

Christopher J Tosh and Daniel Hsu. Simple and near-optimal algorithms for hidden stratification and multi-group learning. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 21633–21657. PMLR, 2022.

Florian Ziel and Bidong Liu. Lasso estimation for GEFCom2014 probabilistic electric load forecasting. *International Journal of Forecasting*, 32(3):1029–1037, 2016.

A Related work

Our work is most closely related to the literature on multi-objective learning that encompasses numerous problems including multicalibration (Hebert-Johnson et al., 2018), multiaccuracy (Kim et al., 2019), multi-group learning (Tosh & Hsu, 2022), and omniprediction (Gopalan et al., 2022). Each of these multi-objective criteria have been studied in both the online and batch settings. Most closely related to our work, Kim et al. (2019) and Globus-Harris et al. (2023) give algorithms for obtaining multi-accurate and multi-calibrated (respectively) predictors in the batch setting that are guaranteed to have accuracy no worse than that of a given base predictor.

In the online adversarial setting, a number of works develop algorithms for obtaining multiaccuracy, multicalibration, and/or omniprediction globally over all time steps (Lee et al., 2022; Garg et al., 2024; Okoroafor et al., 2025; Haghtalab et al., 2023a; Noarov et al., 2025). Our work will in particular build on the algorithmic framework developed in Lee et al. (2022). This methodology has deep roots in the online learning literature and builds on ideas arising from Blackwell approachability (Blackwell, 1956) and its connection to no-regret learning (Abernethy et al., 2011).

To obtain time-local guarantees we will draw on the literature on adaptive regret (Herbster & Warmuth, 1998; Daniely et al., 2015; Jun et al., 2017; Haghtalab et al., 2023b). Our work will most closely rely upon the work of Gradu et al. (2023) to obtain multi-objective error bounds over any local time interval. In the context of multi-objective learning, local guarantees have been discussed previously in Lee et al. (2022). However, the literature contains no empirical evaluations of these methods. We provide experiments evaluating the algorithms of Lee et al. (2022) in Section 5 and find that our approach achieves significantly lower error rates in practice.

B Proofs

B.1 Proof of Lemma 1

We follow the calculations of Gradu et al. (2023) and Gibbs & Candès (2024). Note that while in those earlier papers the losses are nonnegative, in our work the losses may take on negative values.

Let $W^{(t+1)} := \sum_\ell w_\ell^{(t)} \exp\left(\eta \cdot \ell(p_t(x_t, x_t, y_t))\right)$. We initialize $w_\ell^{(t)} = 1$ for all $\ell \in \mathcal{L}$. By construction, the probabilities are defined as follows: $q_\ell^{(t)} := \frac{w_\ell^{(t)}}{\sum_\ell w_\ell^{(t)}}$. Thus,

$$\frac{W^{(t+1)}}{W^{(t)}} = \sum_{\ell \in \mathcal{L}} q_\ell^{(t)} \exp\left(\eta \cdot \ell(p_t(x_t, x_t, y_t))\right).$$

Since η is small and ℓ is bounded between $[-1, 1]$, $|\eta \cdot \ell(p_t(x_t, x_t, y_t)| \leq 1$. We use the inequalities $1 - a \leq \exp(-a)$ and for $|x| \leq 1, \exp(a) \leq 1 + a + a^2$ to get

$$\frac{W^{(t+1)}}{W^{(t)}} \leq \exp\left(\eta q^{(t)\top} \ell^{(t)} + \eta^2 q^{(t)\top} {\ell_{\mathcal{L}}^{(t)}}^2\right).$$

Inductively, this implies that, for interval $I = [r, s]$,

$$\frac{W^{(s+1)}}{W^{(r)}} \leq \exp\left(\sum_{t=r}^{s} \eta q^{(t)\top} \ell^{(t)} + \eta^2 q^{(t)\top} {\ell_{\mathcal{L}}^{(t)}}^2\right).$$

On the other hand, for any fixed $\ell \in \mathcal{L}$, $w_\ell^{(t+1)} \geq w_\ell^{(t)}(1-\gamma)\exp\left(\eta \ell^{(t)}\right)$. Without loss of generality, we proceed with a fixed ℓ, noting that the same calculations will follow for all $\ell \in \mathcal{L}$. This gives

$$\begin{aligned}
\frac{W^{(s+1)}}{W^{(r)}} \geq \frac{w_\ell^{(s+1)}}{W^{(r)}} &\geq (1-\gamma)^{|I|} q_\ell^{(t)} \exp\left(\sum_{t=r}^{s} \eta \ell^{(t)}\right) \\
&\geq (1-\gamma)^{|I|} \frac{\gamma}{|\mathcal{L}|} \exp\left(\sum_{t=r}^{s} \eta \ell^{(t)}\right).
\end{aligned}$$

Combining the two inequalities and taking logarithm on both sides yields

$$|I|(1-\gamma) + \log\left(\frac{\gamma}{|\mathcal{L}|}\right) + \sum_{t=r}^{s} \eta \ell^{(t)} \leq \sum_{t=r}^{s} \eta q^{(t)\top} \ell_{\mathcal{L}}^{(t)} + \eta^2 q^{(t)\top} {\ell_{\mathcal{L}}^{(t)}}^2.$$

We rearrange to get the following inequality

$$\sum_{t=r}^{s} q^{(t)\top} \ell_{\mathcal{L}}^{(t)} \geq \sum_{t=r}^{s} \ell^{(t)} - \eta\left(\sum_{t=r}^{s} q^{(t)\top} {\ell_{\mathcal{L}}^{(t)}}^2\right) + \frac{1}{\eta}|I|(1-\gamma) + \log\left(\frac{\gamma}{|\mathcal{L}|}\right).$$

As $\gamma \leq 1/2$, we can use the inequality $\log(1-\gamma) \geq -2\gamma$ to get the final inequality

$$\sum_{t=r}^{s} q^{(t)\top} \ell_{\mathcal{L}}^{(t)} \geq \sum_{t=r}^{s} \ell^{(t)} - \eta\left(\sum_{t=r}^{s} q^{(t)\top} {\ell_{\mathcal{L}}^{(t)}}^2\right) - \frac{1}{\eta}\left(\log\left(\frac{|\mathcal{L}|}{\gamma}\right) + |I|2\gamma\right).$$

As the same calculation holds for any objective $\ell \in \mathcal{L}$, we get the final result

$$\sum_{t=r}^{s} q^{(t)\top} \ell_{\mathcal{L}}^{(t)} \geq \max_{\ell \in \mathcal{L}} \sum_{t=r}^{s} \ell^{(t)} - \eta\left(\sum_{t=r}^{s} q^{(t)\top} {\ell_{\mathcal{L}}^{(t)}}^2\right) - \frac{1}{\eta}\left(\log\left(\frac{|\mathcal{L}|}{\gamma}\right) + |I|2\gamma\right).$$

B.2 Proof of Lemma 2

This result was shown in Lee et al. (2022) and we include their argument here for completeness.

Let $u^{(t)}(p, y) := \sum_{\ell} q_{\ell}^{(t)} \ell(p, x_t, y)$. Let $\Delta(\mathcal{Y})$ denote the space of distributions over $\mathcal{Y}$. Applying Sion's Minimax Theorem, we get

$$\begin{aligned}\min_{P \in \Delta(\mathcal{Y})} \max_{y \in \mathcal{Y}} \mathbb{E}_{p \sim P}[u^{(t)}(p, y)] &= \min_{P \in \Delta(\mathcal{Y})} \max_{Q \in \Delta(\mathcal{Y})} \mathbb{E}_{p \sim P, y \sim Q}[u^{(t)}(p, y)] \\ &= \max_{Q \in \Delta(\mathcal{Y})} \min_{P \in \Delta(\mathcal{Y})} \mathbb{E}_{p \sim P, y \sim Q}[u^{(t)}(p, y)].\end{aligned}$$

This conveys that the minimax-optimal strategy p_t of the learner can achieve $u^{(t)}(p, y)$ as low as if the adversary moved first and the learner could best-respond. Now, for a fixed distribution Q on Y we have that by Assumption 1 there exists p^* such that $\mathbb{E}_{y \sim Q}[u^{(t)}(p^*, y)] \leq 0$.

Thus, the minimax optimal strategy guarantees that $\min_{P \in \Delta(\mathcal{Y})} \max_{y \in \mathcal{Y}} \mathbb{E}_{p \sim P}[u^{(t)}(p, y)] \leq 0$ for all $t \in [T]$. This yields the desired inequality

$$\sum_{t=r}^{s} q^{(t)\top} \ell_{\mathcal{L}}^{(t)} \leq 0.$$

B.3 Proof of Theorem 2

Applying Lemma 2 to the inequality (6) in Lemma 1 gives

$$\max_{\ell \in \mathcal{L}} \sum_{t=r}^{s} \ell^{(t)} - \eta\left(\sum_{t=r}^{s} q^{(t)\top} {\ell_{\mathcal{L}}^{(t)}}^2\right) - \frac{1}{\eta}\left(\log\left(\frac{|\mathcal{L}|}{\gamma}\right) + |I|2\gamma\right) \leq 0.$$

Rearranging and dividing both sides by $|I|$ yields the desired inequality,

$$\max_{\ell \in \mathcal{L}} \frac{1}{|I|} \sum_{t=r}^{s} \ell^{(t)} \leq \frac{\eta}{|I|}\left(\sum_{t=r}^{s} q^{(t)\top} {\ell_{\mathcal{L}}^{(t)}}^2\right) + \frac{1}{\eta |I|}\left(\log\left(\frac{|\mathcal{L}|}{\gamma}\right) + |I|2\gamma\right).$$

B.4 Proof of Corollary C.1

The proof follows by instantiating the set of objectives $\mathcal{L}$ for multiaccurate mean estimation in Theorem 2. We take $\mathcal{L} := \{\ell_{\mathrm{MA}_{f,\sigma}} : f \in \mathcal{F}_{\mathrm{MA}}, \sigma \in \{\pm\}\} \cup \{\ell_{\mathrm{pred}}\}$, where $\mathcal{F}_{\mathrm{MA}} \subseteq \{f : \mathcal{X} \to [0, 1]\}$ is the function class that we desire multiaccuracy with respect to and ℓ_{pred} is the prediction error objective. Plugging the objectives in (3), this gives us the desired bound

$$\max\left\{\max_{f,\sigma} \frac{1}{|I|} \sum_{t=r}^{s} \ell_{\mathrm{MA}_{f,\sigma}}^{(t)}, \frac{1}{|I|} \sum_{t=r}^{s} \ell_{\mathrm{pred}}^{(t)}\right\} = O\left(\sqrt{\frac{\log(|\mathcal{L}| \cdot |I|)}{|I|}}\right).$$

B.5 Proof of Corollary C.2

The proof follows by instantiating the set of objectives $\mathcal{L}$ for multiaccurate quantile estimation in Theorem 2. We take $\mathcal{L} := \{\sigma f(x_t)(\mathbb{1}\{y_t \le \theta_t\} - \alpha) : f \in \mathcal{F}_{\text{MA}}, \sigma \in \{\pm\}\} \cup \{\ell_\alpha(\theta_t, y_t) - \ell_\alpha(\tilde{\theta}_t, y_t)\}$, where $\mathcal{F}_{\text{MA}} \subseteq \{f : \mathcal{X} \to [0,1]\}$ is the function class that we desire multiaccuracy with respect to. Plugging the objectives in (3), this gives us the desired bound

$$\max\left\{\max_{f,\sigma}\frac{1}{|I|}\sum_{t=r}^{s}\sigma f(x_t)(\mathbb{1}\{y_t \le \theta_t\}-\alpha), \frac{1}{|I|}\sum_{t=r}^{s}\ell_\alpha(\theta_t, y_t)-\ell_\alpha(\tilde{\theta}_t, y_t)\right\} = O\left(\sqrt{\frac{\log(|\mathcal{L}|\cdot|I|)}{|I|}}\right).$$

C Deferred Algorithms

C.1 Fixed Share

Algorithm 2 gives a formal statement of the Fixed Share procedure (Herbster & Warmuth, 1998).

Algorithm 2 Fixed-Share weight update

Input: Weights at current timestep $q^{(t)}$; hyperparameters η, γ.
Input: Losses for current timestep $\{\ell^{(t)}\}_{\ell\in\mathcal{L}}$
1: **for** each $t \in [T]$ **do**
2: $\quad \tilde{q}_\ell^{(t+1)} = \dfrac{q_\ell^{(t)} \exp\left(\eta \cdot \ell^{(t)}\right)}{\sum_{\ell'\in\mathcal{L}} q_{\ell'}^{(t)} \exp\left(\eta \cdot \ell'^{(t)}\right)}$, for all $\ell \in \mathcal{L}$
3: $\quad q_\ell^{(t+1)} = (1-\gamma)\,\tilde{q}_\ell^{(t+1)} + \frac{\gamma}{|\mathcal{L}|}$
Output: Weights for the next time step $q^{(t+1)}$

C.2 Multiaccurate Mean Estimation

We present the algorithm for locally adaptive multiaccurate mean estimation discussed in Section 4 in Algorithm 3. We use the shorthands $\ell^{(t)}_{\text{MA}_{f,\sigma}} := \sigma f(x_t)(y_t - p_t(x_t))$ and $\ell^{(t)}_{\text{pred}} := c(p_t(x_t), y_t) - c(\tilde{p}_t(x_t), y_t)$ to denote the realized losses. Assuming c is convex, we note that since the objectives $\ell_{\text{MA}_{f,\sigma}}$ and ℓ_{pred} are convex we may assume without loss of generality that the predictor $p_t(x_t)$ is deterministic. The weights $q^{(t)}_{\text{MA},f,\sigma}$ and $q^{(t)}_{\text{pred}}$ in Algorithm 3 are used to denote the entries of $q^{(t)}$ associated with the multiaccuracy and prediction error objectives, respectively.

Corollary C.1. *Consider the weights learned using Algorithm 3 with* $\eta = \sqrt{\frac{\log(|\mathcal{L}|\cdot 2|I|)+1}{\sum_{t=r}^{s} q^{(t)\top}(\ell_{\mathcal{L}}^{(t)})^2}}$. *Assume that* $\gamma \le 1/2$. *Then, for any interval* $I = [r, s] \subseteq [T]$,

$$\max\left\{\frac{\max_{f,\sigma}\sum_{t=r}^{s}\ell^{(t)}_{\text{MA}_{f,\sigma}}}{|I|}, \frac{\sum_{t=r}^{s}\ell^{(t)}_{\text{pred}}}{|I|}\right\} = O\Big(\sqrt{\tfrac{\log(|\mathcal{L}|\cdot|I|)}{|I|}}\Big).$$

In Section 4.1, we discussed the significance of the prediction error objective in preserving the accuracy relative to a base predictor sequence $\tilde{p}_t(x_t)$. When $\tilde{p}_t(x_t)$ is not available in advance, we can combine our procedure with a standard online learning algorithm (e.g., online gradient or mirror descent) that provides an appropriate baseline. Algorithm 4 gives a complete description of this approach. In what follows, the weights $q^{(t)}_{\text{MA},f,\sigma}$ and $q^{(t)}_{\text{pred}}$ are used to denote the entries of $q^{(t)}$ associated with the multiaccuracy and prediction error objectives, respectively.

C.3 Multiaccurate Quantile Estimation

Our algorithm can also be employed for quantile estimation. For a user-specified quantile level $\alpha \in (0,1)$, we seek to estimate quantile predictors $\theta_t(x_t)$ that minimize

$$\left|\frac{1}{T}\sum_{t=1}^{T}\mathbb{1}\{y_t \le \theta_t(x_t)\} - \alpha\right|.$$

Algorithm 3 Locally adaptive multiaccurate mean estimation

Input: Function class $\mathcal{F}_{\text{MA}} \subseteq \{f : \mathcal{X} \to [0,1]\}$; base predictor sequence $\tilde{p}_t(x_t), t \in [T]$; hyperparameters η, γ.
Input: Sequence of samples $\{(x_1, y_1), \ldots, (x_T, y_T)\}$

$q^{(1)}_{\text{MA}_{f,\sigma}} = \frac{1}{2|\mathcal{F}_{\text{MA}}|+1}, \quad \forall f \in \mathcal{F}_{\text{MA}}, \sigma \in \{\pm 1\}.$

$q^{(1)}_{\text{pred}} = \frac{1}{2|\mathcal{F}_{\text{MA}}|+1}$

for each $t \in [T]$ **do**

$$p_t(x_t) := \operatorname*{argmin}_p \max_{y \in \mathcal{Y}} \sum_{f,\sigma} q^{(t)}_{\text{MA}_{f,\sigma}} \sigma f(x_t)(y-p) + q^{(t)}_{\text{pred}}(c(p,y) - c(\tilde{p}_t(x_t), y))$$

$$\tilde{q}^{(t+1)}_{\text{MA}_{f,\sigma}} = \frac{q^{(t)}_{\text{MA}_{f,\sigma}} \exp\left(\eta \cdot \sigma f(x_t)(y_t - p_t(x_t))\right)}{\sum_{\ell' \in \mathcal{L}} q^{(t)}_{\ell'} \exp\left(\eta \cdot \ell'(p_t(x_t), x_t, y_t)\right)} \text{ for all } f \in \mathcal{F}_{\text{MA}}, \sigma \in \{\pm 1\}$$

$$\tilde{q}^{(t+1)}_{\text{pred}} = \frac{q^{(t)}_{\text{pred}} \exp\left(\eta \cdot (c(p_t(x_t), y_t) - c(\tilde{p}_t(x_t), y_t))\right)}{\sum_{\ell' \in \mathcal{L}} q^{(t)}_{\ell'} \exp\left(\eta \cdot \ell'(p_t(x_t), x_t, y_t)\right)}$$

$$q^{(t+1)}_{\text{MA}_{f,\sigma}} = (1-\gamma)\, \tilde{q}^{(t+1)}_{\text{MA}_{f,\sigma}} + \frac{\gamma}{2|\mathcal{F}_{\text{MA}}|+1}$$

$$q^{(t+1)}_{\text{pred}} = (1-\gamma)\, \tilde{q}^{(t+1)}_{\text{pred}} + \frac{\gamma}{2|\mathcal{F}_{\text{MA}}|+1}$$

Output: Sequence of (randomized) predictors $p_1, \ldots, p_T$

Algorithm 4 Locally adaptive multiaccurate mean estimation (learning $\tilde{p}_t$ online)

Input: Function class $\mathcal{F}_{\text{MA}} \subseteq \{f : \mathcal{X} \to [0,1]\}$; $\mathcal{F}_{\text{pred}} = \{f_\beta : \beta \in \mathbb{R}\}$; hyperparameters η, γ.
Input: Sequence of samples $\{(x_1, y_1), \ldots, (x_T, y_T)\}$

$q^{(1)}_{\text{MA}_{f,\sigma}} = \frac{1}{2|\mathcal{F}_{\text{MA}}|+1}, \quad \forall f \in \mathcal{F}_{\text{MA}}, \sigma \in \{\pm 1\}.$

$q^{(1)}_{\text{pred}} = \frac{1}{2|\mathcal{F}_{\text{MA}}|+1}$

$\beta_1 = 0$

for each $t \in [T]$ **do**

$\beta_{t+1} = \beta_t - \gamma \nabla_\beta \ell\left(f_{\beta_t}(x_t), y_t\right)$

$\tilde{p}_t(x_t) := f_{\beta_t}(x_t)$

$$p_t(x_t) := \operatorname*{argmin}_p \max_{y \in \mathcal{Y}} \sum_{f,\sigma} q^{(t)}_{\text{MA}_{f,\sigma}} \sigma f(x_t)(y-p) + q^{(t)}_{\text{pred}}(c(p,y) - c(\tilde{p}_t(x_t), y))$$

$$\tilde{q}^{(t+1)}_{\text{MA}_{f,\sigma}} = \frac{q^{(t)}_{\text{MA}_{f,\sigma}} \exp\left(\eta \cdot \sigma f(x_t)(y_t - p_t(x_t))\right)}{\sum_{\ell' \in \mathcal{L}} q^{(t)}_{\ell'} \exp\left(\eta \cdot \ell'(p_t(x_t), x_t, y_t)\right)} \text{ for all } f \in \mathcal{F}_{\text{MA}}, \sigma \in \{\pm 1\}$$

$$\tilde{q}^{(t+1)}_{\text{pred}} = \frac{q^{(t)}_{\text{pred}} \exp\left(\eta \cdot (c(p_t(x_t), y_t) - c(\tilde{p}_t(x_t), y_t))\right)}{\sum_{\ell' \in \mathcal{L}} q^{(t)}_{\ell'} \exp\left(\eta \cdot \ell'(p_t(x_t), x_t, y_t)\right)}$$

$$q^{(t+1)}_{\text{MA}_{f,\sigma}} = (1-\gamma)\, \tilde{q}^{(t+1)}_{\text{MA}_{f,\sigma}} + \frac{\gamma}{2|\mathcal{F}_{\text{MA}}|+1}$$

$$q^{(t+1)}_{\text{pred}} = (1-\gamma)\, \tilde{q}^{(t+1)}_{\text{pred}} + \frac{\gamma}{2|\mathcal{F}_{\text{MA}}|+1}$$

Output: Sequence of (randomized) predictors $p_1, \ldots, p_T$

We refer to this objective as *coverage* and its interpretation is that $\theta_t(x_t)$ lies above y_t with frequency α. It is well known that minimizing the quantile loss ℓ_α (also referred to as pinball loss) produces the desired quantile predictors. Given a sequence of baseline quantile predictors $\tilde{\theta}_t(x_t)$, our goal is to update the predictions to satisfy the multiaccuracy criterion specified by $\mathcal{F}_{\text{MA}}$ while preserving the quantile loss ℓ_α relative to $\tilde{\theta}_t(x_t)$. We define $\mathcal{L} := \{\sigma f(x_t)(\mathbb{1}\{y_t \le \theta_t(x_t)\} - \alpha) : f \in \mathcal{F}_{\text{MA}}, \sigma \in \{\pm\}\} \cup \{\ell_\alpha(\theta_t(x_t), y_t) - \ell_\alpha(\tilde{\theta}_t(x_t), y_t)\}$. We provide the explicit algorithm in Algorithm 5 and its guarantee in Corollary C.2. Note that we have to allow $\theta_t(x_t)$ to be random in this algorithm. The weights $q^{(t)}_{\text{MA},f,\sigma}$ and $q^{(t)}_{\text{pred}}$ in Algorithm 5 are used to denote the entries of $q^{(t)}$ associated with the multiaccuracy and quantile loss objectives, respectively.

Corollary C.2. *Consider the weights are learned using Algorithm 5 with* $\eta = \sqrt{\frac{\log(|\mathcal{L}|\cdot 2|I|)+1}{\sum_{t=r}^{s} q^{(t)\top}(\ell_{\mathcal{L}}^{(t)})^2}}$. *Assume that* $\gamma \le 1/2$. *Then, for any interval* $I = [r, s] \subseteq [T]$,

$$\max\left\{\max_{f,\sigma}\frac{1}{|I|}\sum_{t=r}^{s}\mathbb{E}_{\theta\sim\Theta_t(x_t)}[\sigma f(x_t)(\mathbb{1}\{y_t\le\theta\}-\alpha)],\right.$$
$$\left.\frac{1}{|I|}\sum_{t=r}^{s}\mathbb{E}_{\theta\sim\Theta_t(x_t)}[\ell_\alpha(\theta,y_t)-\ell_\alpha(\tilde{\theta}_t(x_t),y_t)]\right\} = O\left(\sqrt{\frac{\log(|\mathcal{L}|\cdot|I|)}{|I|}}\right).$$

Algorithm 5 Locally adaptive multiaccurate quantile estimation

Input: Function class $\mathcal{F}_{\text{MA}} \subseteq \{f : \mathcal{X} \to [0,1]\}$; quantile level α; baseline quantile predictors $\tilde{\theta}_t(x_t), t \in [T]$; hyperparameters η, γ.

Input: Sequence of samples $\{(x_1, y_1), \ldots, (x_T, y_T)\}$

$q^{(1)}_{\text{MA}_{f,\sigma}} = \frac{1}{2|\mathcal{F}_{\text{MA}}|+1}, \quad \forall f \in \mathcal{F}_{\text{MA}}, \sigma \in \{\pm 1\}.$

$q^{(1)}_{\text{pred}} = \frac{1}{2|\mathcal{F}_{\text{MA}}|+1}$

for each $t \in [T]$ **do**

  $\Theta_t(x_t) := \underset{\Theta\in\Delta(\mathcal{Y})}{\text{argmin}} \max_{y\in\mathcal{Y}} \mathbb{E}_{\theta\sim\Theta}\left[\sum_{f,\sigma} q^{(t)}_{\text{MA}_{f,\sigma}}\sigma f(x_t)(\mathbb{1}\{y\le\theta\}-\alpha) + q^{(t)}_{\text{pred}}\left(\ell_\alpha(\theta,y)-\ell_\alpha(\tilde{\theta}_t,y_t)\right)\right]$

  Output $\theta_t(x_t) \sim \Theta_t(x_t)$

  $\tilde{q}^{(t+1)}_{\text{MA}_{f,\sigma}} = \frac{q^{(t)}_{\text{MA}_{f,\sigma}} \exp\left(\eta\cdot\mathbb{E}_{\theta\sim\Theta_t(x_t)}[\sigma f(x_t)(\mathbb{1}\{y_t\le\theta\}-\alpha)]\right)}{\sum_{\ell'\in\mathcal{L}} q^{(t)}_{\ell'}\exp\left(\eta\cdot\mathbb{E}_{\theta\sim\Theta_t(x_t)}[\ell'(\theta,x_t,y_t)]\right)}$ for all $f \in \mathcal{F}_{\text{MA}}, \sigma \in \{\pm 1\}$

  $\tilde{q}^{(t+1)}_{\text{pred}} = \frac{q^{(t)}_{\text{pred}} \exp\left(\eta\cdot\mathbb{E}_{\theta\sim\Theta_t(x_t)}[(\ell_\alpha(\theta,y_t)-\ell_\alpha(\tilde{\theta}_t(x_t),y_t)]\right)}{\sum_{\ell'\in\mathcal{L}} q^{(t)}_{\ell'}\exp\left(\eta\cdot\mathbb{E}_{\theta\sim\Theta_t(x_t)}[\ell'(\theta,x_t,y_t)]\right)}$

  $q^{(t+1)}_{\text{MA}_{f,\sigma}} = (1-\gamma)\,\tilde{q}^{(t+1)}_{\text{MA}_{f,\sigma}} + \frac{\gamma}{2|\mathcal{F}_{\text{MA}}|+1}$

  $q^{(t+1)}_{\text{pred}} = (1-\gamma)\,\tilde{q}^{(t+1)}_{\text{pred}} + \frac{\gamma}{2|\mathcal{F}_{\text{MA}}|+1}$

Output: Sequence of (randomized) quantile predictors $\theta_1, \ldots, \theta_T$

D EXTENSIONS

Our general algorithm can be extended to several problems to achieve locally adaptive objective minimization. We present the extensions in Table 1.

	Objectives	Interpretation
Omniprediction	$\ell(p_t(x_t), y_t) - \ell(f(x_t), y_t)$	p_t minimizes losses $\ell \in \mathcal{L}'$ against competitor functions $f \in \mathcal{F}$
Multi-group learning	$\mathbb{1}\{x_t \in g\}(\ell(p_t(x_t), y_t) - \ell(f(x_t), y_t))$	p_t minimizes ℓ for groups $g \in \mathcal{G}$ against competitor functions $f \in \mathcal{F}$
Multi-distribution learning	$(\ell_{\mathcal{D}_i}(p_t(x_t), y_t) - \ell_{\mathcal{D}_i}(f(x_t), y_t))$	p_t minimizes ℓ for distributions $\mathcal{D}_i,\ i \in [k]$ against competitor functions $f \in \mathcal{F}$

Table 1: Examples of extensions of our general algorithm. We define the problem, set of objectives, and the interpretation of the objectives. $\mathcal{L}'$ denotes a finite class of losses.

E Additional Experimental Details

E.1 COMPAS dataset

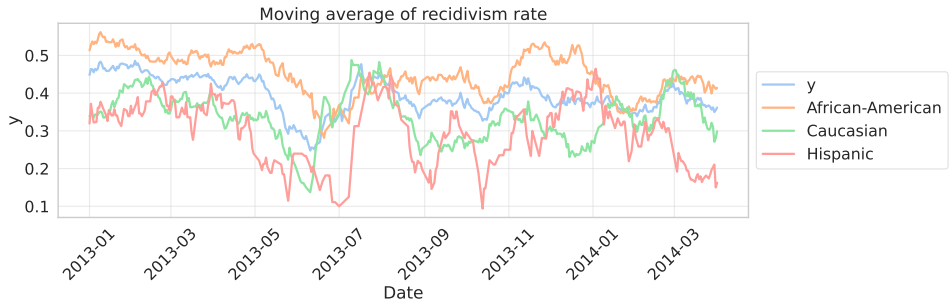


Figure 3: **COMPAS dataset.** Moving average of true recidivism over time. We show 30-day moving averages of y (recidivism indicator), computed overall and separately by racial group. For each calendar date, outcomes are first averaged across all individuals screened that day and then reported as a 30-day rolling mean.

E.2 GEFCom2014 electric load forecasting

For our electric load forecasting experiment, we need to compute the hourly probability that electricity demand exceeds a threshold (150 MW in our example) given quantile forecasts. We use linear interpolation to estimate the full cumulative distribution function of the load from the quantile forecasts of Ziel & Liu (2016). Their method outperforms top entries of the competition. Fix a set of quantile levels $0 < \alpha_1 < \cdots < \alpha_k$ and let the corresponding set of quantile forecasts at hour t be $\hat{\theta}_t^{\alpha_1} < \cdots < \hat{\theta}_t^{\alpha_k}$. Let $Y_t \in \mathbb{R}$ denote the hourly load. We estimate the cumulative distribution function of Y by linearly interpolating between the points $\{(\hat{\theta}_t^{\alpha_i}, \alpha_i)\}_{i=1}^k$. Formally, for any $x \in \mathbb{R}$

$$\widehat{\mathbb{P}}(Y \le x) = \begin{cases} 0, & x < \alpha_1, \\ 1, & x \ge \alpha_k, \\ \alpha_{i-1} + \dfrac{\alpha_i - \alpha_{i-1}}{\hat{\theta}_t^{\alpha_i} - \hat{\theta}_t^{\alpha_{i-1}}} \left(x - \hat{\theta}_t^{\alpha_{i-1}}\right), & \hat{\theta}_t^{\alpha_{i-1}} \le x < \hat{\theta}_t^{\alpha_i}. \end{cases}$$

Figure 4 shows the constructed baseline predictions $\tilde{p}_t$ for the task using the above procedure along with the raw load variable.

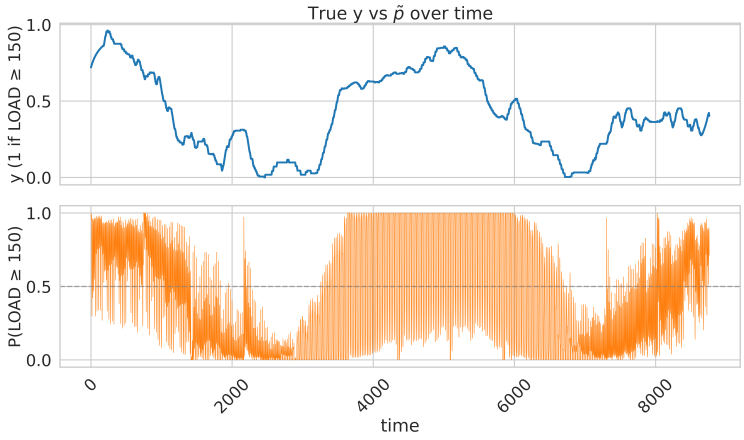


Figure 4: **GEFCom2014-L: True load (y) and constructed predictions $\tilde{p}_t$.** We plot the moving average of the binary y over a window size $|I| = 336$ hours (2 weeks) (top) and the baselines predictions $\tilde{p}_t$ constructed from the quantile forecasts using linear interpolation (bottom) over time.

E.3 Multicalibration implementation

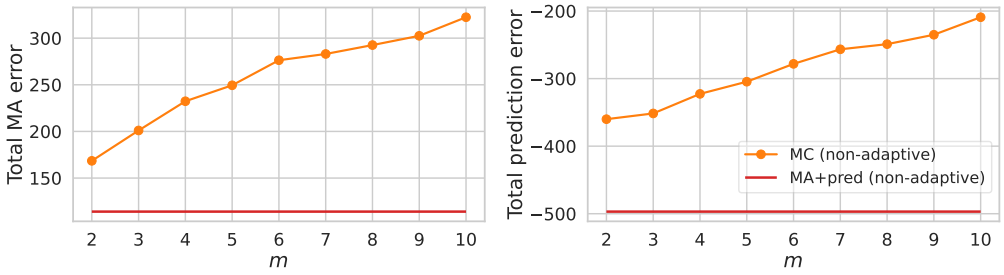
We implement the multicalibration + calibeating algorithm in Lee et al. (2022) and calibeat the baseline forecaster sequence $\tilde{p}_t$. We set the optimal choice of $\eta = \sqrt{\frac{\log(2|\mathcal{L}|m)}{4T}}$ that minimizes the regret bound for the algorithm. We take the number of bins $m = 10$ and 10 level sets of the forecaster throughout. Figure 5 shows the total multiaccuracy error and prediction error for varying values of m. The total multiaccuracy error and prediction error are defined as the sum of the multiaccuracy and prediction errors respectively over all local intervals of width τ. As m decreases, the multicalibration algorithm approaches the multiaccuracy algorithm and the total MA error decreases. Even when $m = 2$, MA+pred has lower total MA error and prediction error than MC on GEFCom2014-L (Figure 5a). While the total MA error of MC drops below MA+pred on COMPAS with smaller m (Figure 5b), this is accompanied by an increase in total prediction error. This method has an additional hyperparameter r used to define a larger action space for the learner. Following Lee et al. (2022), the value of this parameter can be arbitrarily large and we take $r = 1000$.

F Additional Experimental Results

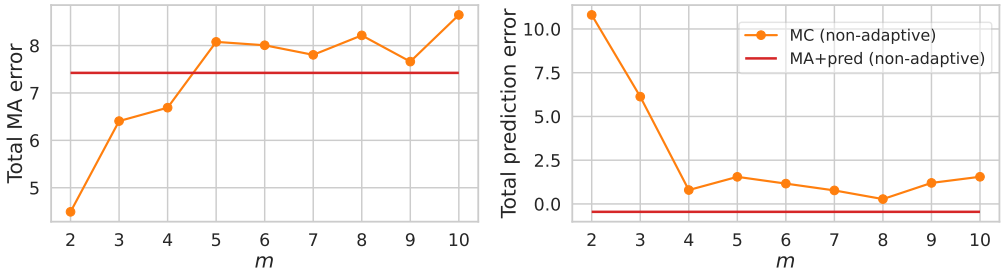
F.1 Local multiaccuracy error evaluation across varying interval widths $|I|$

While we use the fixed width values $\tau = 336$ for GEFCom2014-L and $\tau = 50$ for COMPAS in our locally adaptive algorithm, we perform a general evaluation here over different interval widths $|I|$ in Figures 6 and 7. We find that while adaptivity improves the performance of the multicalibration algorithm, MA+pred (locally adaptive) still has significantly better local multiaccuracy across all interval widths on both datasets.

Next, we show quantitative results for a wider range of window sizes $|I|$. We plot the total multiaccuracy error over all windows for a wide range of varying interval widths $|I|$. Results in Figure 8 show that locally adaptive MA+pred consistently outperforms all other adaptive algorithms despite



(a) GEFCom2014-L



(b) COMPAS

Figure 5: **Total multiaccuracy error and prediction error with varying m,** (a) GEFCom2014-L and (b) COMPAS. This is the same setting as Figure 2a and Figure 2b where we now vary the number of bins m.

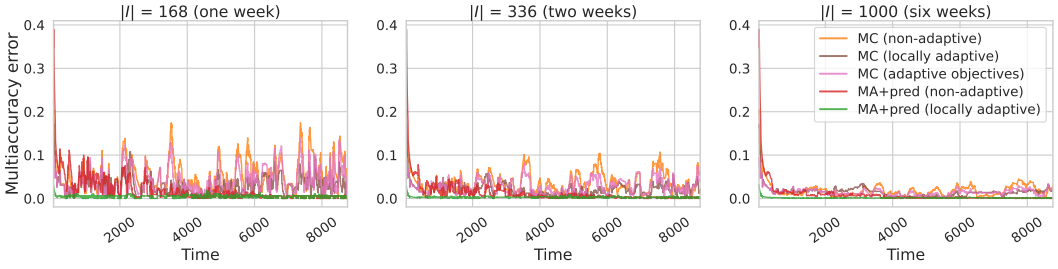


Figure 6: Local multiaccuracy error on GEFCom2014-L for different interval widths. We skip the first thirty time steps when plotting the multiaccuracy error for improved readability.

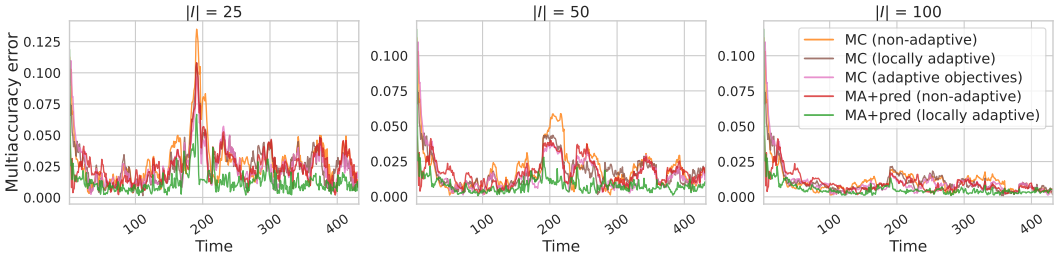
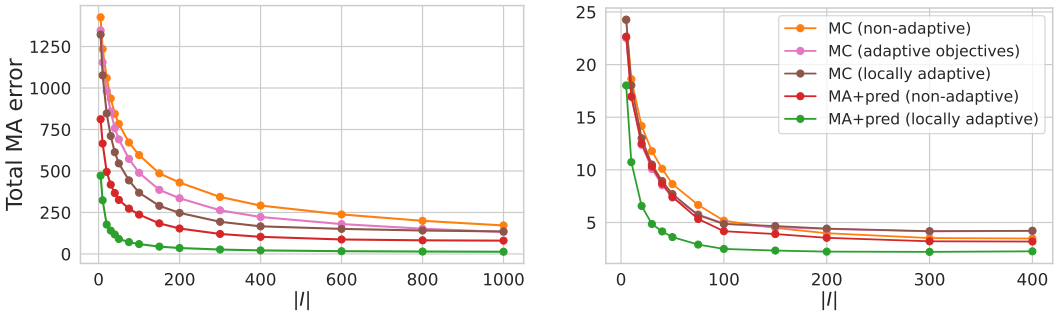


Figure 7: Local multiaccuracy error on COMPAS for different interval widths. We skip the first two time steps when plotting the multiaccuracy error for improved readability.

being tuned with a fixed width. While MC (locally adaptive) improves upon the non-adaptive MC algorithm, the multiaccuracy error remains significantly higher than MA+pred (locally adaptive). It is interesting to note that MC (adaptive objectives) does not achieve lower total multiaccuracy error than MC (locally adaptive) despite its stronger theoretical guarantee over all subintervals.

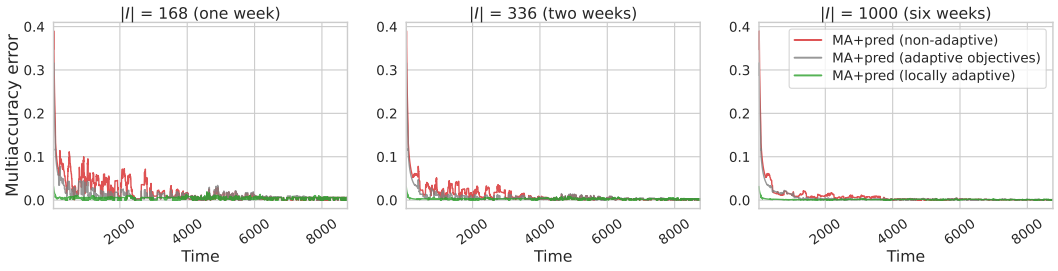


(a) GEFCom2014-L (b) COMPAS

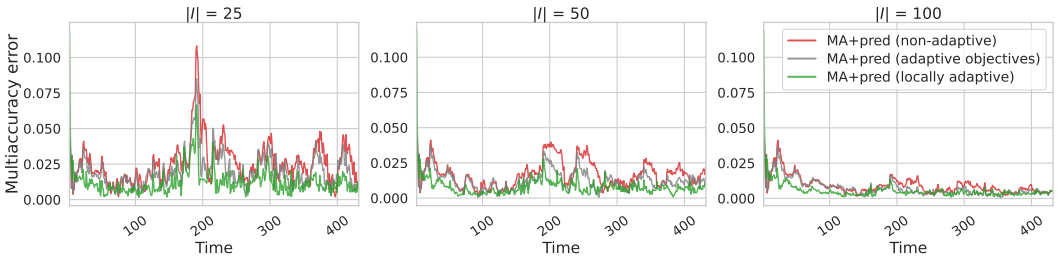
Figure 8: **Total multiaccuracy error with varying interval width $|I|$,** (a) GEFCom2014-L and (b) COMPAS. This is the same setting as Figure 6 and Figure 7. We vary the window width $|I|$ used for the moving average of errors and plot the total multiaccuracy error under the curve.

F.2 Comparison with adaptive objectives MA+pred

In Section 5.3, we compared our proposed locally adaptive MA+pred algorithm with the adaptive online multicalibration algorithm proposed in Lee et al. (2022). Now, we use the adaptive method proposed in Lee et al. (2022) with the MA+pred objectives. See Figures 9 and 10 for the results, where the algorithm is labeled as MA+pred (adaptive objectives). We plot the total multiaccuracy error in Figure 10, which is defined as the sum of the multiaccuracy errors over all local intervals of width $|I|$. Results show that while MA+pred (adaptive objectives) improves the multiaccuracy error over the non-adaptive baseline, it is consistently outperformed by MA+pred (locally adaptive) in all settings. This comparison shows that even when the adaptive baseline has the same objectives, the locally adaptive algorithm exceeds its performance.

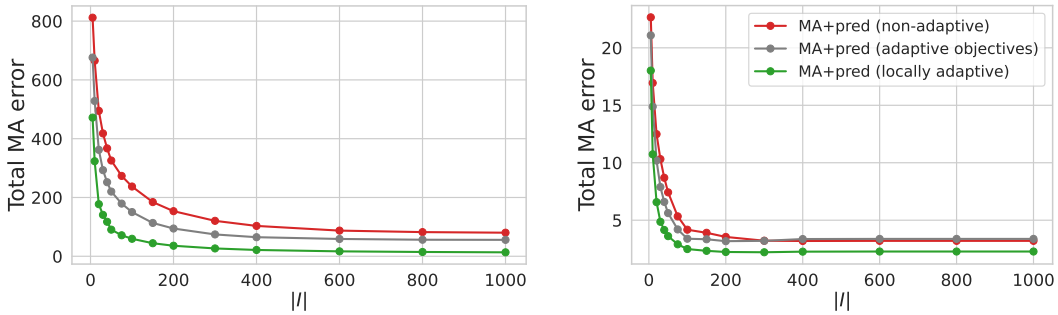


(a) GEFCom2014-L



(b) COMPAS

Figure 9: **Local multiaccuracy error for different interval widths $|I|$,** (a) GEFCom2014-L and (b) COMPAS. This is the same setting as Figure 6 and Figure 7 where we now show comparison with the adaptive MA+pred algorithm.



(a) GEFCom2014-L (b) COMPAS

Figure 10: **Total multiaccuracy error with varying interval width $|I|$,** (a) GEFCom2014-L and (b) COMPAS. This is the same setting as Figure 6 and Figure 7. We vary the window width $|I|$ used for the moving average of errors and plot the total multiaccuracy error under the curve.

G Ablations on Hyperparameters

G.1 Varying η

In this section, we consider three different choices of η in the locally adaptive MA+pred algorithm.

1. $\eta = \sqrt{\frac{\log|\mathcal{L}|}{T}}$: this is the optimal η that minimizes non-adaptive regret bound.
2. $\eta = \sqrt{\frac{\log((2|\mathcal{F}_{\text{MA}}|+1)\cdot 2\tau)+1}{\tau}}$: we substitute the online updates $\sum_{s=t-\tau+1}^{t} {q_{\text{MA}}^{(s)}}^\top {\ell_{\text{MA}}^{(s)}}^2 + q_{\text{pred}}^{(s)} {\ell_{\text{pred}}^{(s)}}^2$ in the adaptive choice of η_t (Section 3) with the interval width τ.
3. $\eta = \eta_t := \sqrt{\frac{\log((2|\mathcal{F}_{\text{MA}}|+1)\cdot 2\tau)+1}{\sum_{s=t-\tau+1}^{t} {q_{\text{MA}}^{(s)}}^\top {\ell_{\text{MA}}^{(s)}}^2 + q_{\text{pred}}^{(s)} {\ell_{\text{pred}}^{(s)}}^2}}$: this is the adaptive choice of η proposed in Section 3, which is the default for our algorithm.

See Figures 11 and 12 for the results that show the local multiaccuracy error and total multiaccuracy error respectively with the above choices of η and varying interval widths. Adaptive η_t consistently dominates, followed by the the choice of η that uses interval width τ. These results establish the importance of the choice of η in achieving local adaptivity separate from the uniform exploration.

G.2 Varying τ and γ

We now vary the fixed interval width τ used for tuning the locally adaptive MA+pred algorithm. This also results in different values of optimal $\gamma = 1/2\tau$. We evaluate the total multiaccuracy error for different choices of τ over windows of varying width $|I|$ in Figure 13. Results show that the total error does not significantly change with different τ values and that the locally adaptive algorithm is robust to the choice of τ.

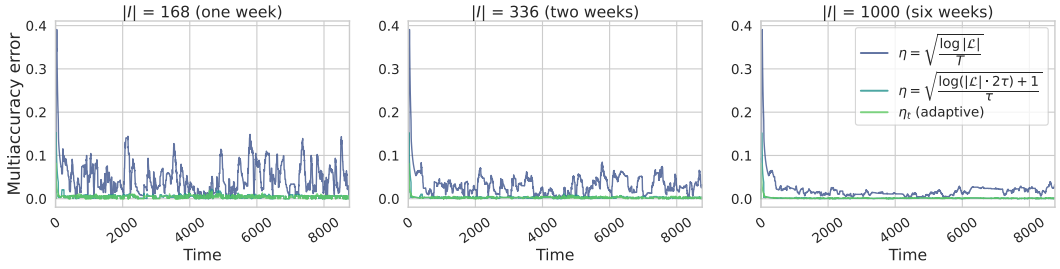
H Simulated Examples

We consider a set of simulated examples where we can control the distribution shifts over time. We focus on a simple setting with a time-varying linear model

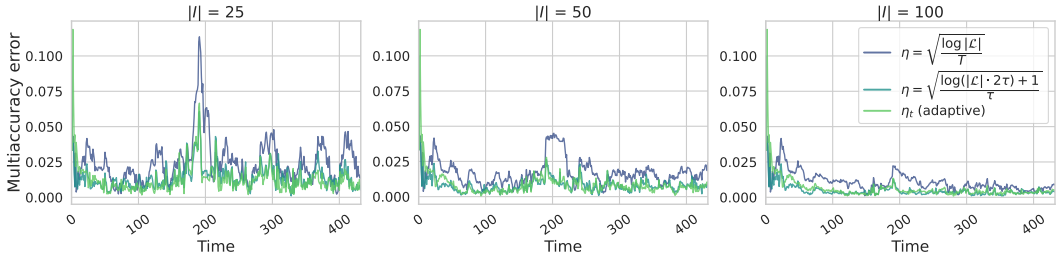
$$Y_t = X_t^\top \beta_t + \varepsilon_t, \qquad \varepsilon_t \sim \mathcal{N}(0, \sigma^2),$$

where the covariates X_t i.i.d. Gaussian,

$$X_t \sim \mathcal{N}(0, I_d),$$

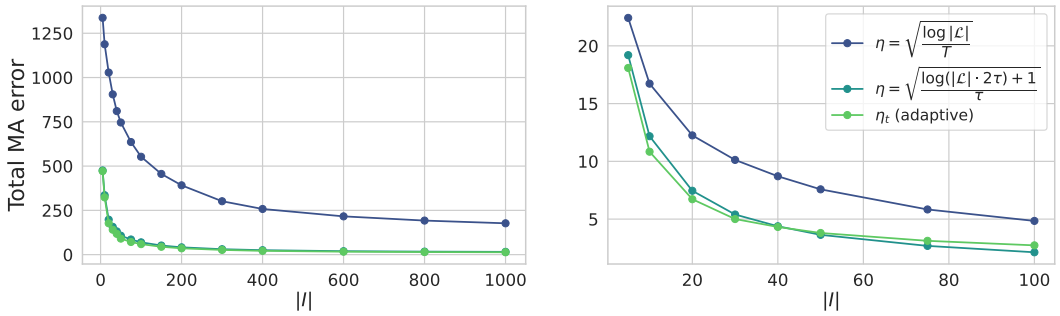


(a) GEFCom2014-L



(b) COMPAS

Figure 11: **Local multiaccuracy error with varying η for different interval widths $|I|$,** (a) GEFCom2014-L and (b) COMPAS. This is the same setting as Figure 6 and Figure 7 where we now show results with different choices of η in the locally adaptive MA+pred algorithm.



(a) GEFCom2014-L

(b) COMPAS

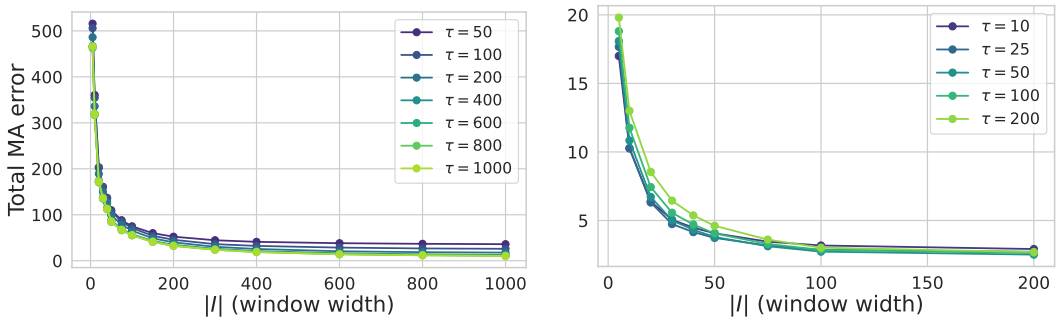
Figure 12: **Total multiaccuracy error with varying η and interval width $|I|$,** (a) GEFCom2014-L and (b) COMPAS. This is the same setting as Figure 6 and Figure 7. We vary the window width $|I|$ used for the moving average of errors and plot the total multiaccuracy error under the curve with different choices of η in the locally adaptive MA+pred algorithm.

and we specify the distribution shift entirely through the coefficients $\beta_t \in \mathbb{R}^d$. The initial $\beta_0 \sim \mathcal{N}\big(0, \frac{1}{d} I_d\big)$ and we set

$$\beta_t \;=\; \beta_0 + \mu_t\, v,$$

where $v \in \mathbb{R}^d$ is a unit direction vector and $(\mu_t)_{t=1}^T$ controls the magnitude of the shift along direction v. We consider *jump* discontinuities in μ_t of varying sizes. Specifically, we divide the time horizon into three equally sized intervals and define μ_t to have small-amplitude jumps in the first and third intervals and large-amplitude jumps in the second interval. We construct three jump–shift datasets (*small, medium*, and *large*) by increasing the small-amplitude range and the large-amplitude range of μ_t as

- *small* shift: μ_t oscillates between $[-0.05, 0.05]$ in the small-amplitude regime and between $[-0.5, 0.5]$ in the large-amplitude regime.



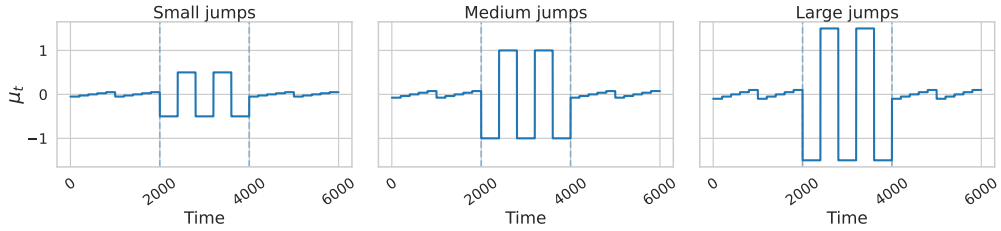
(a) GEFCom2014-L (b) COMPAS

Figure 13: **Total multiaccuracy error for different τ with varying interval width $|I|$,** (a) GEFCom2014-L and (b) COMPAS. This is the same setting as Figure 6 and Figure 7. We vary the fixed width τ used for tuning MA+pred (locally adaptive) and the window width $|I|$ used for the moving average of errors and plot the total multiaccuracy error under the curve.

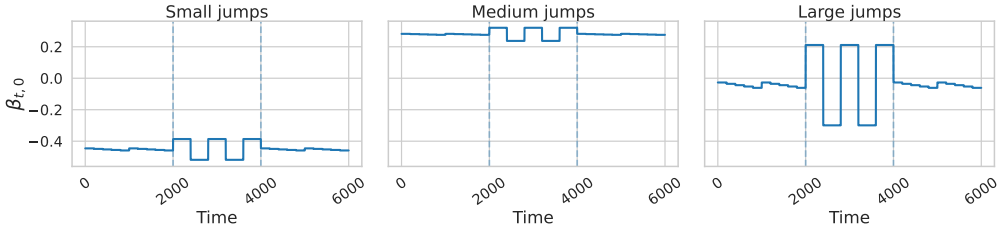
- *medium* shift: μ_t oscillates between $[-0.075, 0.075]$ in the small-amplitude regime and between $[-1.0, 1.0]$ in the large-amplitude regime.
- *large* shift: μ_t oscillates between $[-0.1, 0.1]$ in the small-amplitude regime and between $[-1.5, 1.5]$ in the large-amplitude regime.

We take $d = 5$ in our experiments. Figure 14 shows the final trajectories of μ_t and $\beta_{t,0}$, the first coordinate of β_t, in all three jump shift settings.

We set function class $\mathcal{F}$ to be all the covariates such that $f_j(X) = X_j,\ j \in [d]$. Figure 15 shows the results comparing all algorithms across the three settings. While all algorithms reasonably adapt to the distribution shift in the small jump shift setting, MA+pred (locally adaptive) consistently outperforms all methods as the magnitude of shift increases. The difference is especially substantial in the large jump shift setting. Results from these examples show that the proposed locally adaptive algorithm is able to adapt to discontinuous and abrupt distribution shifts with better rates than existing methods.



(a) Trajectory for μ_t in the *small* (left), *medium* (mid), and *large* (right) jump shift settings.



(b) Trajectory for $\beta_{t,0}$ in the *small* (left), *medium* (mid), and *large* (right) jump shift settings.

Figure 14: **Trajectories for (a) μ_t and (b) $\beta_{t,0}$ in the different jump shift settings.** $\beta_{t,0}$ denotes the first coordinate of β_t. $\beta_{t,j}$ is an affine transformation of μ_t by construction. Dashed vertical lines denote the boundaries between the regime switches where the size of the distribution shift changes.

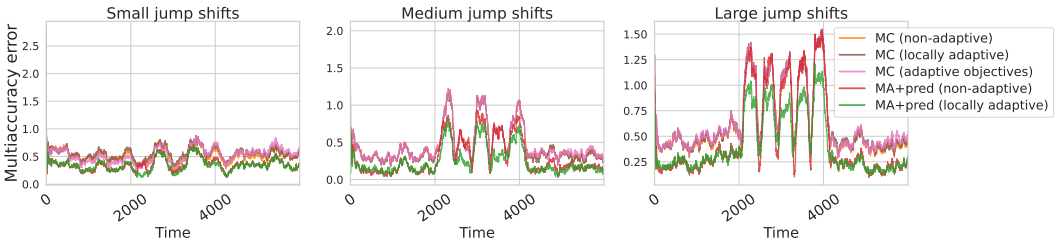


Figure 15: **Local multiaccuracy error in different jump shift settings,** *small* (left), *medium* (mid), and *large* (right) jump shifts.